www.mientayvn.com

Khi đọc qua tài liệu này, nếu phát hiện sai sót hoặc nội dung kém chất lượng xin hãy thông báo để chúng tôi sửa chữa hoặc thay thế bằng một tài liệu cùng chủ đề của tác giả khác. Tài liệu này bao gồm nhiều tài liệu nhỏ có cùng chủ đề bên trong nó. Phần nội dung bạn cần có thể nằm ở giữa hoặc ở cuối tài liệu này, hãy sử dụng chức năng Search để tìm chúng.

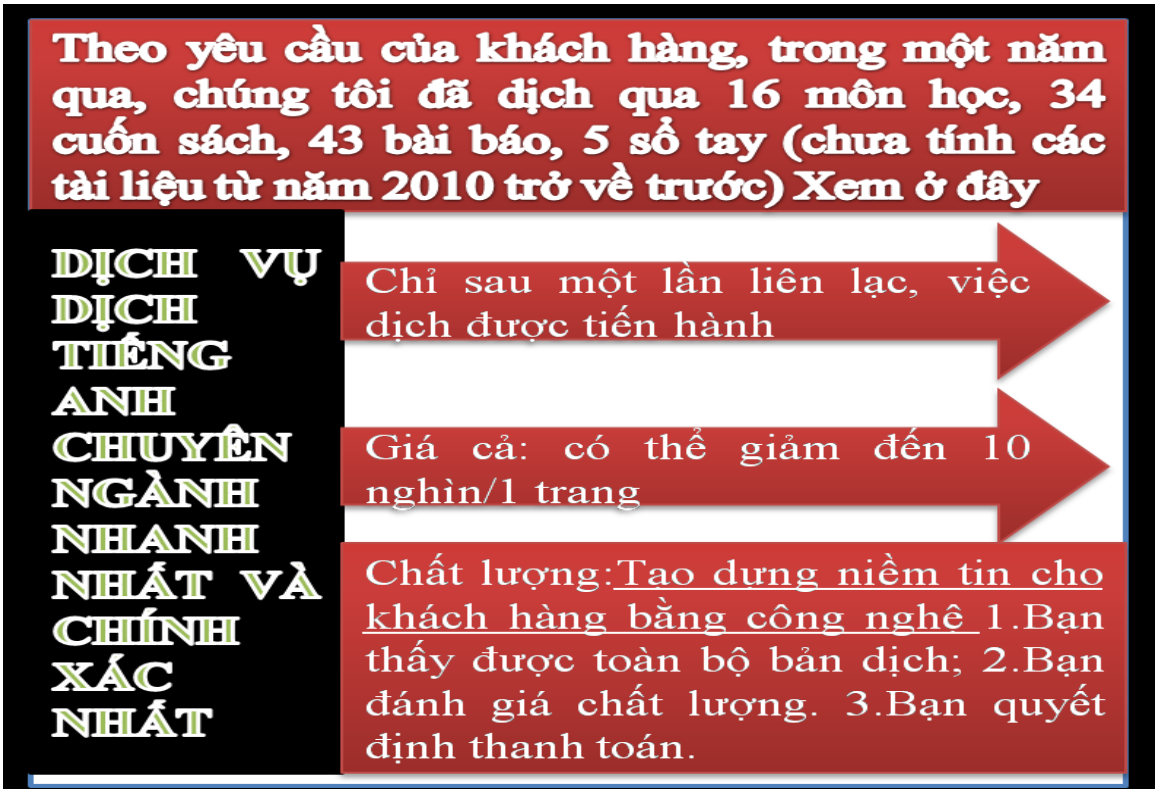
Bạn có thể tham khảo nguồn tài liệu được dịch từ tiếng Anh tại đây:

http://mientayvn.com/Tai_lieu_da_dich.html

Thông tin liên hệ:

Yahoo mail: thanhlam1910_2006@yahoo.com

Gmail: frbwrthes@gmail.com



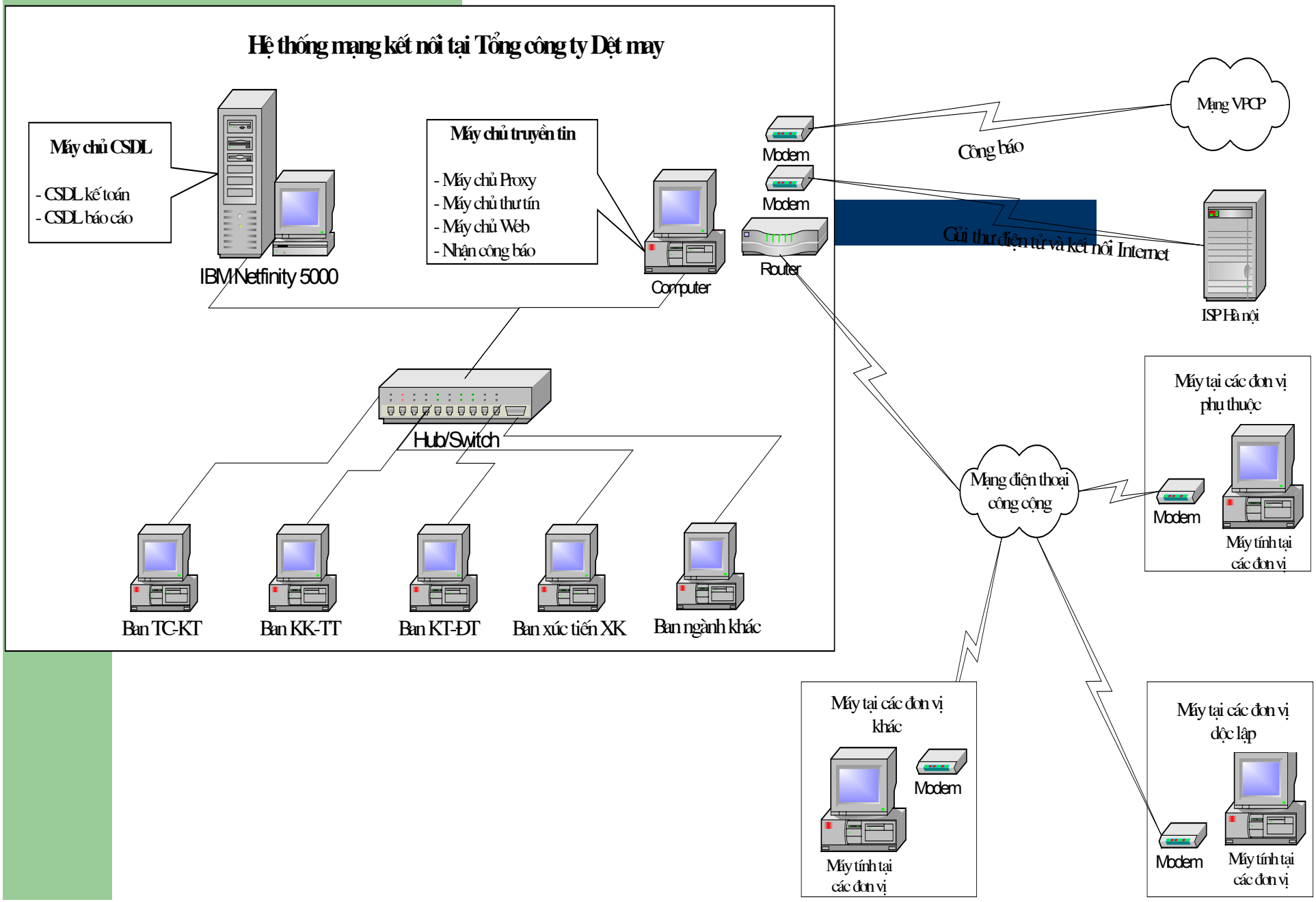
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ – KHOA CÔNG NGHỆ THÔNG TIN

Nhập môn Mạng Máy Tính

Nội dung

- Các kiến thức chung
- Các loại mạng chủ yếu
- Thiết kế mạng
- Mô hình mạng OSI
- Cáp mạng - phương tiện vật lý
- Giao thức
- Kiểm soát lỗi
- Đánh giá độ tin cậy trên mạng
- An toàn thông tin trên mạng
- Quản trị mạng

Bài 1: Các kiến thức chung


Hệ thống mạng kết nối tại Tổng công ty Dệt may
Máy chủ CSDL
- CSDL kế toán
- CSDL báo cáo
IBM Netfinity 5000
Máy chủ truyền tin
- Máy chủ Proxy
- Máy chủ thư tín
- Máy chủ Web
- Nhận công báo
Computer
Modem
Modem
Router
Mạng VPCP
Công báo
Gửi thư điện tử và kết nối Internet
ISP Hà nội
Hub/Switch
Ban TC-KT
Ban KK-TT
Ban KT-ĐT
Ban xúc tiến XK
Ban ngành khác
Mạng điện thoại công cộng
Máy tại các đơn vị phụ thuộc
Modem
Máy tính tại các đơn vị
Máy tại các đơn vị khác
Modem
Máy tính tại các đơn vị
Máy tại các đơn vị độc lập
Modem
Máy tính tại các đơn vị

I. Mạng truyền thông và công nghệ mạng

1. Giới thiệu chung:

- Mạng máy tính là một hệ thống các máy tính tự trị (Autonomous Computer) được kết nối với nhau bởi các đường truyền vật lý và theo một kiến trúc nào đó.
- Từ những năm 70 bắt đầu xuất hiện khái niệm mạng truyền thông (Communication Network) trong đó các thành phần chính của mạng là các nút mạng, được gọi là bộ chuyển mạch (Switching Unit) dùng để hướng thông tin tới đích. Các nút mạng được nối với nhau bằng các đường truyền (Communication Subnet hay Communication Line). Các máy tính xử lý thông tin của người sử dụng - (Host) và các trạm cuối (Terminal) được nối trực tiếp vào các nút mạng khi cần có thể trao đổi thông tin qua mạng. Bản thân các nút thường cũng là một máy tính nên có thể đồng thời đóng vai trò máy của người sử dụng.

1. Giới thiệu chung

Các máy tính được kết nối thành mạng nhằm:

- Làm cho các tài nguyên có giá trị cao, đắt tiền (thiết bị, chương trình, dữ liệu,...) trở nên khả dụng đối với mọi người trên mạng, không phụ thuộc vào khoảng cách địa lý.
- Tăng độ tin cậy của hệ thống nhờ khả năng thay thế khi xảy ra sự cố đối với một máy nào đó.

2. Khái niệm về mạng

- Ở mức độ cơ bản nhất, mạng bao gồm hai máy tính nối với nhau bằng cáp sao cho có thể dùng chung dữ liệu. Trong mọi mạng máy tính, dù có phức tạp đến đâu chăng nữa, chúng cũng đều bắt nguồn từ hệ thống đơn giản đó.
- Mạng máy tính phát sinh từ nhu cầu muốn chia sẻ và dùng chung tài nguyên. Nếu không có hệ thống mạng, để gửi thông tin từ một máy tính này đến một máy tính khác, dữ liệu tin phải được in ra giấy hoặc ghi ra đĩa mềm hoặc các thiết bị nhớ ngoài để chuyển đi.

2. Khái niệm về mạng

- Các máy tính khi đã được nối mạng với nhau, chúng có thể dùng chung các tài nguyên như:
  - Dữ liệu
  - Thông điệp
  - Hình ảnh
  - Máy fax
  - Modem
  - Các tài nguyên khác…

2. Khái niệm về mạng

Mạng liên quan đến nhiều vấn đề bao gồm:

- Giao thức truyền thông (protocol): Mô tả những nguyên tắc mà các thành phần mạng cần phải tuân thủ để có thể trao đổi được với nhau.
- Topo (mô hình ghép nối mạng): Mô tả cách thức nối các thiết bị với nhau.
- Địa chỉ: Mô tả cách định vị một thực thể
- Định tuyến (routing): Mô tả cách dữ liệu được chuyển từ một thiết bị này sang một thiết bị khác thông qua mạng.
- Tính tin cậy (reliability): Giải quyết vấn đề tính toàn vẹn dữ liệu, đảm bảo rằng dữ liệu nhận được chính xác như dữ liệu gửi đi.

2. Khái niệm về mạng

- Khả năng liên tác (interoperability): Chỉ mức độ các sản phẩm phần mềm và phần cứng của các hãng sản xuất khác nhau có thể giao tiếp với nhau trong mạng.
- An ninh (security): Gắn liền với việc đảm bảo an toàn hoặc bảo vệ tất cả các thành phần của mạng.
- Chuẩn hoá (standard): Thiết lập các quy tắc và luật lệ cụ thể cần phải được tuân theo.

3. Tại sao phải dùng mạng?

- **Thiết bị ngoại vi:** Máy in và các thiết bị ngoại vi khác: Trước khi mạng máy tính được đưa vào sử dụng, người ta thường phải tự trang bị máy in, máy vẽ cho máy tính của riêng mình, và mọi người phải thay phiên nhau ngồi trước máy tính được nối với máy máy in đó.
- **Dữ liệu:** Nếu không có mạng máy tính, việc chia sẻ thông tin sẽ bị giới hạn ở: phải truyền đạt thông tin trực tiếp (bằng miệng), gửi thư thông báo, chép thông tin vào đĩa mềm để chuyển thông tin điện tử sang máy tính khác.
- **Ứng dụng:** Mạng được dùng để chuẩn hoá các ứng dụng, chẳng hạn chương trình xử lý văn bản, nhằm đảm bảo rằng mọi người dùng trên mạng đều sử dụng cùng phiên bản của cùng ứng dụng.

4. Thế nào là một mạng máy tính

Mạng bao gồm nhiều thành phần và được nối với nhau theo một cách thức nào đó và sử dụng chung 1 ngôn ngữ:

- Các thiết bị đầu cuối (end system) kết nối với nhau tạo thành mạng có thể là các máy tính hoặc các thiết bị khác.
- Môi trường truyền (media) mà truyền thông được thực hiện qua đó. Môi trường truyền có thể là các loại dây dẫn (cáp), sóng (đối với mạng không dây).
- Giao thức (protocol) là quy tắc quy định cách thức trao đổi dữ liệu giữa các thực thể.

4. Thế nào là một mạng máy tính

- Các thành phần mạng: thiết bị, nút, máy tính
  - Thiết bị được dùng để nối đến bất cứ một thực thể phần cứng nào. Những thực thể này có thể là các thiết bị cuối như: máy tính, máy in, … hoặc một thiết bị phần cứng đặc biệt liên quan đến mạng, ví dụ như các server truyền thông, repeater (bộ lặp), bridge (cầu), switch, router (bộ định tuyến), …
  - Các thiết bị mạng đều dùng 1 số phương pháp cho phép xác định duy nhất chúng, thường thì thiết bị được chính hãng sản xuất gắn 1 số nhận dạng duy nhất. Ví dụ card Ethernet được gán 1 địa chỉ duy nhất bởi hãng sản xuất – địa chỉ này không trùng với bất kỳ địa chỉ nào khác.
  - Khi mô tả các thành phần mạng cần phân biệt giữa khái niệm thiết bị và máy tính. Xem xét ở khía cạnh mạng máy tính thường được gọi là host (hoặc server) hoặc trạm làm việc.

4. Thế nào là một mạng máy tính

- Phương tiện và giao thức truyền thông trên mạng
  - Để chia sẻ thông tin và sử dụng các dịch vụ trên mạng, các thành phần của mạng phải có khả năng truyền thông được với nhau.
  - Để đáp ứng được yêu cầu này chúng ta phải xét tới hai tiêu chí cụ thể của mạng: khả năng liên kết và ngôn ngữ.
  - Khả năng liên kết chỉ đường truyền hoặc kết nối vật lý giữa các thành phần
  - Ngôn ngữ chỉ 1 bảng từ vựng cùng các quy tắc truyền thông mà các thành phần phải tuân theo.

4. Thế nào là một mạng máy tính

- Phương tiện truyền thông (media)
  - Môi trường vật lý được sử dụng để kết nối các thành phần của mạng thường được gọi là phương tiện truyền thông.
  - Phương tiện truyền thông mạng được chia thành 2 loại:
    - Cáp (cable): ví dụ cáp xoắn đôi, cáp đồng trục và cáp sợi quang
    - Không dây (wireless): Có thể là sóng radio (sóng cực ngắn hay truyền thông thông qua vệ tinh), bức xạ hồng ngoại.

II. Các yếu tố của mạng máy tính

1. Đường truyền vật lý:

Đường truyền vật lý dùng để chuyển các tín hiệu điện tử giữa các máy tính. Các tín hiệu điện tử đó biểu thị các giá trị dữ liệu dưới dạng các xung nhị phân (on- off). Chúng hoặc là các sóng điện từ hoặc là tia hồng ngoại. Hiện nay có hai loại đường truyền: hữu tuyến (cable) và vô tuyến (wireless).

1. Đường truyền vật lý:

- Đường truyền hữu tuyến gồm có:
  - Cáp đồng trục (coaxial)
  - Cáp đôi xoắn (twisted -pair cable), có hai loại bọc kim (shielded) và không bọc kim (nushielded).
  - Cáp sợi quang (fiber-optic cable).
- Đường truyền vô tuyến gồm có:
  - Radio
  - Sóng cực ngắn (viba) (microware).
  - Tia hồng ngoại (infrared)

2. Kiến trúc mạng

- Kiến trúc mạng máy tính (network architecture) thể hiện cách nối các máy tính với nhau và ***tập hợp các quy tắc, quy ước*** mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để đảm bảo cho mạng hoạt động tốt. Cách kết nối các máy tính được gọi là ***hình trạng*** hay ***topo*** của mạng, còn tập các quy tắc, quy ước truyền thông gọi là ***các giao thức*** (protocol) của mạng. Tôpô và giao thức mạng là hai khái niệm rất căn bản của mạng máy tính.

2. Kiến trúc mạng

a) Tôpô mạng.

Có hai kiểu kết nối mạng chủ yếu là điểm - điểm (Point to point) và khuếch tán (Broadcast hay Point to multipoint).

- ***Kiểu điểm - điểm***

  Theo kiểu nối này, các đường truyền nối từng cặp nút với nhau và mỗi nút đều có trách nhiệm lưu trữ tạm thời sau đó khi đường truyền rỗi, nó sẽ chuyển tiếp dữ liệu đi cho tới đích. Do vậy mà mạng loại này còn được gọi là mạng "lưu và chuyển tiếp" (store and forward). Nói chung các mạng diện rộng sử dụng nguyên tắc này.

2. Kiến trúc mạng

- ***Kiểu khuếch tán***

  - Theo kiểu nối này, tất cả các nút (các máy tính) dùng chung một đường truyền vật lý. Dữ liệu chuyển đi từ một máy nào đó (một nút) có thể được tất cả các máy khác tiếp nhận. Chỉ cần chỉ ra địa chỉ đích của dữ liệu để mỗi nút kiểm tra xem dữ liệu có phải gửi cho mình hay không.

  - Trong các tôpô dạng ***xa lộ*** (bus) và ***dạng vòng*** (ring) cần có cơ chế "trọng tài" để giải quyết "xung đột" khi nhiều nút muốn truyền tin cùng một lúc. Việc cấp phát đường truyền có thể là "tĩnh" hoặc là "động". Cấp phát "tĩnh" thường dùng cơ chế quay vòng (round robin) để phân chia đường truyền theo các khoảng thời gian định trước. Còn cấp phát "động" là cấp phát theo yêu cầu để hạn chế thời gian "chết" vô ích của đường truyền.

2. Kiến trúc mạng

b) Giao thức mạng

- Việc trao đổi thông tin cho dù đơn giản nhất, đều phải tuân theo những quy tắc nhất định. Hai người nói chuyện muốn cho cuộc nói chuyện kết qủa thì ít nhất cả hai người cũng phải tuân theo nguyên tắc "khi người này nói thì người kia phải nghe và ngược lại".
- Việc truyền tín hiệu trên mạng cũng vậy, cần phải có những quy tắc, quy ước về nhiều mặt từ khuôn dạng (cú pháp, ngữ nghĩa) của dữ liệu cho tới các thủ tục gửi nhận dữ liệu, kiểm soát hiệu quả và chất lượng truyền tin và xử lý các lỗi và sự cố nếu có.
- Tập hợp tất cả các quy tắc, quy ước đó được gọi là giao thức của mạng. Rõ ràng là các mạng có thể tùy ý dùng các giao thức khác nhau tùy sự lựa chọn của người thiết kế.

III. Phân loại mạng máy tính

Có nhiều cách phân loại mạng máy tính tùy thuộc yếu tố chính được chọn để làm chỉ tiêu phân loại, chẳng hạn đó là "khoảng cách địa lý", "kỹ thuật chuyển mạch" hay "kiến trúc mạng",...

1. Nếu lấy **"khoảng cách địa lý"** làm chỉ tiêu phân loại thì ta có các *mạng cục bộ, mạng đô thị, mạng diện rộng và mạng toàn cầu, mạng cá nhân, mạng lưu trữ.*

- **Mạng cục bộ (Local Area Network - viết tắt là LAN)** là mạng được lắp đặt trong một phạm vi tương đối nhỏ (trong một tòa nhà, khu trường học…) với khoảng cách lớn nhất giữa các máy tính nút mạng chỉ trong vòng vài chục mét đến vài km trở lại.

III. Phân loại mạng máy tính

- **Mạng đô thị (Metropolitain Area Networks - viết tắt là MAN)** là mạng được lắp đặt trong phạm vi một đô thị hay một trung tâm kinh tế-xã hội có bán kính khoảng 100 km trở lại.
- **Mạng diện rộng (Wide Area Networks- viết tắt là WAN)** có phạm vi vượt qua biên giới quốc gia thậm chí cả lục địa.
- **Mạng toàn cầu (Global Area Networks - viết tắt là GAN)** có phạm vi trải rộng khắp các lục địa.
- Một loại mạng nữa là **Mạng cá nhân (PAN)** một mạng máy tính nhỏ sử dụng trong gia đình
- *Chú ý rằng khoang cách địa lý dùng làm mốc để phân biệt các loại mạng chỉ có tính tương đối.*

III. Phân loại mạng máy tính

2. Nếu lấy **"kỹ thuật chuyển mạch"** (switching) làm yếu tố chính để phân loại thì ta có: *mạng chuyển mạch kênh, mạng chuyển mạch thông báo và mạng chuyển mạch gói.*

- **Mạng chuyển mạch kênh (circuit-switched networks).** Khi có hai thực thể cần trao đổi thông tin thì giữa chúng sẽ thiết lập một "kênh" (circuit) cố định và duy trì cho đến khi một trong hai bên ngắt liên lạc. Các dữ liệu chỉ được truyền theo con đường cố định đó.
  - ***Có 2 nhược điểm:*** *một là tiêu tốn thời gian để thiết lập kênh cố định giữa hai thực thể và hai là hiệu suất sử dụng đường truyền không cao vì khi hai bên hết thông tin cần truyền, kênh bị bỏ không trong khi các thực thể khác cần không được phép sử dụng kênh.*
  - Mạng điện thoại là một ví dụ điển hình của mạng chuyển mạch kênh

III. Phân loại mạng máy tính

- **Mạng chuyển mạch thông báo (message-switched networks).** Thông báo (message) là một đơn vị thông tin của người sử dụng có khuôn dạng được quy định trước. Mỗi thông báo đều có chứa vùng thông tin điều khiển trong đó chỉ rõ đích của thông báo. Căn cứ vào thông tin này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp theo đường dẫn tới đích của nó. Như vậy, mỗi nút cần phải lưu trữ tạm thời để "đọc" thông tin điều khiển trên thông báo rồi sau đó mới chuyển tiếp thông báo đi. Tùy điều kiện cụ thể của mạng, các thông báo khác nhau có thể được gửi đi trên các con đường khác nhau.

III. Phân loại mạng máy tính

Những ưu điểm:

- Hiệu suất sử dụng đường truyền cao vì không chiếm dụng độc quyền đường truyền mà đường truyền được phân chia giữa nhiều thực thể.
- Mỗi nút mạng có thể lưu trữ thông báo cho tới khi kênh truyền rỗi mới gửi thông báo đi, do đó giảm được tình trạng tắc nghẽn (congestion) mạng.
- Có thể điều khiển việc truyền tin bằng cách sắp xếp theo độ ưu tiên cho các thông báo.
- Có thể tăng hiệu suất sử dụng giải thông của mạng bằng cách gán địa chỉ quảng bá (broadcast addressing) để gửi thông báo đồng thời tới nhiều nút

III. Phân loại mạng máy tính

Những nhược điểm: Không hạn chế kích thước thông báo dẫn đến phí tổn lưu trữ tạm thời cao và ảnh hưởng đến thời gian đáp (response time) và chất lượng truyền. Nó thích hợp với dịch vụ thông tin kiểu thư điện tử (electronic mail) hơn là cho các ứng dụng thời gian thực vì có độ trễ nhất định cho việc lưu trữ và xử lý thông tin điều khiển tại mỗi nút.

- **Mạng chuyển mạch gói (packet-switched networks).** Trong mạng loại này mỗi thông báo được chia ra thành nhiều phần nhỏ hơn gọi là các gói tin (packet) có khuôn dạng quy định trước. Mỗi gói tin cũng có các thông tin điều khiển trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của một thông báo nào đó có thể được chuyển đi qua mạng để tới đích bằng nhiều con đường khác nhau.

III. Phân loại mạng máy tính

- Phương pháp chuyển mạch thông báo và chuyển mạch gói là gần giống nhau. Điều khác biệt là ở chỗ các gói tin được giới hạn kích thước tối đa sao cho các nút mạng (nút chuyển mạch) có thể xử lý toàn bộ gói tin trong bộ nhớ mà không cần phải lưu trữ tạm thời trên đĩa. Chính vì vậy mạng chuyển mạch gói truyền các gói tin qua mạng nhanh hơn và hiệu quả hơn so với mạng chuyển mạch thông báo.
- Vấn đề khó khăn nhất của mạng loại này là việc tập hợp các gói tin để tạo thành bản thông báo ban đầu của người sử dụng, đặc biệt trong trường hợp các gói được truyền theo nhiều đường khác nhau. Cần phải đặt các cơ chế "đánh dấu" gói tin và phục hồi các gói

III. Phân loại mạng máy tính

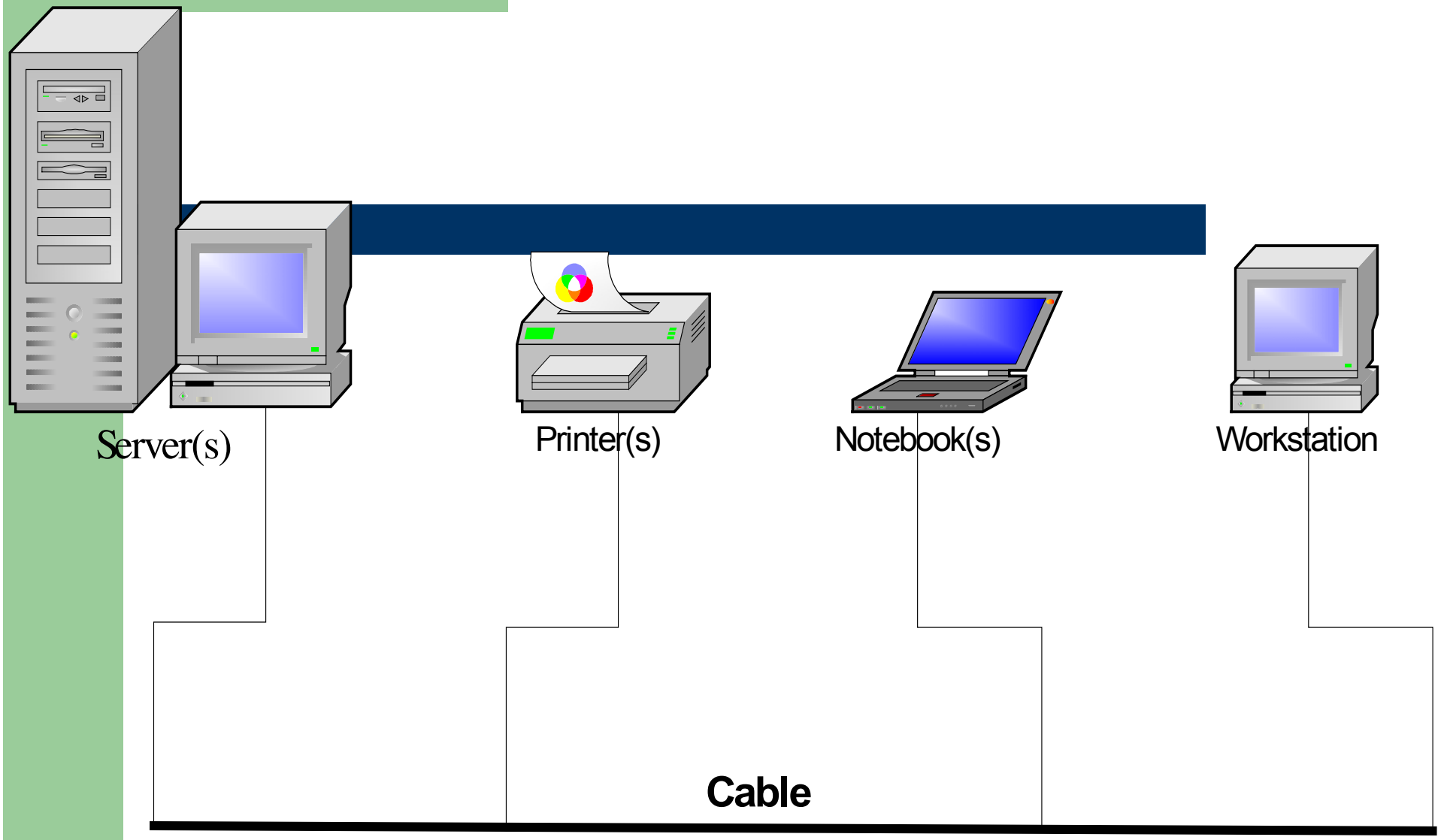
- Do các ưu điểm mềm dẻo và hiệu suất cao hơn nên hiện nay các mạng chuyển mạch gói được dùng phổ biến hơn các mạng chuyển mạch thông báo. Việc tích hợp cả hai kỹ thuật chuyển mạch (kênh và gói) trong một mạng thống nhất (được gọi là mạng dịch vụ tích hợp số - Integrated Service Digital Networks - viết tắt là ISDN) đang là một trong những xu thế phát triển hiện nay.
- Cuối cùng, có thể phân loại mạng theo kiến trúc mạng (tôpô và giao thức sử dụng). Chẳng hạn mạng SNA của IBM, mạng ISO (theo kiến trúc chuẩn quốc tế) hay mạng TCP/IP v.v...

III. Phân loại mạng máy tính

3. Nếu lấy **"kỹ thuật ghép nối"** mô hình Topo gần giống như bản đồ đường phố. Có 3 chiến lược kết nối tổng quát: điểm – điểm (point – to – point), broadcast (điểm – nhiều điểm) và multidrop (đa chặng).

Các cấu hình dạng chuẩn:

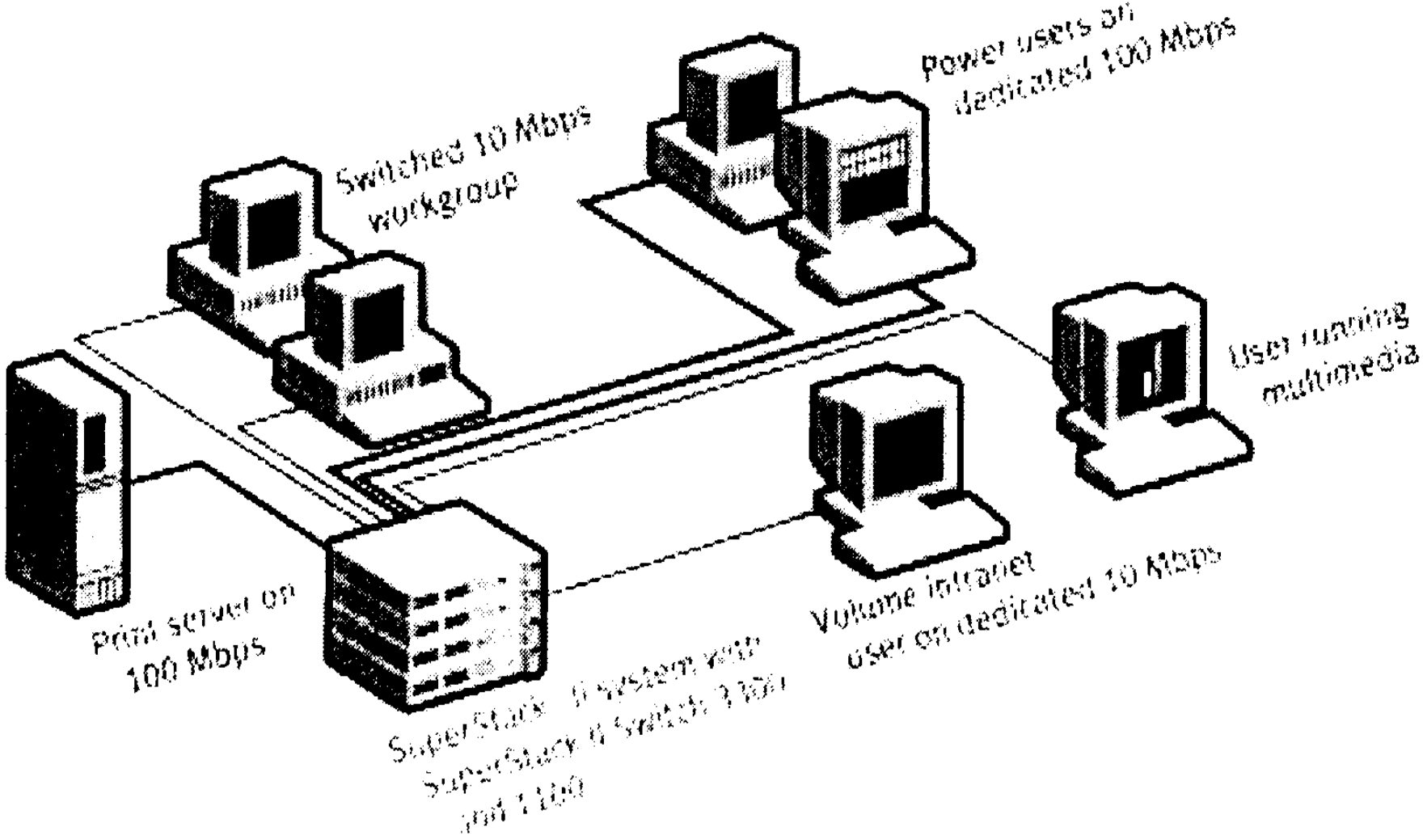
- **Mạng bus**
  - Cấu hình vật lý: bao gồm một dây cáp đơn lẻ nối tất cả các máy tính trong mạng theo một hàng. Đây là phương pháp nối mạng đơn giản và phổ biến nhất.
  - Truyền thông: dữ liệu được gửi và nhận đến một máy tính xác định và đưa dữ liệu đó lên cáp dưới dạng tín hiệu điện tử. Sự hỏng hóc của một máy không ảnh hưởng đến hoạt động của toàn mạng.



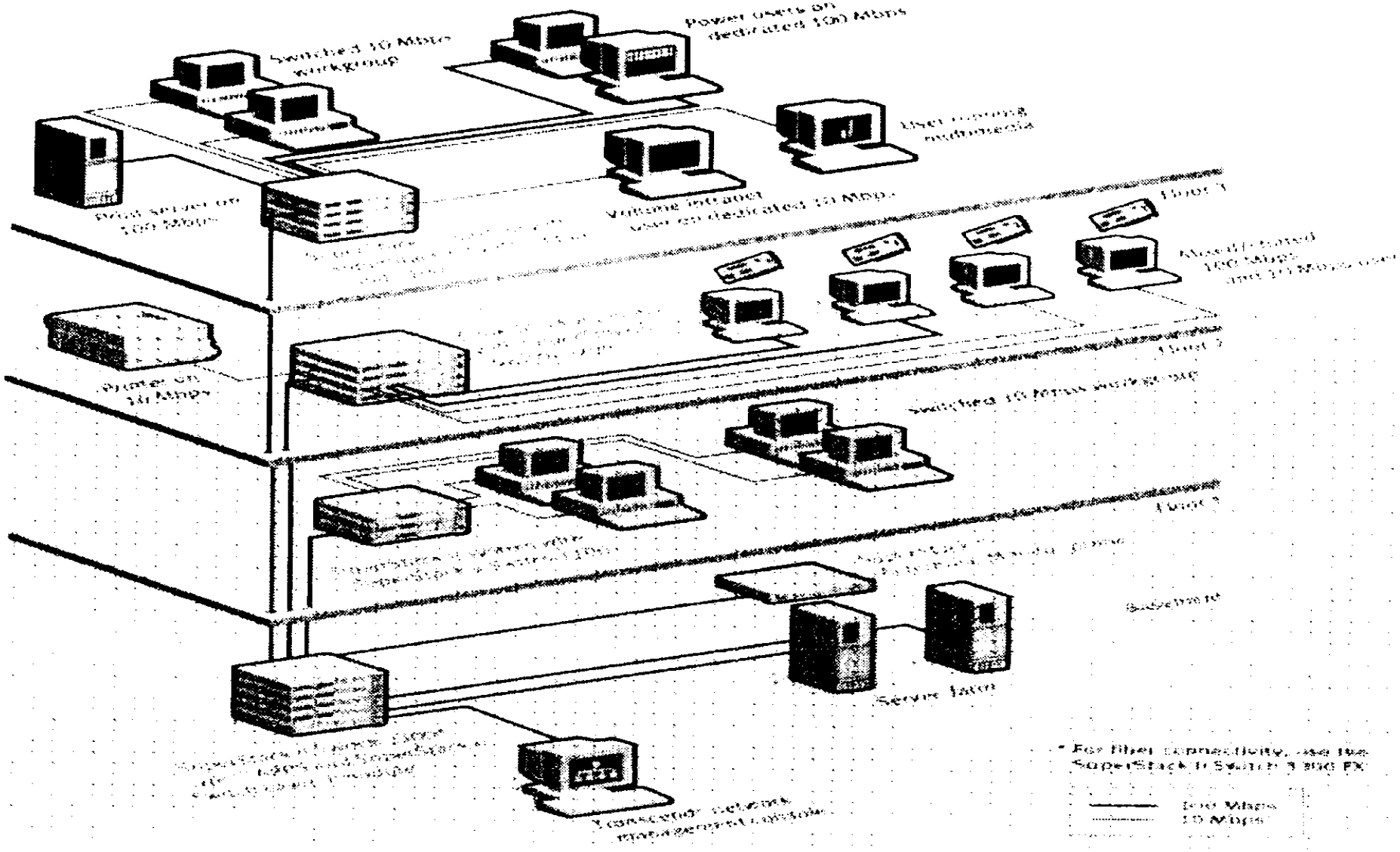
Sơ đồ BUS tuyến tính

III. Phân loại mạng máy tính

- **Mạng star (hình sao)**
  - Cấu hình vật lý: Các máy tính được nối cáp vào một bộ phận được gọi là hub (đầu nối trung tâm). Cấu hình này bắt nguồn từ thời kì đầu, khi việc tính toán dựa trên hệ thống máy tính nối vào một máy chính trung tâm.
  - Truyền thông: Tín hiệu được truyền từ máy tính đến hub để đến tất cả các máy tính trên mạng. Nếu hub trung tâm hỏng, toàn bộ hệ thống mạng sẽ sụp đổ.



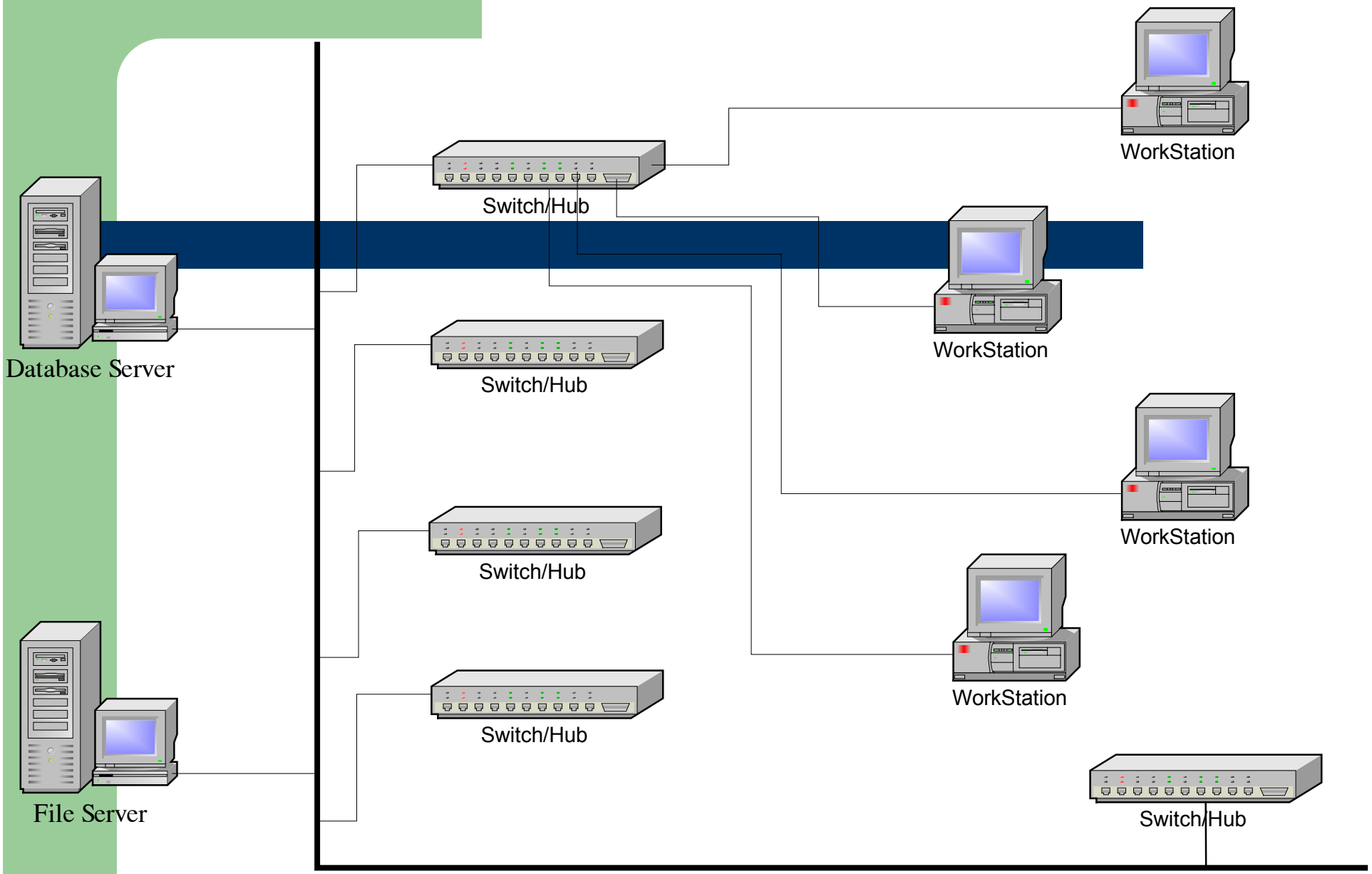
Mô hình mạng hình sao tập trung



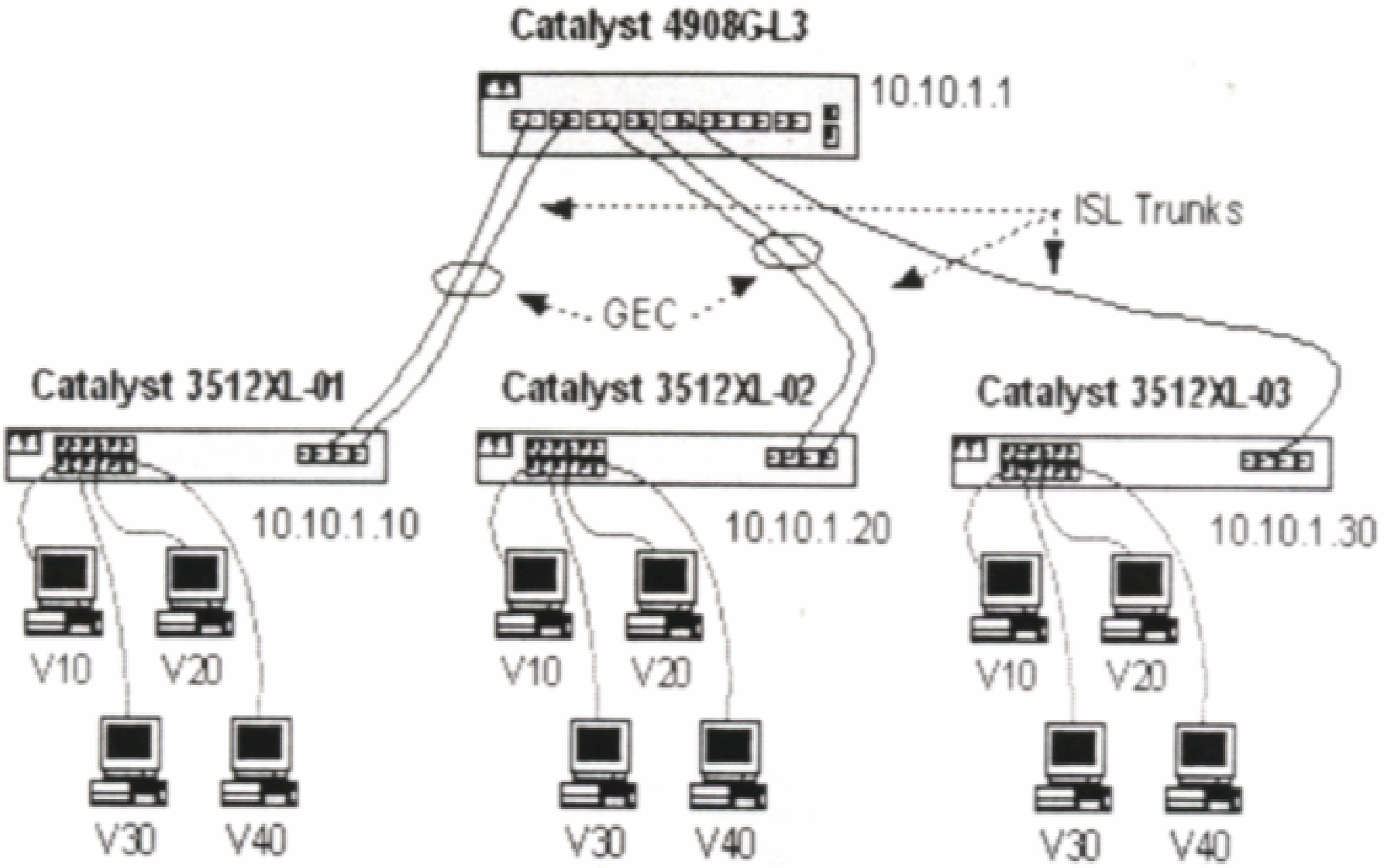
Mô hình mạng hình sao phân tán

III. Phân loại mạng máy tính

- **Mạng ring (vòng khép kín)**
  - Cấu hình vật lý: các máy tính được nối với nhau trên một vòng cáp. Không có đầu nào bị hở.
  - Truyền thông: Tín hiệu được đi qua một chiều và đi qua từng máy tính, mỗi máy tính đóng vai trò như một trạm chuyển tiếp, khuếch đại tín hiệu và gửi nó đến máy tính tiếp theo. Do tín hiệu đi qua từng máy nên sự hỏng hóc của một máy có thể ảnh hưởng đến toàn mạng.
- **Mạng kết hợp**
  - Mạng kết hợp là kiểu ghép nối sắp xếp các máy tính trong mạng kết hợp các cấu hình ghép nối trên (bus, start, ring) để lợi dụng được tối đa ưu nhược điểm của mỗi cấu hình.



Sơ đồ BackBone



Một số ví dụ về kết nối VLAN

Cấu hình mạng	Ưu điểm	Nhược điểm
Bus	Dùng cáp tiết kiệm Phương tiện rẻ tiền và dễ làm việc Đơn giản, đáng tin cậy Để mở rộng	Chạy chậm khi lưu lượng mạng tăng Khó phát hiện và tách ly các vấn đề Cáp đứt có thể ảnh hưởng nhiều đến hoạt động của toàn mạng
Ring	Mọi máy đều có quyền truy cập như nhau Tiến độ thi hành ổn định bất chấp nhiều người dùng	Sự hỏng hóc của một máy tính có thể ảnh hưởng đến các máy còn lại trên mạng Khó phát hiện và tách ly các vấn đề Tái cấu hình mạng sẽ làm mạng ngừng hoạt động
Star	Dễ chỉnh sửa và bổ sung máy tính mới. Theo dõi và quản lý tập trung Sự hỏng hóc của một máy tính không ảnh hưởng đến các máy còn lại trên mạng	Nếu điểm trung tâm bị hỏng thì ảnh hưởng đến toàn mạng.

IV. Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

1. Địa chỉ mạng:

- Gán cho mỗi nút mạng 1 địa chỉ duy nhất – cho phép các thiết bị khác định vị được nó.
  - Ví dụ: Mỗi điện thoại (1 nút) có mã vùng và 1 số (địa chỉ). Mã vùng cung cấp thông tin về vị trí của nút đó trong 1 vùng nào đó, còn số điện thoại là số xác định duy nhất máy điện thoại trong vùng đó. Về thực chất mã mã vùng lại được phân cấp thành mã quốc gia và mã khu vực.

IV. Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

2. Routing – Định tuyến

- Quyết định tuyến đường mà dữ liệu sẽ đi qua khi chuyển từ nút nhận đến nút gửi.
- Chức năng định tuyến được thực hiện bởi 1 thiết bị phần cứng đặc biệt: router (định tuyến).
- Việc lựa chọn tuyến đường tốt nhất phải dựa trên 1 tiêu chuẩn cụ thể - được gọi là độ đo (met).
- Các độ đo định tuyến phổ biến là: khoảng cách, số chặng (hop) vả băng thông.

IV. Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

3. Tính tin cậy:

- Chỉ tính toàn vẹn dữ liệu – đảm bảo rằng dữ liệu nhận được giống hệt dữ liệu được gửi đi. Trong thực tế lỗi có thể xảy ra ở tất cả các môi trường truyền mạng. Vì vậy phải thiết kế sao cho hệ thống có khả năng xử lý lỗi.
- Một trong những chiến lược điển hình là thêm thông tin vào dữ liệu được truyền đi sao cho phía bên nhận phát hiện được lỗi (nếu có). Khi phát hiện lỗi nó có thể thực hiện:
  - Yêu cầu truyền lại dữ liệu bị lỗi
  - Kiểm tra xem dữ liệu đúng là gì và sửa đổi dữ liệu bị truyền lỗi.
- Cách thứ nhất sửa lỗi bằng cách yêu cầu truyền lại, cách thứ hai gọi là khả năng tự sửa lỗi. Việc sửa lỗi nói chung khó thực hiện.

IV. Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

4. Tính liên tác:

- Các sản phẩm của các hãng khác nhau có thể giao tiếp thành công với nhau trên mạng.
- Ngày nay với bộ giao thức "mở" TCP/IP các hãng sản xuất – những người viết và bán các ứng dụng dựa trên TCP/IP được tự do làm những thứ họ muốn, không lo ngại về vi phạm bản quyền.

IV. Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

5. An ninh:

- An ninh mạng chỉ việc bảo vệ mọi thứ liên quan đến 1 mạng bao gồm dữ liệu, phương tiện truyền thông và các thiết bị. An ninh mạng còn bao gồm các chức năng quản trị, các công cụ kỹ thuật và thiết bị như các sản phẩm mã hoá, các sản phẩm kiểm soát truy cập mạng (ví dụ: firewall – thiết bị phần cứng đặc biệt bảo vệ 1 mạng khỏi thế giới bên ngoài).
- An ninh mạng bao gồm việc quy định những chính sáhc sử dụng tài nguyên mạng, kiểm tra xem tài nguyên mạng có được sử dụng phù hợp với chính sách đã định trước hay không, quy định và kiểm tra những người có đủ quyền mới được sử dụng các tài nguyên đó.

V. Mô hình mạng OSI

Truyền thông mạng: Hoạt động mạng là quá trình gửi dữ liệu từ máy tính này sang máy tính khác. Quá trình này có thể được chia thành các tác vụ riêng biệt:

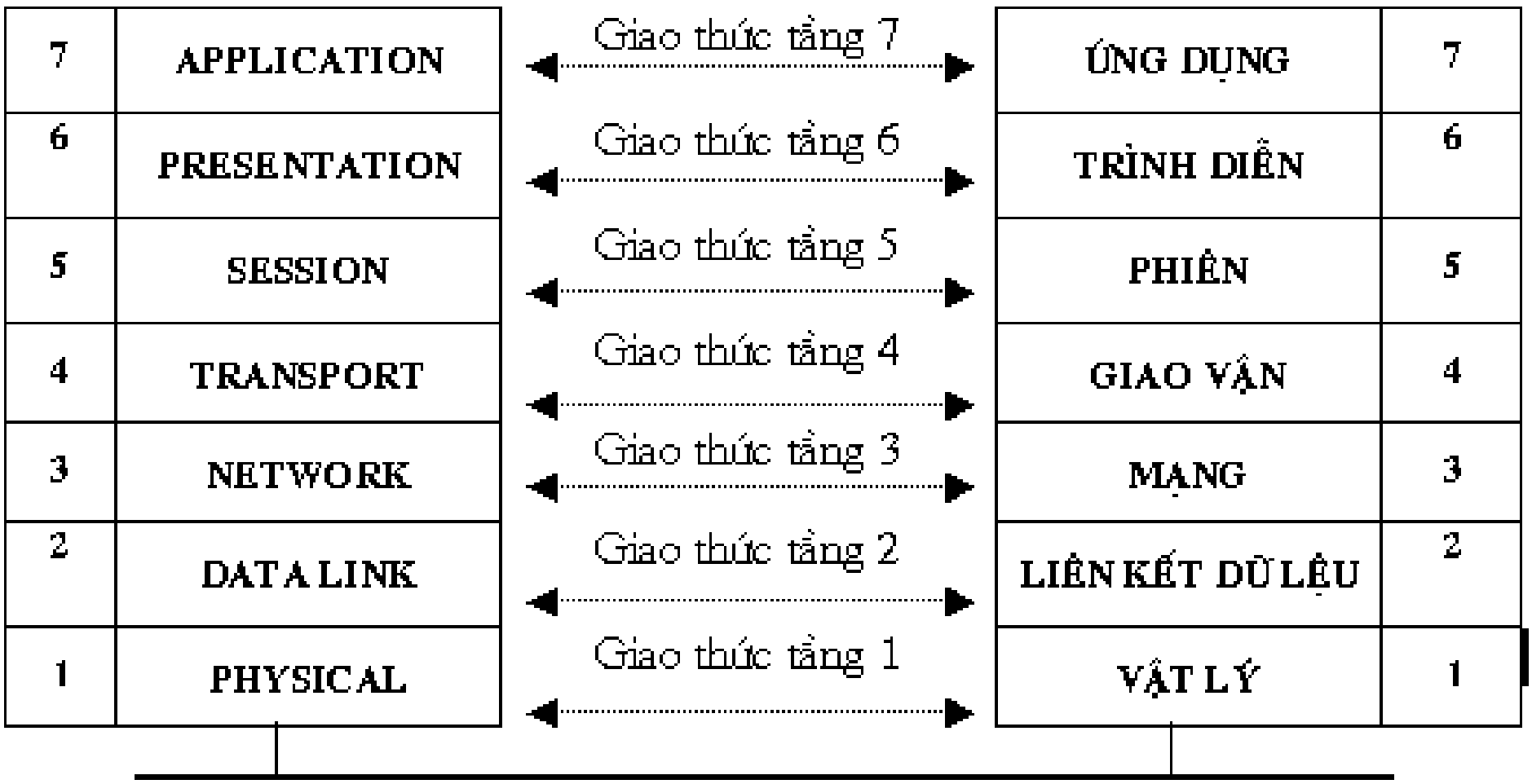
- Nhận biết dữ liệu
- Chia dữ liệu thành từng gói để có thể quản lý được
- Thêm thông tin vào từng gói để xác định địa chỉ máy nhận và vị trí của gói tin.
- Bổ sung thông tin để kiểm tra lỗi và thời lượng
- Đưa dữ liệu lên mạng và gửi đi

Các thủ tục này được HĐH tuân theo một cách nghiêm ngặt, những thủ tục này được gọi là giao thức. Mô hình OSI (Open Systems Interconnection) được tổ chức tiêu chuẩn quốc tế ISO bàn hành để mô tả kiến trúc mạng dành cho việc nối kết những thiết bị không cùng chủng loại.

V. Mô hình mạng OSI

Mô hình OSI:

- Mô hình OSI là kiến trúc chia mạng truyền thông thành 7 tầng.
- Mỗi tầng bao gồm những hoạt động, thiết bị và giao thức mạng khác nhau.
- Mỗi tầng cung cấp dịch vụ hoặc hoạt động chuẩn bị dữ liệu để chuyển giao qua mạng đến máy tính khác.
- Các tầng đều được phân chia bằng ranh giới được gọi là giao diện.
- Mọi yêu cầu đều được chuyển từ tầng này sang tầng khác thông qua giao diện rồi đến tầng tiếp theo. Mỗi tầng đều phải tuân theo chuẩn và hoạt động của tầng bên dưới.



Hình 1.12 Mô hình OSI 7tầng.

V. Mô hình mạng OSI

1. Tầng ứng dụng

- Đóng vai trò như cửa sổ dành cho các hoạt động xử lý của trình ứng dụng nhằm truy nhập các dịch vụ mạng. Tầng này biểu diễn những dịch vụ hỗ trợ trực tiếp các ứng dụng người dùng, như các phần mềm chuyển tập tin, truy cập cơ sở dữ liệu và email

2. Tầng Presentation

- Tầng này quyết định dạng thức dùng trao đổi dữ liệu giữa các máy tính mạng. Tầng Presentation ở máy gửi diễn dịch dữ liệu được truyền từ tầng Ứng dụng sang dạng thức trung gian mà ứng dụng nào cũng có thể nhận biết, phía máy nhận, tầng này kết hợp dữ liệu từ dạng thức trung gian và truyền lên tầng ứng dụng.

V. Mô hình mạng OSI

3. Tầng Session (phiên)

- Cho phép hai chương trình ứng dụng trên các máy tính khác nhau thiết lập, sử dụng và chấm dứt một nối kết gọi là phiên làm việc. Tầng này thi hành các thủ tục cho phép nhận biết tên và thực hiện các chức năng cần thiết như bảo mật. Tiến hành việc đồng bộ hoá bằng cách đặt các điểm check point vào luồng dữ liệu, bằng cách này, nếu mạng bị ngắt, chỉ những dữ liệu sau điểm kiểm tra cuối cùng mới phải chuyển lại.

4. Tầng Transport (giao vận)

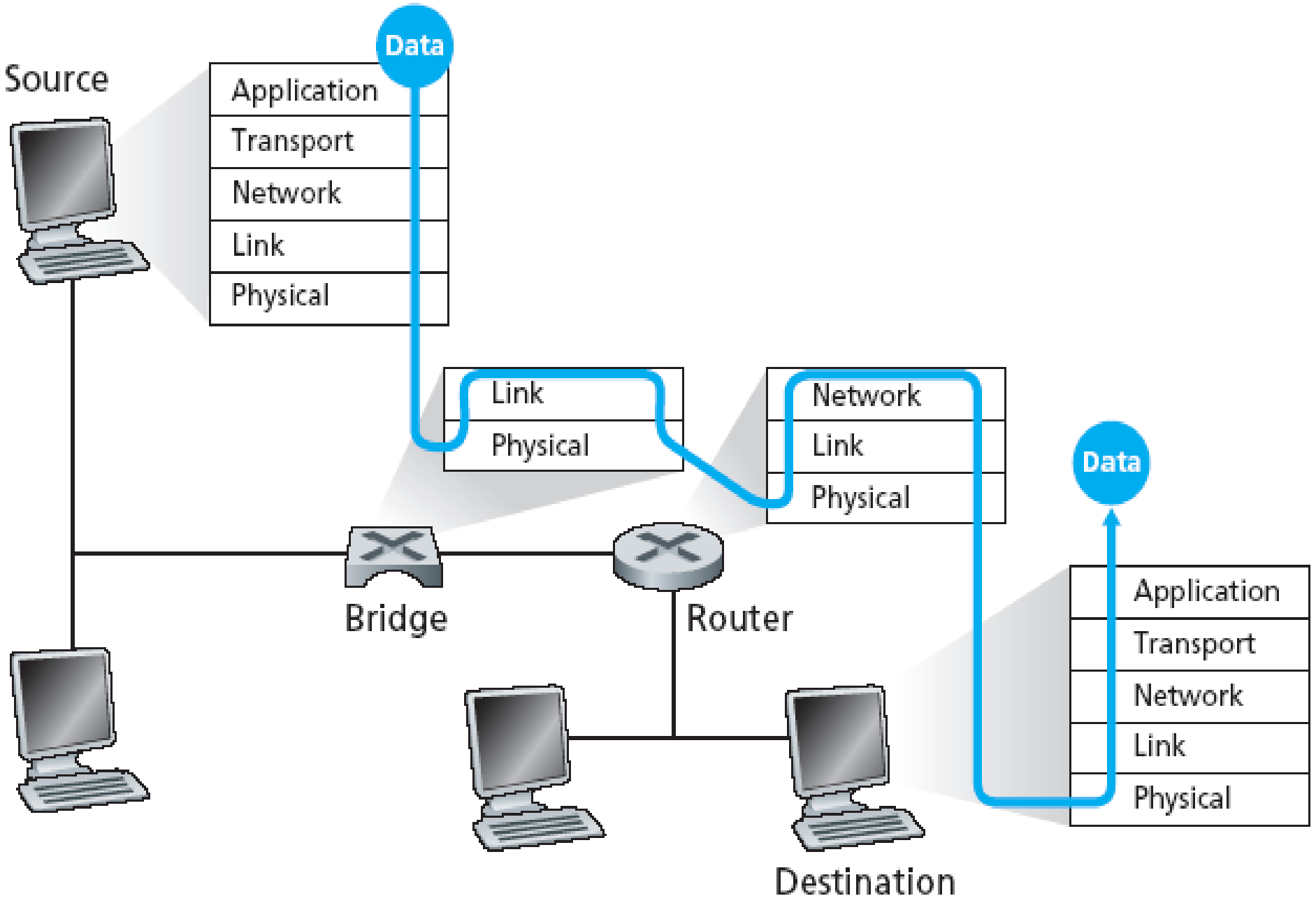
- Tầng này bảo đảm gói tin truyền đi không có lỗi, theo đúng thứ tự, không bị mất mát hay sao chép. Tầng này đóng gói thông điệp, chia thông điệp dài thành nhiều gói. Tại đầu nhận, tầng này mở gói thông điệp, lắp ghép lại cho đúng thứ tự

V. Mô hình mạng OSI

5. Tầng Network (mạng): Chịu trách nhiệm lập địa chỉ các thông điệp, diễn dịch địa chỉ và tên logic thành địa chỉ vật lý. Tầng này quyết định đường đi từ máy chủ đến máy đích, nó sẽ quyết định chọn đường mạng nào để đi

6. Tầng Data Link (Liên kết dữ liệu):Gửi khung dữ liệu từ tầng Network đến tầng Physical. Ở đầu nhận, tầng này đóng gói dữ liệu thô (chưa được xử lý) từ tầng Physical thành khung dữ liệu.

7. Tầng Physical: Tầng này chuyển luồng bit thô qua phương tiện vật lý. Tầng này chịu trách nhiệm liên kết các giao diện hàm, cơ, quang và điện với cáp. Nó định nghĩa cách kết nối cáp với card mạng như thế nào, định rõ từng kĩ thuật truyền nào sẽ được đối với từng loại cáp mạng.


Source
Data
Application
Transport
Network
Link
Physical
Link
Physical
Network
Link
Physical
Data
Bridge
Router
Application
Transport
Network
Link
Physical
Destination

VI. Kết nối các mạng máy tính

- Do nhu cầu trao đổi thông tin ngày càng cao nên việc kết nối các mạng máy tính lại với nhau đã trở thành một vấn đề được quan tâm đặc biệt. Mục tiêu đặt ra là phải làm sao để những người sử dụng trên mạng khác nhau (về chủng loại, về kiến trúc hoặc vị trí địa lý) có thể trao đổi thông tin với nhau một cách dễ dàng và hiệu quả.

VI. Kết nối các mạng máy tính

1. Các chiến lược kết nối.

- Để kết nối các mạng máy tính đang tồn tại lại với nhau người ta thường xuất phát từ hai quan điểm sau:
  - 1. Xem mỗi nút của mạng con như một hệ thống mở.
  - 2. Xem mỗi mạng con như một hệ thống mở.
- Quan điểm 1 cho phép mỗi nút của mạng con có thể truyền thông trực tiếp với mỗi nút của mạng con bất kỳ khác. Như vậy toàn bộ các mạng con cũng sẽ là nút của mạng lớn và tuân thủ một kiến trúc chung.
- Trong khi đó theo cách tiếp cận thứ 2 thì hai nút thuộc hai mạng con khác nhau không thể trực tiếp "bắt tay" nhau được mà phải qua một phần tử trung gian gọi là giao diện nối kết (Interconnection Interface) đặt giữa hai mạng con đó có nghĩa là cũng hình thành một mạng lớn gồm các giao diện kết nối và các máy chủ (Host) được nối với nhau bởi các mạng con.

VI. Kết nối các mạng máy tính

- Tương ứng với hai quan điểm trên có hai chiến lược kết nối các mạng.
  - Tìm cách xây dựng các chuẩn chung cho các mạng (các công trình chuẩn hoá của CCITT và ISO)
  - Cố gắng xây dựng các giao diện nối kết đảm bảo tính độc lập của các mạng con hiện có.
- Sự hội tụ về một chuẩn chung là điều lý tưởng, nhưng thực tế không thể loại bỏ hàng ngàn mạng khác nhau đang tồn tại trên thế giới. Vì vậy trên thị trường xuất hiện hàng loạt các sản phẩm giao diện kết nối cho phép chuyển đổi giữa các mạng khác nhau. Đó là biểu thị tính thực tế hơn của chiến lược thứ 2.

VI. Kết nối các mạng máy tính

2. Giao diện kết nối

- Chức năng cụ thể của một giao diện kết nối phụ thuộc vào sự khác biệt về kiến trúc của các mạng con. Sự khác biệt càng lớn thì chức năng của giao diện càng phức tạp. Một giao diện kết nối có thể thực hiện nối "tay đôi" hoặc "tay ba" hoặc "nhiều tay" tuỳ người thiết kế. Hơn nữa chúng có thể là một máy tính độc lập nhưng cũng có thể được cài đặt ghép vào một nút của một mạng con nào đó.

VI. Kết nối các mạng máy tính

- Tuỳ thuộc vào chức năng cụ thể mà giao diện kết nối có thể có các tên gọi riêng như bridge, router, gateway. Gateway (cửa khẩu) là tên gọi chung nhất cho các giao diện kết nối và thường dùng trong những trường hợp chức năng của các giao diện này là phức tạp, đòi hỏi sự chuyển đổi giữa các họ giao thức khác nhau được dùng trong các mạng con. Trong khi đó bridge (cầu) được dùng để chỉ giao diện kết nối trong trường hợp đơn giản nhất, ví dụ kết nối giữa các mạng cục bộ (LAN) cùng loại. Còn router (bộ chọn đường) hoạt động ở mức cao hơn so với bridge vì nó còn đảm nhận cả việc chọn đường cho các đơn vị dữ liệu để hướng chúng tới đích.

VII. Mô hình tham chiếu TCP/IP

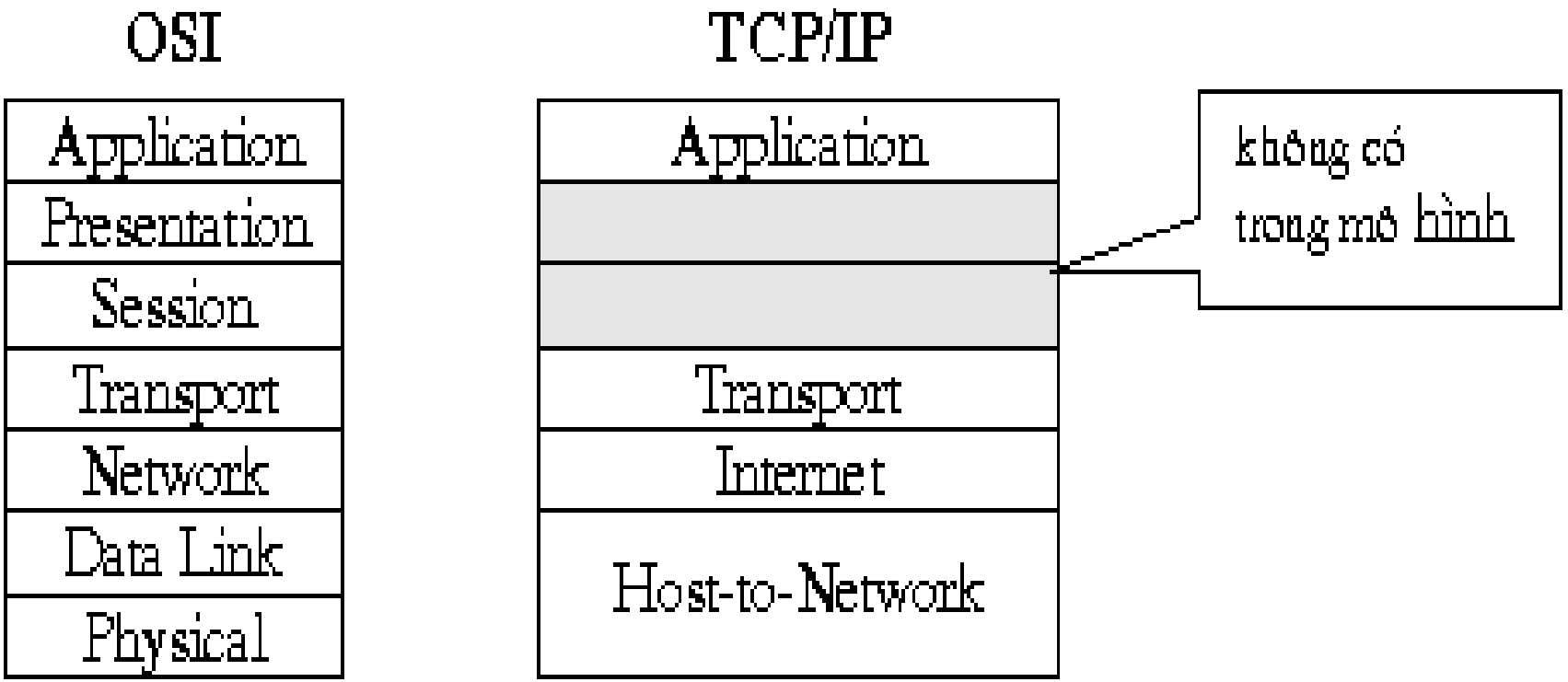
- Mạng ARPANET được xây dựng bởi Bộ Quốc phòng Mỹ từ 1969 với họ giao thức TCP/IP nổi tiếng và là tiền thân của INTERNET ngày nay. Bộ Quốc phòng Mỹ sợ rằng nếu chiến tranh hạt nhân xảy ra, các đường truyền vật lý cũng như các Host, router, Gateway đắt tiền của họ sẽ có thể bị phá hủy ngay từ phút đầu tiên. Mạng ARPANET cần có khả năng hoạt động ngay cả trong trường hợp đó, miễn là máy nguồn và máy đích vẫn còn hoạt động và còn có một đường truyền (vật lý) giữa chúng.
- Sau này khi các mạng vệ tinh và mạng vô tuyến ra đời và bổ sung vào thì các giao thức đang được dùng của ARPANEt không đáp ứng được yêu cầu liên mạng. Cần phải có một mô hình kiến trúc mới có khả năng liên kết nhiều mạng với nhau một cách trong suốt. Kiến trúc này có tên Mô hình tham chiếu TCP/IP, đặt theo tên của 2 giao thức cơ bản của nó.

VII. Mô hình tham chiếu TCP/IP

1. Tầng Internet (The Internet Layer).

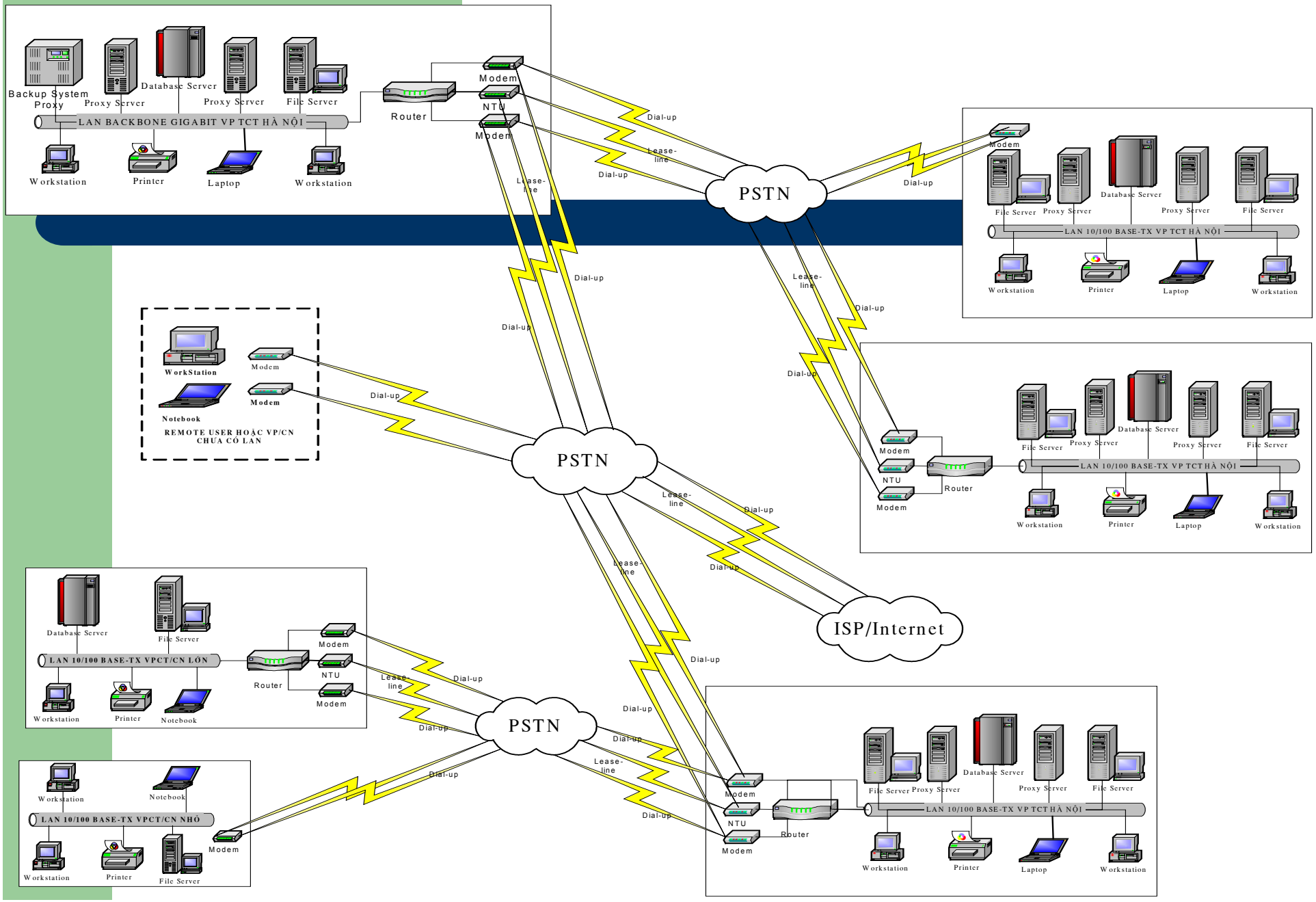
- Các yêu cầu nêu trên dẫn tới một lựa chọn một mạng chuyển mạch gói dựa trên một tầng internetwork không hướng nối. Tầng này được gọi là Internet Layer. Nhiệm vụ của lớp này là chọn đường cho các gói tin (packets routing) và tránh tắc nghẽn (avoiding congestion). Nó cho phép các Host truyền các packet vào mọi mạng, mỗi packet có thể đi đến đích theo các con đường khác nhau, thứ tự nhận các gói tin có thể khác với thứ tự mà chúng được gửi đi, các tầng trên phi tự giải quyết vấn đề thứ tự các packet.
- Tầng internet định nghĩa một khuôn dạng packet và giao thức chính thức (official) được gọi là IP (Intetrnet Protocol). Công việc của Tầng Internet là phân phát các IP packet tới đích của chúng.

VII. Mô hình tham chiếu TCP/IP



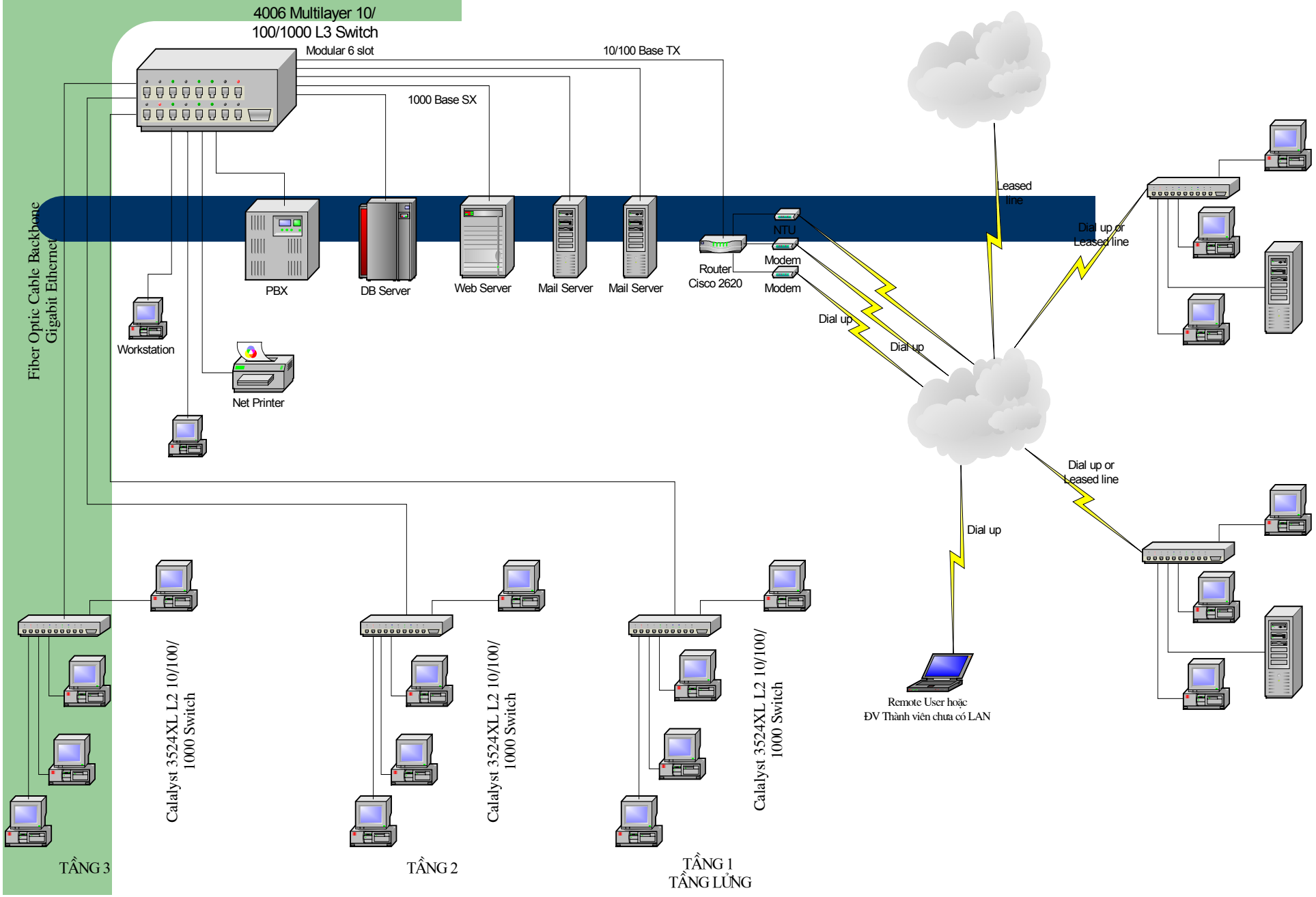
Hình 1. 13 Mô hình tham chiếu TCP/IP

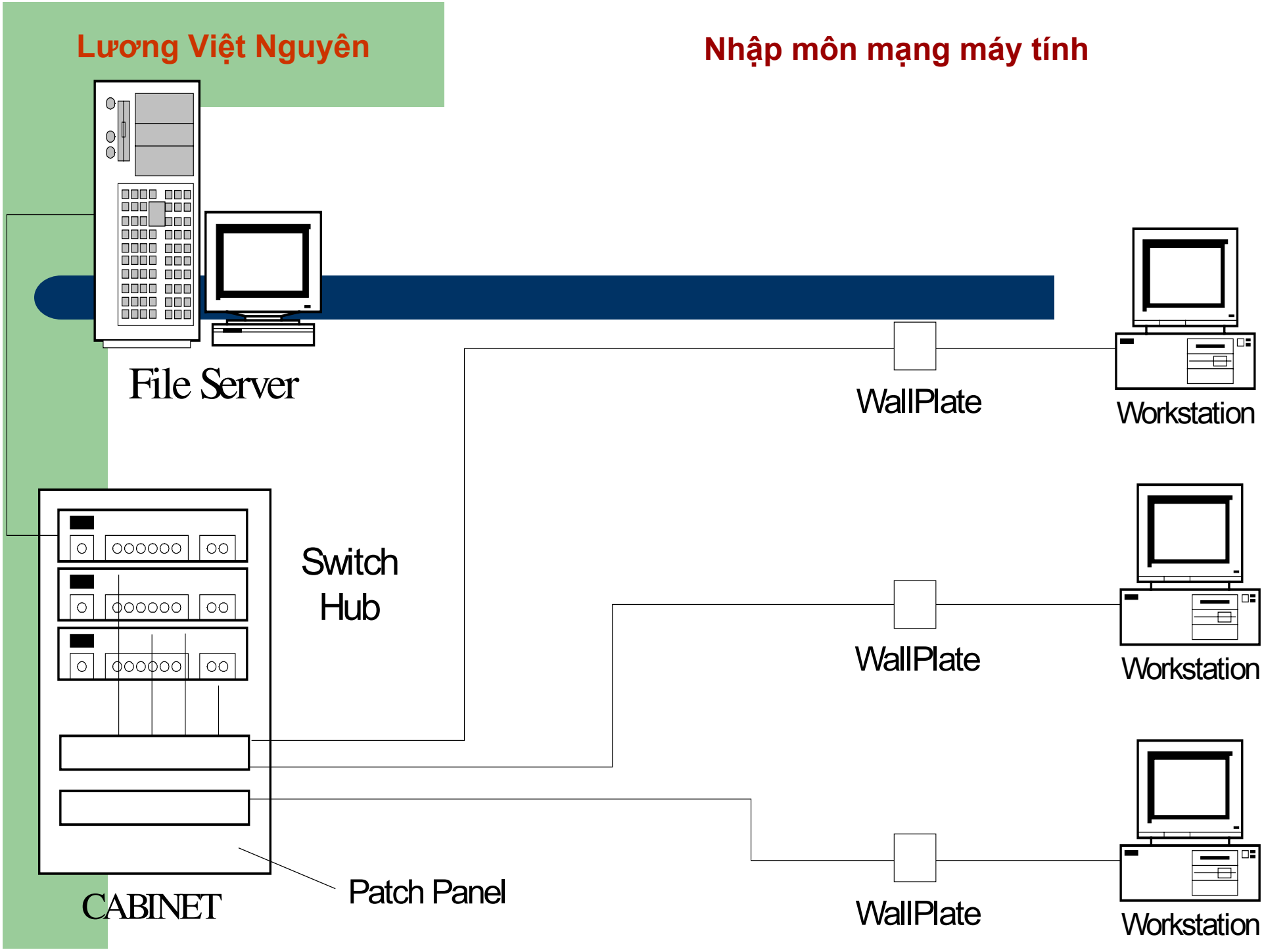
VIII. Ví dụ mạng


Backup System
Proxy
Proxy Server
Database Server
Proxy Server
File Server
LAN BACKBONE GIGABIT VP TCT HÀ NỘI
Workstation
Printer
Laptop
Workstation
Router
Modem
NTU
Modem
Dial-up
Lease-line
PSTN
LAN 10/100 BASE-TX VP TCT HÀ NỘI
WorkStation
Notebook
REMOTE USER HOẶC VP/CN CHƯA CÓ LAN
LAN 10/100 BASE-TX VPCT/CN LỚN
LAN 10/100 BASE-TX VPCT/CN NHỎ
ISP/Internet

Lương Việt Nguyên

Nhập môn mạng máy tính




File Server
WallPlate
Workstation
Switch
Hub
WallPlate
Workstation
CABINET
Patch Panel
WallPlate
Workstation

Bài 2: Các loại mạng chủ yếu

- Nói chung, tất cả các mạng máy tính đều có chung một số thành phần, chức năng, và đặc tính nhất định. Đó là:
  - Máy phục vụ (Server) - Máy cung cấp tài nguyên chung cho người dùng mạng
  - Máy khách (Client) - Máy truy cập tài nguyên mạng dùng chung do máy phục vụ cung cấp
  - Phương tiện truyền dẫn (media) - Cách thức và vật liệu nối máy tính
  - Dữ liệu dùng chung (shared data) - Các tập tin do máy phục vụ cung cấp ngang qua mạng
  - Máy in và các thiết bị ngoại vi - Các tài nguyên khác do máy phục vụ cung cấp
  - Tài nguyên (resource) - Tập tin, máy in, hoặc những thành phần khác mà người dùng mạng sử dụng
- Mặc dù những điểm tương đồng trên, mạng máy tính vẫn được chia làm hai loại rõ rệt:
- Ngang hàng và Dựa trên máy phục vụ

I. Mạng ngang hàng:

- Trong hệ thống mạng ngang hàng, không có bất kỳ máy phục vụ chuyên dụng nào. Mọi máy tính trong hệ thống mạng đều bình đẳng và có vai trò như nhau.
- Vì mỗi máy đều hoạt động với vai trò vừa là máy chủ, vừa là máy phục vụ. Người dùng trên tự quyết định tài nguyên nào sẽ được dùng chung trên mạng.

1. **Quy mô:** Mạng ngang hàng còn được gọi là nhóm làm việc. Mỗi nhóm có khoảng 8 - 10 máy tính

I. Mạng ngang hàng:

2. Phí tổn

- Mạng ngang hàng tương đối đơn giản. Vì mỗi máy tính kiêm cả hai chức năng phục vụ và máy khách, nên không cần phải có máy chủ trung tâm thật mạnh. Mạng ngang hàng có thể rẻ tiền hơn mạng dựa trên máy phục vụ.
- Nên dùng mạng ngang hàng khi:
  - Có dưới 10 người dùng
  - Mọi người dùng đều trong một khu vực
  - Tính bảo mật không là yêu cầu bắt buộc

I. Mạng ngang hàng:

3. Hệ điều hành ngang hàng

- Phần mềm hệ điều hành mạng không nhất thiết phải có khả năng thi hành và tính bảo mật tương xứng với phần mềm điều hành cho máy phục vụ chuyên dụng.
- Chỉ cần sử dụng những hệ điều hành đơn giản như: MS Windows NT Workstations, MS Windows for Workgroups, MS Windows 95 để thích hợp cho mô hình mạng ngang hàng.
- Không cần phải có thêm phần mềm nào khác để thiết lập mạng.

II. Mạng dựa trên máy phục vụ

- Nếu môi trường có nhiều người sử dụng (trên 10 máy), mạng ngang hàng chắc chắn sẽ không thoả đáng. Vì thế, hầu hết các mạng đều có máy chủ phục vụ chuyên dụng. Máy phục vụ chỉ hoạt động như một máy phục vụ chứ không là máy khách như máy trạm làm việc.
- Tuy nhiên với sự phát triển về quy mô và lưu lượng thông tin trên mạng, một mạng máy tính yêu cầu phải có nhiều máy chủ. Phân phối tác vụ giữa các máy chủ để đạt được hiệu quả công việc cao nhất.

II. Mạng dựa trên máy phục vụ

1. Máy phục vụ chuyên dụng

- Máy phục vụ dành cho các mạng lớn được chuyên môn hoá nhằm đáp ứng trọn vẹn nhu cầu của người dùng. Ví dụ, mạng Windows NT có nhiều loại máy phục vụ khác nhau như:
  - Máy phục vụ tập tin/in ấn (file/print server)
  - Máy phục vụ chương trình ứng dụng (application server)
  - Máy phục vụ thư tín (mail server)
  - Máy phục vụ fax (fax server)
  - Máy phục vụ truyền thông (communication server)

II. Mạng dựa trên máy phục vụ

2. Vai trò của phần mềm

- Một máy phục vụ mạng và hệ điều hành mạng phối hợp với nhau như một đơn vị. Cho dù là mạnh mẽ tới đâu chăng nữa, nếu máy chủ không có được một hệ điều hành có khả năng vận dụng tối đa tài nguyên vật lý của nó.
- Hiện nay, có nhiều hệ điều hành mạng được sử dụng để đáp ứng nhu cầu công việc khác nhau như:
  - UNIX
  - Linux
  - Windows NT, Window 2000 family

II. Mạng dựa trên máy phục vụ

3. Ưu điểm của mạng dựa trên máy phục vụ

- Dùng chung tài nguyên: Máy chủ được thiết kế để cung cấp khả năng truy cập nhiều tập tin và máy in, đồng thời duy trì hiệu suất thi hành và sự an toàn cho người dùng. Tài nguyên trên máy chủ phục vụ thường được lắp đặt tập trung nên dễ tìm kiếm và truy xuất hơn là tài nguyên được đặt nằm rải rác ở các máy.
- An toàn và bảo mật: Giải pháp mạng dựa trên máy chủ phục vụ chiếm ưu thế hơn trong các vấn đề về an toàn và bảo mật. Trong một hệ điều hành mạng, người quản trị thường đặt ra các chính sách và áp chính sách đó cho từng người dùng trên mạng.

II. Mạng dựa trên máy phục vụ

- Sao lưu: Do những dữ liệu quan trọng có thể phải đặt tập trung lên một hoặc hai máy chủ để đảm bảo cho dữ liệu được an toàn tuyệt đối.
- Sự dư thừa: Thông qua hệ thống dư thừa dữ liệu, bất cứ dữ liệu nào cũng được sao chép và lưu trữ trên mạng, sao cho vẫn có thể phục hồi lại dữ liệu ban đầu từ các vùng bản sao dữ liệu đó.
- Các yêu cầu về phần cứng: Phần cứng của máy khách thường nhỏ, chỉ đủ cho người dùng. Nhưng phần cứng cho máy chủ phục vụ phải yêu cầu cao hơn, tuỳ thuộc vào mục đích sử dụng của máy chủ phục vụ.

II. Mạng dựa trên máy phục vụ

4. Mạng kết hợp

- Việc kết hợp hai loại mạng trên với nhau để lợi dụng được các đặc tính ưu việt của cả hai loại mạng không có gì lạ. Trong mạng kết hợp, các hệ điều hành hoạt động phối hợp với nhau nhằm tạo cảm giác về một hệ thống hoàn chỉnh.
- Hệ điều hành mạng dựa trên máy chủ phục vụ: như MS Windows NT Server hoặc Novell, NetWare. Hệ điều hành máy khách có thể là MS Windows NT Workstation, MS Windows 98.
- Loại mạng này tuy phổ biến, nhưng nó đòi hỏi nhiều công sức và thời gian hoạch định và đào tạo, để bảo đảm sự thi hành đúng đắn và mức độ an toàn tốt.

III. Cấu hình mạng

- Cấu hình mạng là việc sắp xếp, bố trí vật lý của máy tính, dây cáp, và các thành phần khác trên mạng theo phương diện vật lý. Cấu hình mạng ảnh hưởng đến các khả năng của mạng. Một cấu hình mạng có thể ảnh hưởng đến:
  - Loại thiết bị mạng cần
  - Các khả năng của thiết bị
  - Sự phát triển của mạng
  - Cách thức quản lý mạng

III. Cấu hình mạng

- Cấu hình mạng hay cách xếp đặt các máy tính trong mạng phụ thuộc vào card mạng, dây cáp mạng, hệ điều hành mạng và các thành phần phụ trợ khác.
- Một cấu hình mạng không chỉ quyết định loại cáp sử dụng mà còn quyết định phải đi cáp qua môi trường thực tế như thế nào (trần nhà, sàn nhà, tường). Thậm chí nó còn quyết định đến giao thức giao tiếp giữa các máy tính trong mạng. Cấu hình khác nhau sẽ đòi hỏi phương pháp giao tiếp khác nhau.

Bài 3: Phương tiện vật lý cho việc thiết kế mạng

I. Cáp mạng

Các loại cáp chính:

- Ngày nay, phần lớn các mạng được nối bằng dân dẫn hoặc cáp, dây dẫn và cáp đóng vai trò như phương tiện truyền tín hiệu giữa các máy tính trên mạng.
- Có 3 nhóm cáp chính:
  - Cáp đồng trục (coaxial)
  - Cáp xoắn đôi (twisted-pair)
    - Cáp xoắn đôi trần
    - Cáp xoắn đôi có bọc
  - Cáp sơi quang (fiber-optic)

I. Cáp mạng

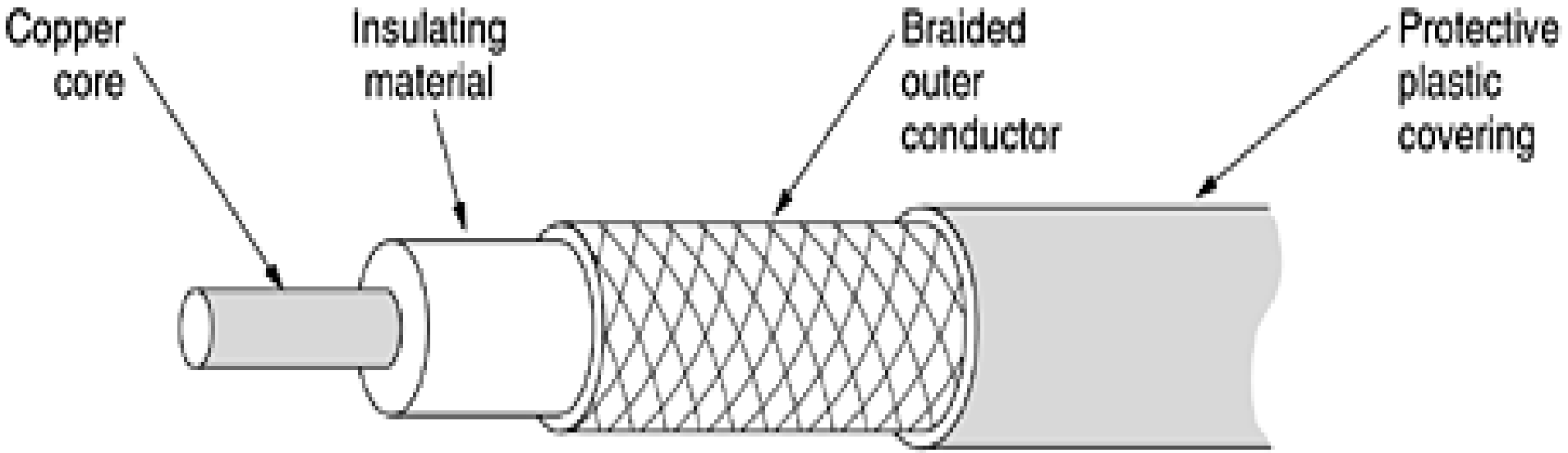
1. Cáp đồng trục

- Cáp đồng trục gồm một lõi đồng nguyên chất được bọc chất cách ly, một lớp bảo vệ bằng lưới kim loại và một lớp vỏ bọc ngoài.
- Lớp bảo vệ là tấm lưới kim loại bọc quanh một số loại cáp, có tác dụng hút tín hiệu điện tử chạy lạc không cho ảnh hưởng đến tín hiệu dữ liệu được truyền trên dây cáp.
- Lõi đồng trục mang tín hiệu điện tử tạo thành dữ liệu. Đây là lõi đặc hoặc lõi có dạng bện.
- Bao quanh lõi là một lớp cách ly, ngăn cách lõi với lưới kim loại.

I. Cáp mạng

Các loại cáp đồng trục:

- ***Loại cáp mảnh (thinnet)***
  - Có đường kính khoảng 0.5cm.
  - Mang tín hiệu đi xa tới 185m
- ***Loại cáp dày (thicknet)***
  - Có đường kính khoảng 1.3cm
  - Mang tín hiệu đi xa tới 500m


Copper core
Insulating material
Braided outer conductor
Protective plastic covering

I. Cáp mạng

2. Cáp xoắn đôi

- Gồm hai sợi dây đồng cách ly quấn vào nhau, có hai loại: Cáp xoán đôi trần (UTP) và Cáp xoắn đôi có bọc (STP).
- Cáp xoắn đôi sử dụng bộ nối điện thoại RJ-45 để nối tới máy tính

I. Cáp mạng

3. Cáp sợi quang

- Trong cáp sợi quang, sợi quang truyền tín hiệu dữ liệu dạng số ở hình thái xung ánh sáng điều biến. Cáp sợi quang truyền khối lượng dữ liệu với vận tốc rất cao do tín hiệu không bị suy yếu trong quá trình truyền và do độ trong sạch (không bị nhiễu) của tín hiệu.
- Sợi quang gồm một sợi thuỷ tinh cực mảnh, gọi là lõi, được bao quanh bởi một lớp thuỷ tinh đông tâm gọi là lớp vở bọc.

I. Cáp mạng

4. Chọn kiểu đi cáp

- Muốn xác định kiểu đi cáp nào thích hợp nhất cho một địa điểm, người thiết kế đường mạng phải quan tâm đến các vấn đề sau:
  - Lưu lượng truyền trên mạng có nhiều không?
  - Yêu cầu an toàn mạng là gì?
  - Khoảng cách mà mạng phải kéo tới là bao nhiêu?
  - Các chọn lựa cáp là gì?
  - Tiền kéo cáp là bao nhiêu?

I. Cáp mạng

- Những yêu tô phải quan tâm đên:
  - Tính hợp lý: Cáp có dễ lắp đặt không? Nếu lắp đặt mạng ở phạm vi hẹp, và độ bảo mật không thành vấn đề, không cần phải chọn cáp dày, cồng kềnh và đắt tiền.
  - Vỏ bọc bảo vệ: Nếu môi trường có nhiều nhiễu điện thì đường cáp cần có vỏ bọc bảo vệ nhiễu điện.
  - Tốc độ truyền: Tuỳ thuộc vào nhu cầu cần thiết mà người thiết kế mạng lựa chọn loại cáp nào để thi công. Nói chung các loại cáp đồng thường có tốc độ chậm khoảng 10Mbps đến 100Mbps.
  - Phí tổn: Sự chọn lựa loại cáp tốt, tốc độ truyền cao thường làm cho phí tổn rất lớn.
  - Sự suy yếu: Sự suy yếu tín hiệu thường xảy ra khi đường đi cáp quá dài, máy nhận sẽ khổng hiểu được tín hiệu từ máy truyền tới. Trong trường hợp đó, ta phải thiết lập các hệ thống kích tín hiệu và kiểm tra lỗi.

II. Các thiết bị mạng

1. NIC – Network Interface Card

- Là thiết bị phổ dụng nhất đối với máy tính. Trong NIC có bộ thu phát (Tranceiver) hoạt động như một Transmitter và một Receiver. Transmitter chuyển đổi các tín hiệu bên trong máy tính thành tín hiệu có thể truyền đi được qua đường mạng. Receiver làm ngược lại.

II. Các thiết bị mạng

2. Hup

2.1 HUB bị động (HUB – Passive)

- Không chứa các lịnh kiện điện tử các xử lý tín hiệu, chỉ có chức năng tổ hợp các tin hiệu từ một số các đoạn mạng. Khoảng cách lớn nhất giữa một máy tính với hub không thể lớn hơn một nửa khoảng cách cho phép giữa 2 máy tính.

2.2 HUB chủ động (HUB – Active)

- Có các linh kiện điện tử có thể khuyếch đại và xử lý tín hiệu. Cho phép khoảng cách giữa các thiết bị tăng lên.

II. Các thiết bị mạng

2.3 HUB thông minh (Intelligent Hub)

- Là hub chủ động nhưng có thêm các chức năng mới sau:
  - Quản trị hub: được bổ sung các giao thức quản trị mạng, cho phép hub gửi các thông tin về trạm điều khiển mạng trung tâm. Và cho phép trạm trung tâm quản lý hub.
  - Chuyển mạch: chứa các vi mạch cho phép chọn đường nhanh cho các tín hiệu giữa các cổng trên hub. Thay vì chuyển gói tin cho toàn bộ các cổng của hub, chúng đang thay thế dần cho các bridge và router.

II. Các thiết bị mạng

3. Repeater (Bộ chuyển tiếp)

- Có chức năng tiếp nhận và chuyển tiếp các tín hiệu dữ liệu, thường được dùng nối 2 đoạn cáp mạng Ethernet để mở rộng mạng. Có khả năng khuếch đại và tái sinh tín hiệu.

II. Các thiết bị mạng

4. Bridge (Cầu)

- Là một thiết bị mềm dẻo hơn repeater. Một repeater chuyển đi tất cả các tín hiệu mà nó nhận được. Nhưng Bridge có chọn lọc và chỉ chuyển đi các tín hiệu có đích ở phần mạng phía bên kia.

II. Các thiết bị mạng

5. Multiplexor (bộ dồn kênh)

- Là thiết bị có chức năng tổ hợp một số tín hiệu để chúng có thể truyền được với nhau và sao đó khi nhận, lại được tách ra trở lại các tín hiệu gốc.

II. Các thiết bị mạng

6. Modem (Modulation/Demodulation)

- Là thiết bị có chức năng chuyển đổi tín hiệu thành tín hiệu tương tự và ngược lại, để kết nối các máy tính qua đường điện thoại.
- Cho phép trao đổi thư điện tử, truyền tệp, truyền fax và trao đổi dữ liệu theo yêu cầu.

II. Các thiết bị mạng

7. Router (Bộ chọn đường)

- Là thiết bị thông minh Bridge vì có còn thực hiện các giải thuật chọn đường đi tối ưu cho các gói tin. Bridge hoạt động ở hai tầng Physical và Datalink, trong khi router có thể hoạt động lên tới tầng 3 (Network).
- Cho phép kết nối nhiều mạng với nhau tạo thành liên mạng.

II. Card mạng

Vai trò của card mạng

- Card mạng đóng vai trò như giao diện hoặc kết nối vật lý giữa máy tính và cáp mạng. Có những vai trò sau:
  - Chuẩn bị dữ liệu cho cáp mạng
  - Gửi dữ liệu đến máy tính khác
  - Kiểm soát luồng dữ liệu giữa máy tính và hệ thống cáp

II. Card mạng

1. Chuẩn bị dữ liệu:

- Dữ liệu trước khi truyền đi phải được card mạng chuyển đổi từ dạng mà máy tính có thể hiểu được sang dạng tín hiệu mà có thể gửi được qua cáp mạng. Trên cáp mạng, dữ liệu phải đi theo một luồng bit đơn lẻ. Khi chúng đi trên cáp mạng, các bit được truyền đi nối đuôi nhau, dữ liệu chạy trên cáp chỉ theo một hướng.
- Điều này có nghĩa là: tại mỗi thời điểm, máy tính chỉ có thể hoặc đang nhận dữ liệu, hoặc đang gửi dữ liệu.

II. Card mạng

2. Địa chỉ mạng:

- Bên cạnh việc biến đổi dữ liệu, card mạng còn phải cho biết địa chỉ của nó để phân biệt với các card mạng khác trong mạng. Việc định địa chỉ cho card mạng cho viện công nghệ điện và điện tử (IEEE – Institude of Electrical and Electrics Engineers) quyết định.
- Việc này cung cấp cho mỗi hãng sản xuất một địa chỉ, các hãng sản xuất sẽ nối thêm mã để tích hợp vào từng card mạng. Vì thế tất cả các card mạng trên thế giới đều có địa chỉ khác nhau.

II. Card mạng

3. Gửi và kiểm soát dữ liệu:

- Trước khi gửi dữ liệu, hai card mạng ở hai máy tính đều phải thống nhất với nhau cách thức truyền dữ liệu như: kích thước cụm dữ liệu, lượng dữ liệu được gửi đi, thời gian chờ ngắt quãng giữa các cụm dữ liệu.

II. Card mạng

4. Khả năng tương thích của card mạng

- Card mạng là một modun được gắn với máy tính, vì thế để máy tính và card mạng có thể làm việc được với nhau, card mạng phải:
  - Vừa vặn với cấu trúc bên trong của máy tính
  - Có bộ nối cáp thích hợp với hệ thống cáp

Mạng máy tính
&
Hệ thống thông tin công nghiệ

Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Chương 1: Các khái niệm cơ bản trong hệ đo & điều khiển Công nghiệp.

1.1 Chức năng của một hệ thông đo và điều khiển trong CN
- 1.1.1 Chức năng điều khiển các quá trình.
- 1.1.2 Chức năng điều khiển logic, liên động cảnh báo.
- 1.1.3 Chức năng giao tiếp người và hệ thống.
- 1.1.4 Chức năng thu thập và quản lý thông tin

1.2 Các khái niệm về các hệ và các thiết bị hiện đại trong CN:
- 1.2.1 Smart Device.
- 1.2.2 PLC (Programmable Logic Controller).
- 1.2.3 SCADA (Supervisory Control And Data Acquisition)
- 1.2.4 DCS(Distributed Control System)

Chức năng điều khiển các quá trình

- Điều khiển các quá trình nhiệt độ, áp suất, lưu lượng..Theo một hàm thời gian.
- Các bộ điều khiển tương tự 4-20 mA.
- Các bộ điều khiển lai, số.
- DCS

Chức năng điều khiển Logic, liên động, cảnh báo

- Điều khiển logic, liên động các thiết bị , điều khiển tuần tự, cảnh báo.
- Relay cơ điện, Timer, Counter.
- IC số.
- PLC.

Chức năng giao tiếp giữa người và hệ thống

- Người vận hành có thể theo dõi quá trình, điều khiển quá trình, thay đổi Setpoint..
- Panel điều khiển, công tắc, nút ấn, chiết áp.
- Đèn báo, đồng hồ (Analog, digital)
- Giao diện bằng máy tính dựa trên phần mềm HMI

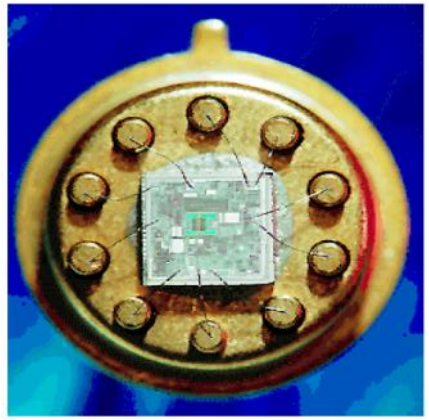
Chức năng thu thập và quản lý thông tin

- Đo, thu thập và quản lý thông tin.
- bằng tay (thủ công).
- Các đồng hồ tự ghi, relay tự rơi, recorder.
- Sử dụng máy tính.

Smart Device

- Thiết bị số trên cơ sở uP có khả năng xử lý thông tin.
- Hiển thị tại chỗ hay từ xa.
- Mềm dẻo, kinh tế.
- Có khả năng tự động kiểm tra, chuẩn đoán.

Smart Device



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

PLC (Programmable Logic Controller)

Thiết bị điều khiển khả trình (PLC, programmable logic controller) là một loại máy tính điều khiển chuyên dụng, do nhà phát minh người Mỹ Richard Morley lần đầu tiên đưa ra ý tưởng vào năm 1968. Dựa trên yêu cầu kỹ thuật của General Motors là xây dựng một thiết bị có khả năng lập trình mềm dẻo thay thế cho mạch điều khiển logic cứng, hai công ty độc lập là Allen Bradley và Bedford Associates (sau này là Modicon) đã đưa ra trình bày các sản phẩm đầu tiên. Các thiết bị này chỉ xử lý được một tập lệnh logic cơ bản, 128 điểm vào/ra (1 bit) và 1kByte bộ nhớ.

PLC (Programmable Logic Controller)

- Thiết bị số trên cơ sở uP
- Phát triển để thay thế cho Relay,Timer....
- Sử dụng để điều khiển quá trình, liên động với các đầu I/O số.
- Chương trình viết bằng ngôn ngữ Ladder Logic.

PLC (Programmable Logic Controller)

MCS1200 Controller

22x CPUs

21x CPUs

S7-200 Quick Reference Information

Intelligent PI/O Modules

SCADA (Supervisory Control And Data Acquisition)

Điều khiển giám sát (và thu thập dữ liệu)

-Hỗ trợ con người trong việc quan sát và điều khiển từ xa

- Có giao diện người-máy
- HMI (Human-Machine Interface) - Giao diện người-máy
  - + Thành phần trong một hệ SCADA, hoặc
  - + Các phương tiện quan sát/thao tác ở cấp thấp hơn
- Các trạm điều khiển giám sát trung tâm
  - + Engineering Station (ES)
  - + Operator Station (OS)
  - + Server Station (SS)

SCADA (Supervisory Control And Data Acquisition)

- Các trạm thu thập dữ liệu trung gian
  + Remote Terminal Unit (RTU)
  + Data Collection Unit (DCU): PLC, PC, I/O
- Hệ thống truyền thông
  + Mạng truyền thông công nghiệp
  + Mạng viễn thông/truyền dữ liệu đường dài (vô tuyến, hữu tuyến)
  + Các thiết bị chuyển đổi, dồn kênh (Modem, Multiplexer)

SCADA (Supervisory Control And Data Acquisition)

- Các công cụ phát triển ứng dụng

*Giao diện người-máy (HMI)

+Sơ đồ hệ thống, sơ đồ công nghệ
+ Hiển thị các biến quá trình qua các "thiết bị ảo"
+ Đồ thị thời gian thực, đồ thị dữ liệu tĩnh
+ Các phím thao tác, nút điều khiển (controls)

* Hỗ trợ trao đổi tin tức (Messaging), xử lý sự kiện (Event), sự cố (Alarm)

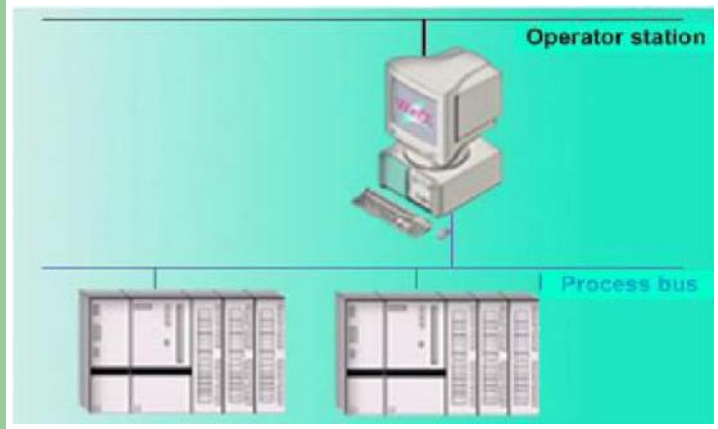
* Hỗ trợ việc thống kê và lập báo cáo (Reporting)

SCADA (Supervisory Control And Data Acquisition)

* Phần mềm kết nối với các nguồn dữ liệu (drivers cho các PLC, các module vào/ra, cho các hệ thống bus trường)

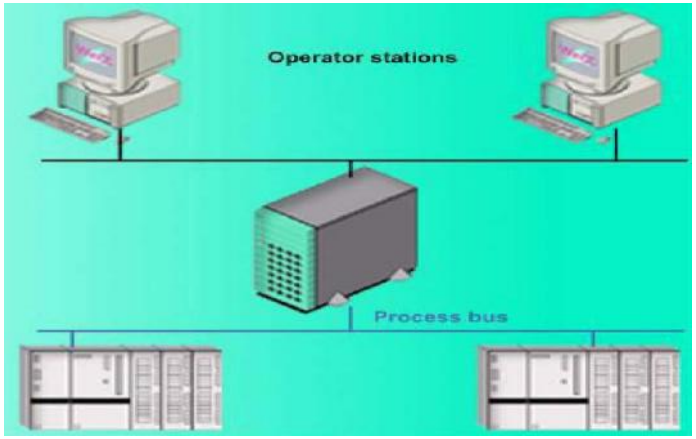
* Cơ sở dữ liệu quá trình, dữ liệu cấu hình hệ thống

Hệ một người dùng



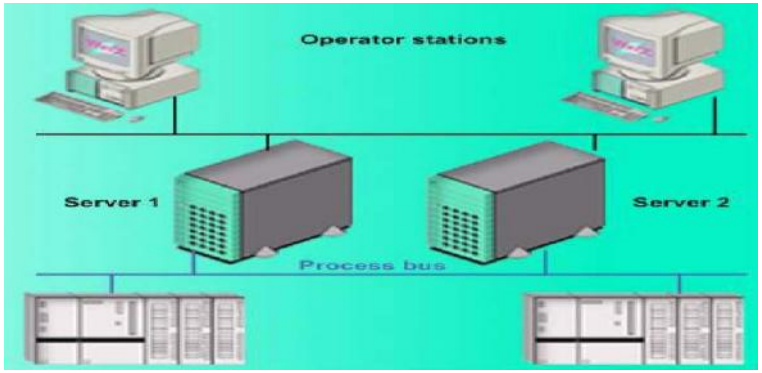
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Hệ nhiều người dùng

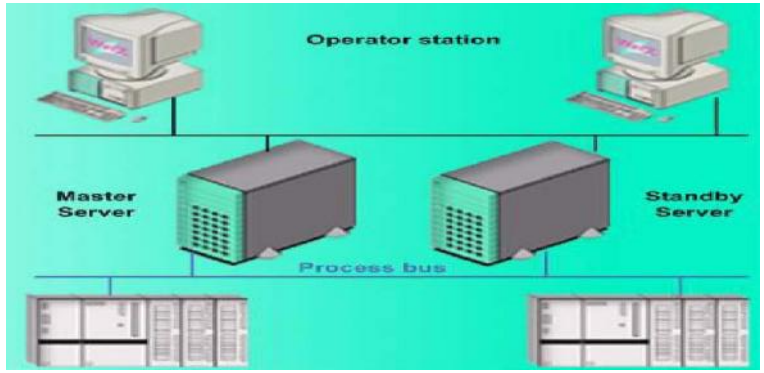


Hệ Web Client

Hệ phân tán



Hệ chạy dự phòng



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

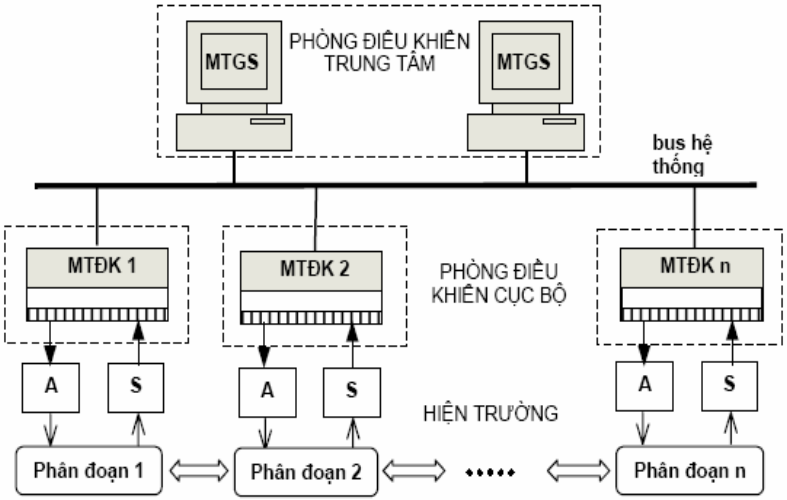
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Communication

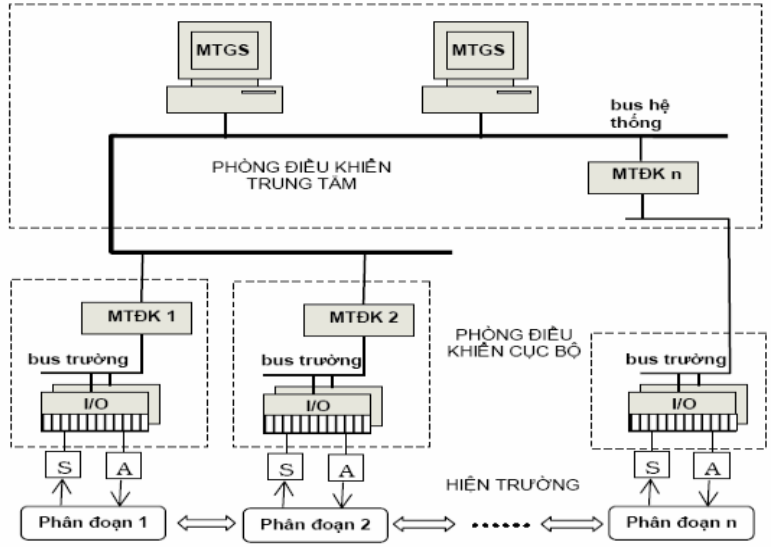
DCS (Distributed Control System)

- Các thiết bị điều khiển số + Phần cứng phần mềm thu thập thông tin.
- Đường truyền tốc độ cao.
- Các module bố trí phân tán.
- Mỗi Module thực hiện một chức năng riêng.
- Có giao diện để nối các máy tính điều khiển giám sát và các bộ điều khiển.

DCS (Distributed Control System)

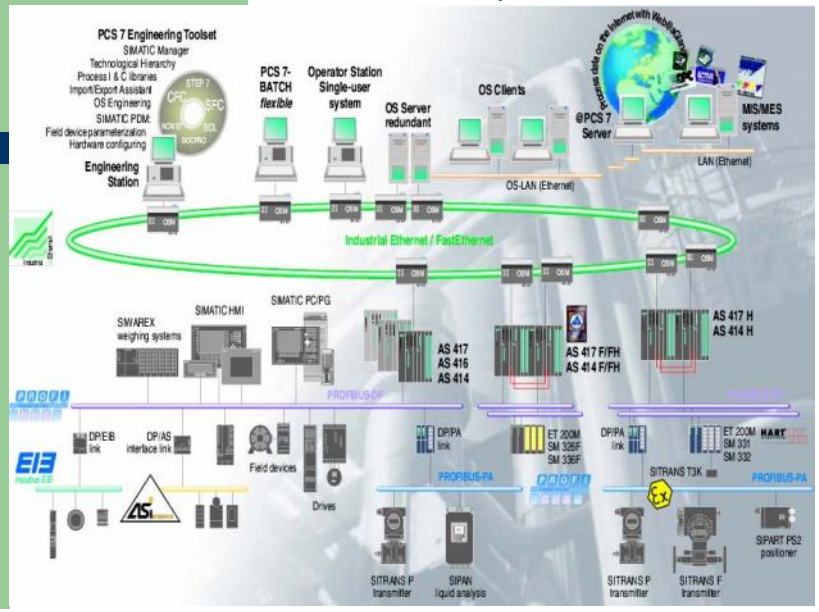


DCS (Distributed Control System)

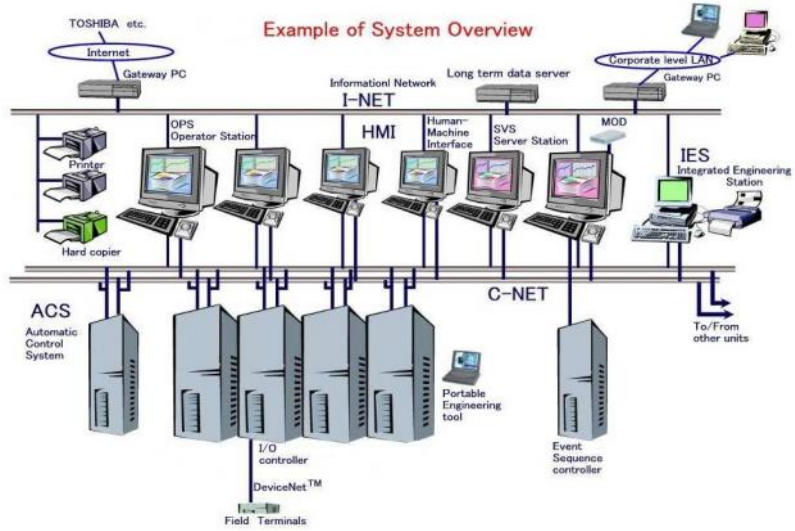


Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

DCS (Distributed Control System)



DCS (Distributed Control System)



DCS (Distributed Control System)

DCS Truyền thống.

- Các hệ này sử dụng các bộ điều khiển quá trình đặc chủng theo kiến trúc riêng của nhà sản xuất.
- Các hệ cũ thường đóng kín, ít tuân theo các chuẩn giao tiếp công nghiệp, các bộ điều khiển được sử dụng cũng thường chỉ làm nhiệm vụ điều khiển quá trình, vì vậy phải sử dụng kết hợp PLC cho các bài toán điều khiển logic và điều khiển trình tự.
- Các hệ mới có tính năng mở tốt hơn, một số bộ điều khiển lai đảm nhiệm cả các chức năng điều khiển quá trình, điều khiển trình tự và điều khiển logic (hybrid controller).

DCS (Distributed Control System)

- Để hỗ trợ các bài toán điều khiển quá trình diễn ra đồng thời, khối xử lý trung tâm được cài đặt một hệ điều hành thời gian thực, đa nhiệm - hoặc của riêng nhà sản xuất phát triển hoặc một sản phẩm thông dụng như pSOS, TSOS, VRTX,... Chu kỳ thời gian nhỏ nhất thực hiện các mạch vòng điều khiển thường nằm trong khoảng 10-100ms, trong trường hợp đặc biệt (ví dụ cho nhà máy điện) có thể tới 1ms.

DCS (Distributed Control System)

- Một số sản phẩm tiêu biểu cùng với tên trạm điều khiển cục bộ được liệt kê dưới đây:

• AdvantOCS (ABB): Advant Controller, hệ điều hành riêng

• Freelance 2000 (ABB): D-PS hợc D-FC, hệ điều hành pSOS

• DeltaV (Fisher-Rosermount): Visual Controller, hệ điều hành TSOS

• PlantScape (Honeywell): PlantScape Controller, hệ điều hành riêng

•Centum CS1000/CS3000 (Yokogawa): PFCx-E, AFS10x/AFS20x, hệ điều hành ORKID

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

DCS (Distributed Control System)

DSC trên nền PLC

Một số hệ DCS trên nền PLC tiêu biểu là SattLine (ABB), Process Logix (Rockwell), Modicon TSX (Schneider Electric), PCS7 (Siemens),… Thực chất, ngày nay đa số các PLC vừa có thể sử dụng cho bài toán điều khiển logic và điều khiển quá trình. Tuy nhiên, các PLC được sử dụng trong các hệ điều khiển phân tán thường có cấu hình mạnh, hỗ trợ điều khiển trình tự cùng với các phương pháp lập trình hiện đại (ví dụ SFC).

DCS (Distributed Control System)

DSC trên nền PC

Giải pháp sử dụng máy tính cá nhân (PC) trực tiếp làm thiết bị điều khiển không những được bàn tới rộng rãi, mà đã trở thành thực tế phổ biến trong những năm gần đây. Nếu so sánh với các bộ điều khiển khả trình (PLC) và các bộ điều khiển DCS đặc chủng thì thế mạnh của PC không những nằm ở tính năng mở, khả năng lập trình tự do, hiệu năng tính toán cao và đa chức năng, mà còn ở khía cạnh kinh tế. Các bước tiến lớn trong kỹ thuật máy tính, công nghiệp phần mềm và công nghệ bus trường chính là các yếu tố thúc đẩy khả năng cạnh tranh của PC trong điều khiển công nghiệp.

DCS (Distributed Control System)

DCS trên nền PC là một hướng giải pháp tương đối mới, mới có một số sản phẩm trên thị trường như PCS7 (Siemens, giải pháp Slot-PLC), 4Control (Softing), Stardom (Yokogawa), Ovation (Westinghouse-Emerson Process Management)… Hướng giải pháp này thể hiện nhiều ưu điểm về mặt giá thành, hiệu năng tính toán và tính năng mở. Một trạm điều khiển cục bộ chính là một máy tính cá nhân công nghiệp được cài đặt một hệ điều hành thời gian thực và các card giao diện bus trường và card giao diện bus hệ thống.

Mạng máy tính
&
Hệ thống thông tin công nghiệ

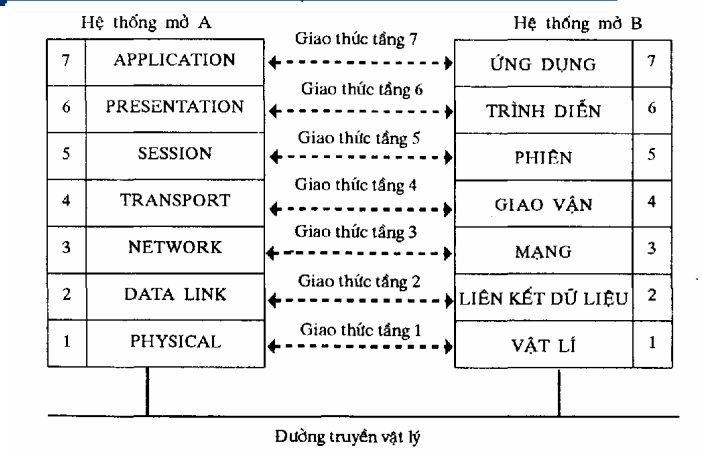
Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Thế nào là hệ kín, hệ mở,mô tả chung về OSJ

- Mạng mà chỉ có thiết bị của một nhà sản xuất và nó làm việc với phần cứng và giao thức đặc biệt gọi là hệ kín.
- Hệ mở được thiết kế theo nguyên tắc mở cho tất cả, nó cho phép tất cả các thiết bị phù hợp với chuẩn thì có thể tham gia vào mạng chuẩn

Thế nào là hệ kín, hệ mở,mô tả chung về OSJ



Thế nào là hệ kín, hệ mở,mô tả chung về OSJ

Mô hình hệ mở có thể coi như là tập hợp của các thực thể (chương trình phần mềm và mạch phần cứng) đặt trong các lớp. Các thực thể trong cùng một lớp nhưng ở các nút khác nhau gọi là thực thể tương đương. Các thực thể chỉ có thể thông tin với thự thể tương đương trong hệ khác. Mạch dù các thực thể tương đương có thể thông tin với nhau nhưng số liệu thực sự cần phải đi qua các lớp bên dưới trong hệ truyền và đi theo chiều ngược lại trong hệ nhận.

Lớp vật lý

Theo định nghĩa của ISO, tầng vật lý định nghĩa các thông số điện, cơ, các chức năng, thủ tục để kích hoạt, duy trì và đình chỉ liên kết vậy lý giữa các hệ thống. Thuộc tính điện, cơ:

- Cấu trúc mạng (dạng vòng, dạng sao, dạng Bus....).
- Định nghĩa khía cạnh dòng điện, điện áp biểu diễn các bit.
- Kỹ thuật điều chế tín hiệu sử dụng trong mạng.
- Định nghĩa về mặt cơ học của kết nối (kiểu đầu nối, chân tín hiệu, kiểu cáp).

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Lớp liên kết dữ liệu

Tầng Liên kết dữ liệu cung cấp các phương tiện để truyền thông tin qua liên kết vật lý đảm bảo tin cậy thông qua các cơ chế đồng bộ hóa, kiểm soát lỗi và kiểm soát luồng, dữ liệu.

- Định nghĩa giao thức cần phải tuân theo cho việc truy nhập mạng để truyền và nhận bản tin.
- Chia một khối lớn thông tin ra thành các khung định dạng nhỏ hơn (Framming).
- Trả lời các thông tin đã nhận được (thông tin đồng bộ hoá, kiểm soát lỗi, kiểm soát luồng dữ liệu).

Lớp liên kết dữ liệu

* DLP *dị bộ*: Các DLP *dị bộ* sử dụng phương thức truyền dị bộ, trong đó các bit đặc biệt START và STOP được dùng để tách các xâu bit biểu diễn các kýu trong dòng dữ liệu cần truyền đi. Phương thức này được gọi là dị bộ là vì không cần có sự đồng bộ liên tục giữa người gửi và người nhận tin. Nó cho phép một ký tự dữ liệu được truyền đi bất kỳ lúc nào mà không cần quan tâm đến các tín hiệu đổng bộ trước đó.

Lớp liên kết dữ liệu

* DLP *đồng bộ*: Phương thức truyền đồng bộ không dùng các bit đặc biệt START, STOP để đóng khung mỗi ký tự mà chèn các ký tự đặc biệt như SYN (Synchronization), EOT (End Of Transmission) hay đơn giản hơn, một cái "cờ" (flag) giữa các dữ liệu của người sử dụng để báo hiệu cho người nhận biết được dữ liệu "đang đến" hoặc "đã đến". Cần lưu ý rằng các hệ thống truyền thông đòi hỏi hai mức đồng bộ hóa :

- ở mức vật lý : để giữ đồng bộ giữa các đồng hồ của người gửi và người nhận.

- ở mức liên kết dữ liệu : để phân biệt dữ liệu của người sử dụng với các "cờ" và các vùng thông tin điều khiển khác.

Lớp vận chuyển

Các nhiệm vụ cụ thể của lớp vận chuyển bao gồm:

- Quản lý về tên hình thức cho các trạm sử dụng
- Định vị các đối tác truyền thông qua tên hình thức và/hoặc địa chỉ
- Xử lý lỗi và kiểm soát dòng thông tin, trong đó có cả việc lập lại quan hệ liên kết và thực hiện các thủ tục gửi lại dữ liệu khi cần thiết
- Dồn kênh các nguồn dữ liệu khác nhau
- Đồng bộ hóa giữa các trạm đối tác.

Để thực hiện việc vận chuyển một cách hiệu quả, tin cậy, một dữ liệu cần chuyển đi có thể được chia thành nhiều đơn vị vận chuyển *(data segment unit)* có đánh số thứ tự kiểm soát trước khi bổ sung các thông tin kiểm soát lưu thông.

Lớp phiên

Tầng phiên (Session) là tầng thấp nhất trong nhóm các tầng cao và nằm ở ranh giới giữa hai nhóm tầng nói trên. Mục tiêu của nó là cung cấp cho người sử dụng cuối các chức năng cần thiết để quản trị các *phiên* ứng dụng của họ, cụ thể là :

- Điều phối việc trao đổi dữ liệu giữa các ứng dụng bằng cách thiết lập và giải phóng (một cách lôgic) các *phiên* (hay còn gọi là các hộp *hội thoại* - dialogues) .
- Cung cấp các điểm đồng bộ hóa để kiểm soát việc trao đổi dữ liệu.
- áp đặt các qui tắc cho các tương tác giữa các ứng dụng của người sử dụng.
- Cung cấp cơ chế "lấy lượt" (nắm quyền) trong quá trình trao đổi dữ liệu.

Lớp biểu diễn

Mục đích của tầng Trình diễn (Presentation layer) là đảm bảo cho các hệ thống cuối có thể truyền thông có kết quả ngay cả khi chúng sử dụng các biểu diễn dữ liệu khác nhau. Để đạt được điều đó nó cung cấp một biểu diễn chung để dùng trong truyền thông và cho phép chuyển đổi từ biểu diễn cục bộ sang biểu diễn chung đó.

Lớp ứng dụng

Bên cạnh truyền tải thông tin, lớp ứng dụng còn cung cấp các dịch vụ như sau:

- Nhận dạng các chủ thể tham gia thông tin qua tên và địa chỉ
- Xác định khả năng hiện hành của một chủ thể tham gia thông tin
- Thiết lập thẩm quyền thông tin
- Thống nhất cơ cấu bảo mật
- Cấp quyền cho các chủ thể tham gia thông tin
- Chọn qui tắc đàm thoại, bao gồm các thủ tục khởi tạo và xóa bỏ
- Thống nhất trách nhiệm khắc phục lỗi
- Xác nhận các ràng buộc trên cú pháp dữ liệu (các tập ký tự, các cấu trúc dữ liệu..)

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Mô hình hệ mở đơn giản

- Application.
- Data Link.
- Physical.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Mô hình hệ mở đơn giản

Khi giảm mô hình thì việc thực hiện sẽ có các giới hạn sau:

- Kích thước lớn nhất của bản tin sẽ bị giới hạn bởi kích thước kênh truyền (Transport).
- Không thể định tuyến bản tin giữa các mạng (Network).
- Chỉ cho phép thông tin Full-duplex (Session)
- Định dạng của bản tin phải như nhau ở mọi nút (Presentation).

Mạng máy tính
&
Hệ thống thông tin công nghiệ

Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Chương 2: Các giao thức công nghiệp

2.1 Khái niệm về giao thức:
- Giao thức ?
- Các đặc điểm quan trọng của giao thức.

2.2 Các yêu cầu riêng cho giao thức CN.

Chương 2: Các giao thức công nghiệp

2.3 ModBus:
- Mô tả chung về giao thức.
- Hai chế độ truyền ASCII và RTU.
- Khung bản tin của ModBus.
- Các phương pháp kiểm tra lỗi.
- Dữ liệu và các chức năng điều khiển.
- Các hàm phụ chuẩn đoán trong mạng.
- Báo lỗi.

Giao thức (Protocol)

- Giao thức thiết lập một tiêu chuẩn chung cho việc trao dữ liệu giữa phần thu và phát trên mạng.
- Nó thường kết hợp với gói tin.
- Giao thức điều khiển một khung bản tin chung cho tất cả các thiết bị trên mạng.
- Giao thức thiết lập hoạt động đúng cho hệ thông tin

Giao thức (Protocol)

Các đặc điểm của giao thức:

- Khởi động: Khởi động các thông số của giao thức để bắt đầu truyền số liệu qua kênh liên lạc.
- Tạo khung và đồng bộ khung.
- Điều khiển luồng dữ liệu.
- Điều khiển truy nhập đường truyền.
- Phát hiện và sửa lỗi.
- Kiểm soát Time-out.

Các yêu cầu riêng cho giao thức CN

- Đơn giản nhất có thể để dễ khắc phục sự cố:
  + CN là nơi có sự hiểu biết về mạng thông tin CN ít.
  + Đòi hỏi hoạt động liên tục.
  + Có ý thức lựa chọn giao thức đơn giản nhất có thể.
- Độ đảm bảo dữ liệu truyền cao:
  + Hoạt động trong môi trường có nhiễu điện lớn.
  + Các thiết bị công suất lớn tập trung với mật độ cao.
  + Đòi hỏi không có lỗi khi truyền.
  + Chọn giao thức có mức độ cao của việc kiểm tra lỗi.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các yêu cầu riêng cho giao thức CN

- Chuẩn hoá giao thức:
  + Có thể có nhu cầu cho việc kết nối giữa các thiết bị của các nhà SX khác nhau hay các hệ khác nhau.
  + Cần phải chuẩn hoá giao thức.
- Tốc độ cập nhật thông số cao:
  + Không đòi hỏi số lượng thông số lớn.
  + Yêu cầu cập nhật một loạt các setpoint cho một loạt các thiết bị gần như đồng thời.
  + Mộ số giao thức Field Bus mới có thể đáp ứng yêu cầu này.

Modbus

Mô tả chung về Modbus:

+ Modbus được phát triển bởi Modicon (AEG) cho hệ thống điều khiển các quá trình

ModBus

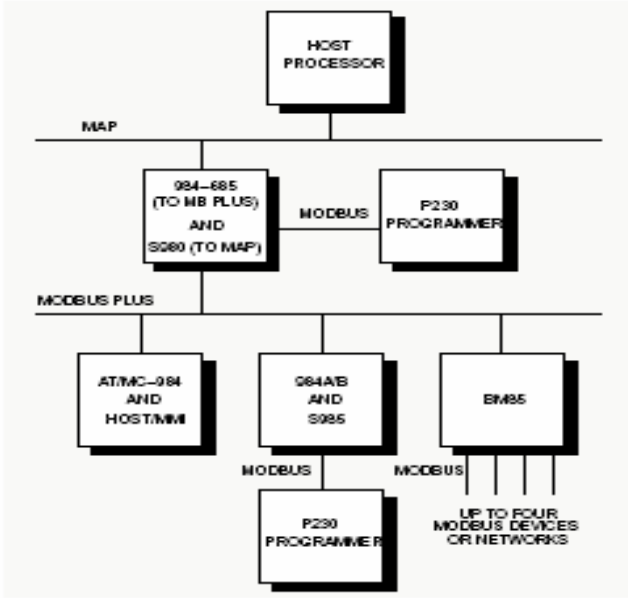


Figure 1 Overview of Modbus Protocol Application

ModBus

+ Modbus chuẩn của bộ điều khiển Modicon sử dụng cổng RS-232. Bộ điều khiển có thể nối mạng trực tiếp hay qua Modem.

+ Người dùng có thể lựa chọn các chuẩn RS-422, RS-485, 20mA Current loop, tất cả các chuẩn trên đều tương thích với tốc độ truyền của giao thức.

+ Thông tin giữa các bộ điều khiển sử dụng kỹ thuật Master-Slave. Chỉ có Master mới có quyền khởi động việc truyền dữ liệu, các thiết bị khác là Slave trả lời bằng cách cung cấp các dữ liệu được yêu cầu từ Master hoặc đáp lại các hoạt động.

+ Master có thể là các máy chủ, PC hay các Panel lập trình.

+ Slave là các bộ điều khiển có tối đa 247 Slave.

ModBus

+ Master có thể địa chỉ từng Slave riêng hay gửi một bản tin quảng bá tới tất cả các Slave.

+ Khi có yêu cầu bởi địa chỉ riêng thì sẽ có bản tin trả lời. Không có bản tin trả lời với yêu cầu quảng bá.

+ Modbus cung cấp một định dạng khung bản tin chung cho các bản tin truyền giữa Master và Slave. Bản tin bao gồm địa chỉ của thiết bị, mã chức năng định nghĩa các hoạt động yêu cầu, số liệu cần gửi và trường kiểm tra lỗi.

+ Slave trả lời bằng một bản tin nó chính là kết quả của hoạt động. Nếu có lỗi thì nó cũng báo lỗi nào đã xảy ra.

ModBus

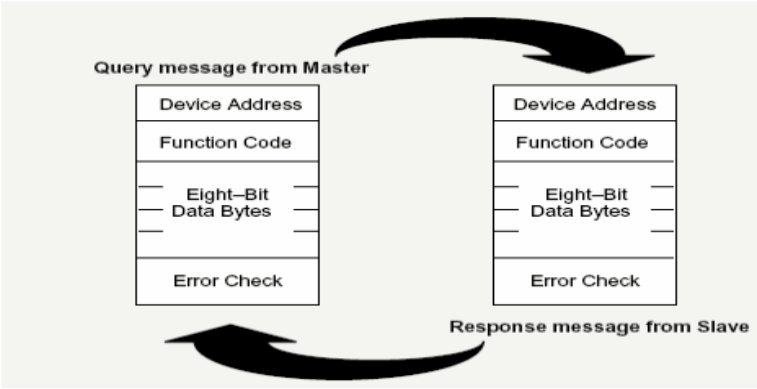
+ Ngoài ra các bộ điều khiển Modbus có thể thông tin trên Modbus Plus sử dụng cổng thông tin có sẵn hay cộng mạng và truyền trên MAP.

+ ở đay thông tin giữa các bộ điều khiển dùng kỹ thuật Pear-Pear.(ứng dụng vẫn là Master-Slave).

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus

The Query–Response Cycle



ModBus - Hai chế độ truyền

+ Bộ điều khiển trên mạng Modbus có thể truyền ở hai chế độ: ASCII và RTU.

+ Ta có thể chọn chế độ truyền cũng như các thông số của cổng thông tin nhưng nó phải như nhau ở tất cả các bộ điều khiển.

ModBus - ASCJJ Mode

+ Khi các bộ điều khiển sử dụng chế độ ASCII mỗi một byte-8bits truyền như là 2 ký tự ASCII.

+ Ưu điểm chính là cho thời gian truyền giữa các ký tự lên đến 1s mà không gây ra lỗi

+ Mã: Hexadecimal, ASCII 0-9,A-F. 1 Hexa ->ASCII

+ Bit trên ký tự: 1 Start bit; 7 data bit; 1,0 Parity bit; 1,2 Stop bit (10 bit).

+ Kiểm tra lỗi: LRC

ModBus - RTU Mode

+ Khi các bộ điều khiển hoạt động ở chế đọ RTU mỗi một Byte-8bit gửi như là hai số Hexadecimal -4 bit.

+ Ưu điểm của phương pháp này là có mật độ ký tự lớn cho phép truyền tốt hơn chế độ ASCII với cùng một tốc độ bit.

+ Mỗi một bản tin cần phải truyền thành một chuỗi liên tục.

+ Mã: 8 bit, Hexa 0-9,A-F. Hai số Hexa chứa trong một trường 8 bit.

+ Số bit trên Byte: 1 Start bit; 8 data bit; 1,0 Parity bit; 1,2 Stop bit (11 bit).

+ Kiểm tra lỗi: CRC

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus – Cấu trúc khung bản tin

+ Trong cả hai chế độ truyền bản tin Modbus được bên phát đặt trong một khung có điểm bắt đầu, kết thúc.

+ Bên thu nhận bản tin định vị các trường khác và phát hiện ra lỗi có trong bản tin.

+ Có hai chế độ truyền có hai kiểu khung bản tin

ModBus – ASCJJ Frame

START	ADDRESS	FUNCTION	DATA	LRC CHECK	END
1 CHAR :	2 CHARS	2 CHARS	*n* CHARS	2 CHARS	2 CHARS CRLF

ModBus – ASCJJ Frame

+ Tất cả các thiết bị nối vào mạng sẽ kiểm tra bus liên tục cho đến khi nhận được ký tự ':' . Nó sẽ giải mã trường địa chỉ. Nếu gửi cho nó thì nó nhận và xử lý các trường tiết theo.

+ Thời gian cho phép giữa các ký tự có thể lên đến 1 s-> không gây ra lỗi.

ModBus – RTU Frame

START	ADDRESS	FUNCTION	DATA	CRC CHECK	END
T1-T2-T3-T4	8 BITS	8 BITS	*n* x 8 BITS	16 BITS	T1-T2-T3-T4

ModBus – RTU Frame

+ Các thiết bị nối vào mạng sẽ kiểm tra bus trong suốt quá trình rỗi của bus. Trường đầu tiên nhận được sẽ là trường địa chỉ và nó sẽ so sánh với địa chỉ của nó.
+ Nếu thời gian nghỉ > 3.5 lần thười gian truyền 1 byte thì kết thúc bản tin.

ModBus – Cấu trúc khung bản tin

Trường địa chỉ:

+ Chứa 2 ký tự ASCII hay 8 bit.
+ Giá trị từ 0-247.
+ Từng Slave địa chỉ hoá từ 1-247.
+ Master địa chỉ hoá Slave bằng cách đặt địa chỉ của nó vào trường địa chỉ.
+ Slave trả lời báo cho Master biết Slave nào đã trả lời.
+ Địa chỉ 0 sử dụng ở chế độ quảng bá.

ModBus – Cấu trúc khung bản tin

Trường chức năng:

+ Bao gồm 2 ký tự ASCII hay 1 byte.
+ Giá trị từ 1-255.
+ Một vài mã áp dụng cho các bộ điều khiển. Một vài mã chỉ áp dụng cho một mô hình nào đó. Một số dành cho tương lai.
+ Master->Slave chỉ ra Slave phải làm gì?
+ Slave->Master báo là hoạt động bình thường hay báo lỗi. Nếu bình thường thì phản hồi về mã chức năng ban đầu. Nếu có lỗi thì phản hồi về mã chức năng ban đầu với bít cao nhất bằng 1.

ModBus – Cấu trúc khung bản tin

Trường dữ liệu:

- Master->Slave các dữ liệu cần cho hoạt động được định nghĩa bởi mã chức năng.
- Slave->Master nếu không có lỗi nó chức các dữ liệu trả về. Nếu có lỗi nó chứa mã lỗi.
- Trường dữ liệu có thể không có trong một số bản tin.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus – Cấu trúc khung bản tin

Kiểm tra lỗi:

+ ASCII mode: kết quả kiểm tra theo LRC -> 1byte -> 2 ký tự ASCII.

+ RTU mode: kiểm tra theo PP CRC nội dung bản tin. 16 bit -> 2 byte

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus – Cấu trúc khung bản tin

- Khi truyền ở chế độ Modbus chuẩn các ký tự hay các byte được truyền các bít thấp trước, cao sau.

With ASCII character framing, the bit sequence is:

With Parity Checking

Start	1	2	3	4	5	6	7	Par	Stop

Without Parity Checking

Start	1	2	3	4	5	6	7	Stop	Stop

ModBus – Cấu trúc khung bản tin

With RTU character framing, the bit sequence is:

With Parity Checking

Start	1	2	3	4	5	6	7	8	Par	Stop

Without Parity Checking

Start	1	2	3	4	5	6	7	8	Stop	Stop

ModBus – Các phương pháp kiểm tra lỗi

- Modbus chuẩn được áp dụng hai phương pháp kiểm tra lỗi. Kiểm tra chẵn lẻ áp dụng cho từng ký tự. Kiểm tra khung (LRC, CRC) được áp dụng cho toàn bộ khung bản tin.

- Cả hai phương pháp này sẽ được Master thực hiện trước khi truyền khung bản tin và được Slave kiểm tra trong quá trình nhận bản tin.

- Nếu Slave phát hiện ra lỗi trong bản tin thì bản tin sẽ bị bỏ đi, không có đáp ứng cho Master. Master đợi quá thời gian Time-out để bỏ quá trình truyền. Thời gian Time-out đủ lớn để cho bất kỳ Slave nào có thể trả lời bình thường được. Như vậy khi Time-out chương trình ứng dụng trên Master biết có một lỗi xảy ra.

ModBus – Các phương pháp kiểm tra lỗi

Kiểm tra chẵn lẻ:

- Ta có thể đặt là kiểm tra chẵn, kiểm tra lẻ hay không kiểm tra chẵn lẻ.
- Khi truyền các bit chẵn lẻ sẽ được tính toán và truyền cùng với ký tự. Bên thu sẽ kiểm tra lại. Tất cả các thiết bị phải dùng chung một phương pháp.

ModBus – Các phương pháp kiểm tra lỗi

Kiểm tra LRC:

Kiểm tra CRC:

ModBus -Các chức năng của Modbus

Modbus cung cấp một loạt các chức năng sau:

Code	Name	384	484	584	884	M84	984
01	Read Coil Status	Y	Y	Y	Y	Y	Y
02	Read Input Status	Y	Y	Y	Y	Y	Y
03	Read Holding Registers	Y	Y	Y	Y	Y	Y
04	Read Input Registers	Y	Y	Y	Y	Y	Y
05	Force Single Coil	Y	Y	Y	Y	Y	Y
06	Preset Single Register	Y	Y	Y	Y	Y	Y
07	Read Exception Status	Y	Y	Y	Y	Y	Y
08	Diagnostics (see Chapter 3)						

ModBus -Các chức năng của Modbus

09	Program 484	N	Y	N	N	N	N
10	Poll 484	N	Y	N	N	N	N
11	Fetch Comm. Event Ctr.	Y	N	Y	N	N	Y
12	Fetch Comm. Event Log	Y	N	Y	N	N	Y
13	Program Controller	Y	N	Y	N	N	Y
14	Poll Controller	Y	N	Y	N	N	Y
15	Force Multiple Coils	Y	Y	Y	Y	Y	Y
16	Preset Multiple Registers	Y	Y	Y	Y	Y	Y
17	Report Slave ID	Y	Y	Y	Y	Y	Y
18	Program 884/M84	N	N	N	Y	Y	N
19	Reset Comm. Link	N	N	N	Y	Y	N
20	Read General Reference	N	N	Y	N	N	Y
21	Write General Reference	N	N	Y	N	N	Y

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus -Các chức năng của Modbus

Code	Name	384	484	584	884	M84	984
22	Mask Write 4X Register	N	N	N	N	N	(1)
23	Read/Write 4X Registers	N	N	N	N	N	(1)
24	Read FIFO Queue	N	N	N	N	N	(1)

ModBus -Các chức năng của Modbus

QUERY

Field Name	Example (Hex)	ASCII Characters	RTU 8–Bit Field
Header		: (colon)	None
Slave Address	06	0 6	0000 0110
Function	03	0 3	0000 0011
Starting Address Hi	00	0 0	0000 0000
Starting Address Lo	6B	6 B	0110 1011
No. of Registers Hi	00	0 0	0000 0000
No. of Registers Lo	03	0 3	0000 0011
Error Check		LRC (2 chars.)	CRC (16 bits)
Trailer		CR LF	None
	Total Bytes:	17	8

ModBus -Các chức năng của Modbus

RESPONSE

Field Name	Example (Hex)	ASCII Characters	RTU 8–Bit Field
Header		: (colon)	None
Slave Address	06	0 6	0000 0110
Function	03	0 3	0000 0011
Byte Count	06	0 6	0000 0110
Data Hi	02	0 2	0000 0010
Data Lo	2B	2 B	0010 1011
Data Hi	00	0 0	0000 0000
Data Lo	00	0 0	0000 0000
Data Hi	00	0 0	0000 0000
Data Lo	63	6 3	0110 0011
Error Check		LRC (2 chars.)	CRC (16 bits)
Trailer		CR LF	None
	Total Bytes:	23	11

ModBus -Các kiểu dữ liệu của Modbus

- Modbus sử dụng 4 kiểu dữ liệu khác nhau:
  - + Đầu vào số.
  - + Đầu ra số (Coil).
  - + Thanh ghi vào (Input Register).
  - + Thanh ghi giữ (Holding Register)
- Các biến đầu vào và ra sô là 1 bit.
- Các biến thanh ghi là 2 byte.
- Mỗi một chức năng gắn liền với một kiểu dữ liệu.
- Địa chỉ mà khung bản tin sử dụng là địa chỉ offset tương đối với địa chỉ thấp nhất của kiểu dữ liệu.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus –Mô tả chi tiết các mã chức năng

QUERY

Field Name	Example (Hex)
Slave Address	11
Function	01
Starting Address Hi	00
Starting Address Lo	13
No. of Points Hi	00
No. of Points Lo	25
Error Check (LRC or CRC)	—

ModBus –Mô tả chi tiết các mã chức năng

RESPONSE

Field Name	Example (Hex)
Slave Address	11
Function	01
Byte Count	05
Data (Coils 27–20)	CD
Data (Coils 35–28)	6B
Data (Coils 43–36)	B2
Data (Coils 51–44)	0E
Data (Coils 56–52)	1B
Error Check (LRC or CRC)	—

ModBus –Mô tả chi tiết các mã chức năng

QUERY

Field Name	Example (Hex)
Slave Address	11
Function	02
Starting Address Hi	00
Starting Address Lo	C4
No. of Points Hi	00
No. of Points Lo	16
Error Check (LRC or CRC)	—

ModBus –Mô tả chi tiết các mã chức năng

RESPONSE

Field Name	Example (Hex)
Slave Address	11
Function	02
Byte Count	03
Data (Inputs 10204–10197)	AC
Data (Inputs 10212–10205)	DB
Data (Inputs 10218–10213)	35
Error Check (LRC or CRC)	—

Simpo PDF Merge and Split Unregistered Version - http://www.simopdf.com

ModBus -Mô tả chi tiết các mã chức năng

QUERY

Field Name	Example (Hex)
Slave Address	11
Function	05
Coil Address Hi	00
Coil Address Lo	AC
Force Data Hi	FF
Force Data Lo	00
Error Check (LRC or CRC)	—

ModBus -Mô tả chi tiết các mã chức năng

RESPONSE

Field Name	Example (Hex)
Slave Address	11
Function	05
Coil Address Hi	00
Coil Address Lo	AC
Force Data Hi	FF
Force Data Lo	00
Error Check (LRC or CRC)	—

ModBus -Function 08 Diagnostics

QUERY

Field Name	Example (Hex)
Slave Address	11
Function	08
Subfunction Hi	00
Subfunction Lo	00
Data Hi	A5
Data Lo	37
Error Check (LRC or CRC)	—

ModBus -Function 08 Diagnostics

RESPONSE

Field Name	Example (Hex)
Slave Address	11
Function	08
Subfunction Hi	00
Subfunction Lo	00
Data Hi	A5
Data Lo	37
Error Check (LRC or CRC)	—

Simpo PDF Merge and Split Unregistered Version - http://www.simopdf.com

ModBus -Function 08 Diagnostics

Code	Name	384	484	584	884	M84	984
00	Return Query Data	Y	Y	Y	Y	Y	Y
01	Restart Comm Option	Y	Y	Y	Y	Y	Y
02	Return Diagnostic Register	Y	Y	Y	Y	Y	Y
03	Change ASCII Input Delimiter	Y	Y	Y	N	N	Y
04	Force Listen Only Mode	Y	Y	Y	Y	Y	Y
05–09	Reserved						
10	Clear Ctrs and Diagnostic Reg.	Y	Y	(1)	N	N	(1)
11	Return Bus Message Count	Y	Y	Y	N	N	Y
12	Return Bus Comm. Error Count	Y	Y	Y	N	N	Y

ModBus -Function 08 Diagnostics

13	Return Bus Exception Error Cnt	Y	Y	Y	N	N	Y
14	Return Slave Message Count	Y	Y	Y	N	N	N
15	Return Slave No Response Cnt	Y	Y	Y	N	N	N
16	Return Slave NAK Count	Y	Y	Y	N	N	Y
17	Return Slave Busy Count	Y	Y	Y	N	N	Y
18	Return Bus Char. Overrun Cnt	Y	Y	Y	N	N	Y
19	Return Overrun Error Count	N	N	N	Y	N	N
20	Clear Overrun Counter and Flag	N	N	N	Y	N	N
21	Get/Clear Modbus Plus Statistics	N	N	N	N	N	Y
22–up	Reserved						

ModBus -Trả lời báo lỗi

Ngoại trừ bản tin quảng bá. Khi Master gửi một bản tin tới hỏi Slave thì có 4 trường hợp có thể xảy ra:

+ Slave nhận bản tin không có lỗi và có thể trả lời bản tin. Trả lời thường.

+ Nếu Slave không nhận được bản tin hỏi vì lỗi thông tin, sẽ không có bản tin trả lời.Master sử lý sự kiện Time-out.

+ Nếu Slave nhận được bản tin hỏi nhưng có lỗi thông tin (Parity, LRC,CRC), sẽ không có bản tin trả lời.Master sử lý sự kiện Time-out.

+Nếu Slave nhận được bản tin không bị lỗi thông tin nhưng không thể thực hiện được thì nó sẽ trả lời bản tin báo lỗi. Nó báo cho Master lỗi nào đã xảy ra.

ModBus -Trả lời báo lỗi

Bản tin báo lỗi bao gồm:

+ Trường địa chỉ báo thiết bị nào trả lời.
+ Trường chức năng với bit cao nhất bằng 1.
+ Trường dữ liệu báo lỗi nào đã xảy ra.
+ Trường kiểm tra lỗi.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus -Trả lời báo lỗi

QUERY

Byte	Contents	Example
1	Slave Address	0A
2	Function	01
3	Starting Address Hi	04
4	Starting Address Lo	A1
5	No. of Coils Hi	00
6	No. of Coils Lo	01
7	LRC	4F

EXCEPTION RESPONSE

Byte	Contents	Example
1	Slave Address	0A
2	Function	81
3	Exception Code	02
4	LRC	73

ModBus -Trả lời báo lỗi

Code	Name	Meaning
01	ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the slave. If a Poll Program Complete command was issued, this code indicates that no program function preceded it.
02	ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the slave.
03	ILLEGAL DATA VALUE	A value contained in the query data field is not an allowable value for the slave.

ModBus -Trả lời báo lỗi

04	SLAVE DEVICE FAILURE	An unrecoverable error occurred while the slave was attempting to perform the requested action.
05	ACKNOWLEDGE	The slave has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the master. The master can next issue a Poll Program Complete message to determine if processing is completed.
06	SLAVE DEVICE BUSY	The slave is engaged in processing a long–duration program command. The master should retransmit the message later when the slave is free.

ModBus -Trả lời báo lỗi

07	NEGATIVE ACKNOWLEDGE	The slave cannot perform the program function received in the query. This code is returned for an unsuccessful programming request using function code 13 or 14 decimal. The master should request diagnostic or error information from the slave.
08	MEMORY PARITY ERROR	The slave attempted to read extended memory, but detected a parity error in the memory. The master can retry the request, but service may be required on the slave device.

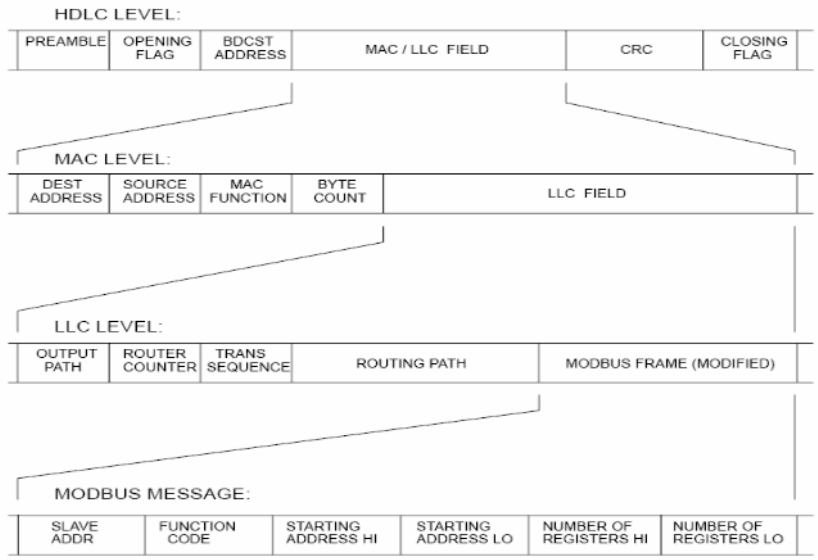
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

ModBus Plus

- Là một hệ thống Bus dựa trên Modbus nhưng có giá thành thấp, dễ láp đặt, cài đặt.
- Cho phép đành địa chỉ 64 nút trên mạng, tốc độ truyền 1 Mbps.
- Mạng Peer-to-peer, sử dung MAC là Token passing.

ModBus Plus



ModBus Plus

- Output Path (1 byte):Đường dẫn đầu ra chỉ một kênh logic của trạm chủ, có vai trò trong việc dồn kênh/phân kênh.
- Router counter (1 byte): đếm số router mà khung bản tin đã đi qua.
- Transaction Sequence Number: Mã số giao dịch
- Routing Path (5 byte): Mã số đường dẫn chức thông tin chọn đường tối ưu trong liên mạng.
- DA (1 byte): địa chit trạm đích.
- SA (1 byte): Địa chỉ trạm nguồn.
- MAC Function (1 byte): mã hàm điều khiển truy nhập đường truyền.

ModBus Plus

- Byte Count (2 byte) số lượng byte trong phần LLC được truyền.
- Preamble (1 byte): dãy bit báo hiệu đầu khung.
- Opening Flag (1 byte): Cờ mở đầu khung.
- Broadcast Address (1 byte): địa chỉ gửi đồng loạt.
- CRC (2 byte): kiểm tra lỗi CRC.
- Closing Flag (1 byte): cờ báo kết thúc

Mạng máy tính
&
Hệ thống thông tin công nghiệ

Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN (Controller Area Network)

Mô tả chung về giao thức:

- CAN là một giao thức thông tin nối tiếp, cung cấp hệ điều khiển thời gian thực , phân tán với độ tin cậy cao.
- CAN là một chuẩn của ISO (ISO11898).
- CAN được phát triển năm 1980 bởi BOSCH.
- Nó ứng dụng trong CN SX ô-tô, máy công cụ, máy đóng bao...

CAN (Controller Area Network)

CAN bao gồm các lớp sau:

- the (CAN-) object layer
- the (CAN-) transfer layer
- the physical layer.

CAN (Controller Area Network)

- Object:
  + Phát hiện các bản tin đã được truyền đi.
  + Quyết định bản tin nào sẽ được nhận bởi lớp Transfer và được sử dụng.
  +Cung cấp giao diện tới người dùng và các phần cứng liên quan.

Người dùng có thể định nghĩa các đối tượng kết nối.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN (Controller Area Network)

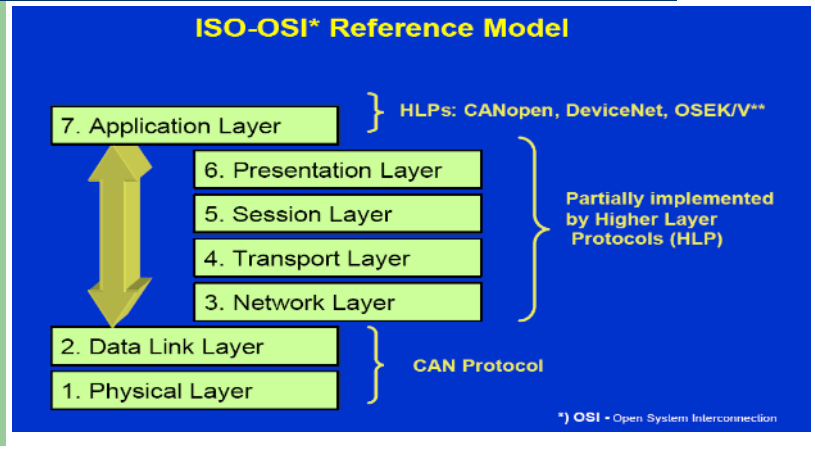
- Transfer:

Điều khiển việc định khung, định thời, thực hiện chức năng trọng tài, kiểm tra lỗi, phát hiện lỗi và hạn chế lỗi.Quyết định khi nào thì truyền và nhận bản tin. Người dụng không được tự do thay đổi phần này.

- Physical:

Truyền các bit giữa các nút với 1 tiêu chuẩn về điện.

CAN (Controller Area Network)



CAN

Layered Structure of a CAN Node

Application Layer
Object Layer - Message Filtering - Message and Status Handling
Transfer Layer - Fault Confinement - Error Detection and Signaling - Message Validation - Acknowledgement - Arbitration - Message Framing - Transfer Rate and Timing
Physical Layer - Signal Level and Bit Representation - Transmission Medium

CAN - Các khái niệm cơ bản

- Bản tin: thông tin được truyền theo 1 vài định dạng cố định, có độ dài hạn chế. khi Bus rỗi một thiết bị có thể truyền 1 bản tin.
- Định tuyến thông tin: Một nut CAN không chứa bất kỳ thông tin nào về hệ
  + Hệ mềm dẻo.
  + Định tuyến bản tin: sử dụng Identifier và Message Filtering.
- Multi Cast.
- Độ bảo toàn dữ liệu.

CAN - Các khái niệm cơ bản

- Tốc độ truyền.
- Mức độ ưu tiên của bản tin.
- Yêu cầu dữ liệu từ xa.
- Multi Master.
- Trọng tài.

CAN - Các khái niệm cơ bản

- An toàn dữ liệu:
  - + Kiểm tra lỗi:
    - . Giám sát.
    - .CRC
    - . Nhồi bit.
    - .Kiểm tra khung bản tin.
  - + Hiệu quả:
    - .Tất cả các lỗi toàn cục.
    - .Lỗi tại bộ truyền.
    - . 5 lỗi phân bố ngẫu nhiên.
    - .chuỗi lỗi < 15 bit
    - . Các lỗi lẻ
    - . Còn lại 4.7 10-11

CAN - Các khái niệm cơ bản

- Báo lỗi và thời gian phục hồi.
- Hạn chế lỗi.
- Kết nối trong CAN
- Kênh truyền.
- Giá trị bit.
- ACK
- Sleep/Wake up.

CAN - Cấu trúc bản tin

Các bản tin được truyền và xử lý trong CAN theo 4 kiểu khác nhau của khung bản tin:

- Data Frame: mang thông tin từ nơi phát đến nơi thu.
- Remote Frame: gửi đi một yêu cầu truyền một Data Frame với cùng một Identifier.
- Error Frame: truyền đi bởi bất kỳ một nút nào phát hiện ra lỗi trên Bus.
- Overload Frame: cung cấp một thời gian trễ giữa Data Frame và Remote Frame.

Giữa Data Frame và Remote Frame được phan biệt với nhau bởi InterFrame Space.

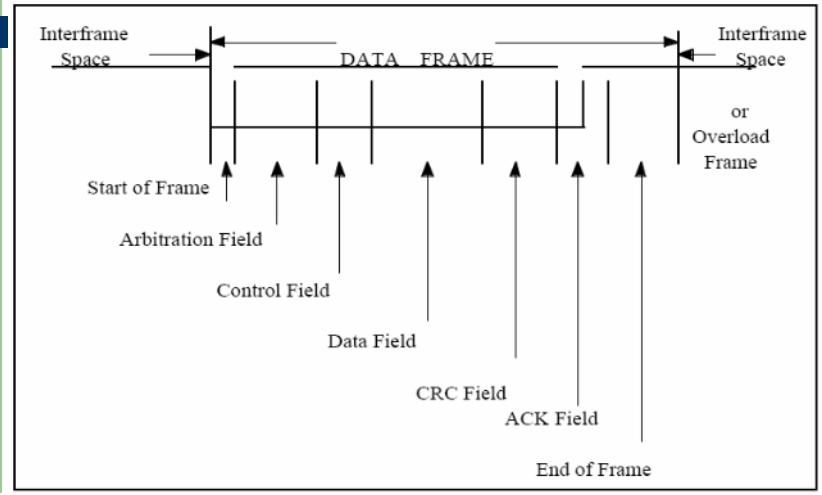
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN – Cấu trúc bản tin

Data Frame: bao gồm 7 trường bit.

- START OF FRAME.
- ARBITRATION FIELD
- CONTROL FIELD
- DATA FIELD
- CRC FIELD
- ACK FIELD
- END OF FRAME

CAN – Cấu trúc bản tin



CAN – Cấu trúc bản tin

- START OF FRAME: đánh dấu việc bắt đầu một Data Frame, hay Remote Frame. Nó gồm 1 bit "trội" (Dominant)

- Trạm có thể gửi số liệu khi bus rỗi.
- Các trạm sẽ đồng bộ với sườn củaSTART OF FRAME.

CAN – Cấu trúc bản tin

ARBITRATION FIELD: bao gồm IDENTIFIER and the RTR-BIT.

IDENTIFIER bao gồm 11 bit ID10 - ID0. Các bit cao truyền đi trước.

7 bit cao nhất từ ID10-ID4 không được tất cả là "lặn" (Recessive)

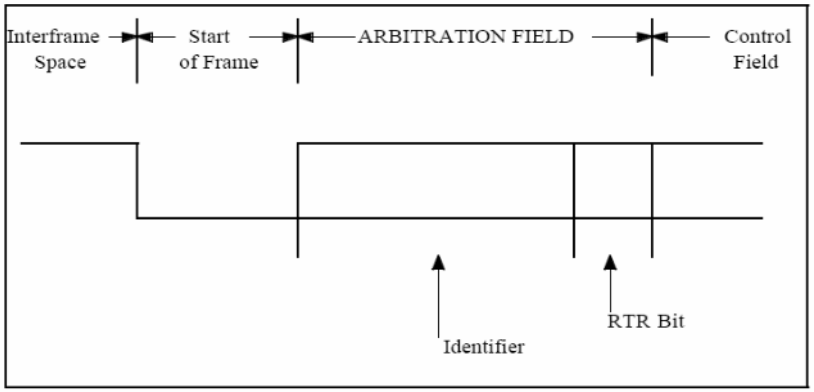
RTR BIT (Remote Transmission Request BIT)

" trội" nếu là Data Frame.

" lặn" nếu là Remote Frame.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN – Cấu trúc bản tin



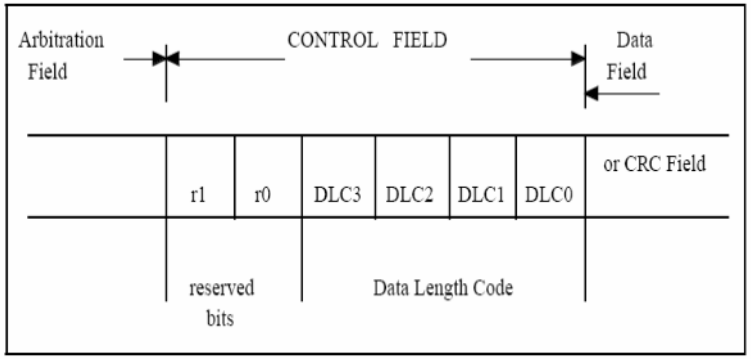
CAN – Cấu trúc bản tin

CONTROL FIELD: bao gồm 6 bit.

4 bit mã hoá độ dài của trường dữ liệu 0...8

2 bit danh cho mở rộng trong tương lai

CAN – Cấu trúc bản tin



CAN – Cấu trúc bản tin

Number of Data Bytes	Data Length Code			
	DLC3	DLC2	DLC1	DLC0
0	d	d	d	d
1	d	d	d	r
2	d	d	r	d
3	d	d	r	r
4	d	r	d	d
5	d	r	d	r
6	d	r	r	d
7	d	r	r	r
8	r	d	d	d

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

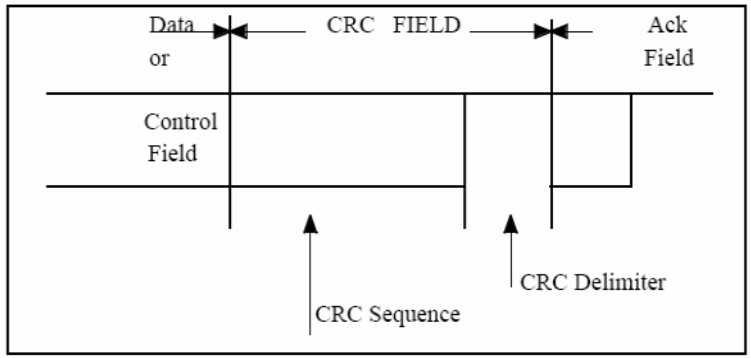
CAN – Cấu trúc bản tin

- Data Field : Chứa 0...8 byte dữ liệu.
- CRC Field: bao gồm CRC sequence và CRC delimiter

CRC sequence là kết quả tính toán theo phương pháp CRC các trường: START OF FRAME, ARBITRATION FIELD, CONTROL FIELD, DATA FIELD

- Sử dụng đa thức: $X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1$.
- CRC delimiter bao gồm một bit lặn.

CAN – Cấu trúc bản tin

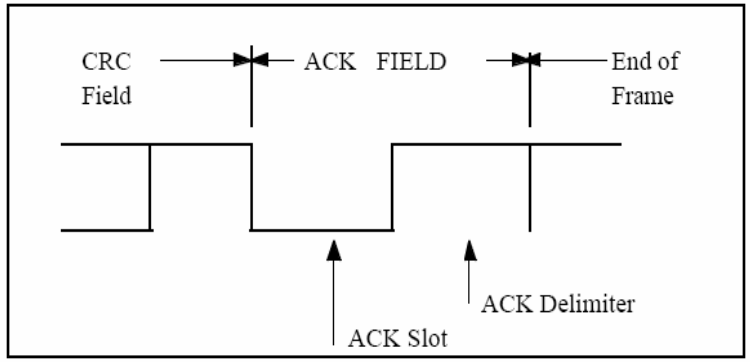


CAN – Cấu trúc bản tin

- ACK FIELD bao gồm 2 bit ACK SLOT và ACK DELIMITER.
- Khi truyền bên truyền sẽ gửi đi hai bit lặn.
- Nếu bên nhận nhận tốt bản tin sẽ gửi bit trội vào ACK SLOT

- End of Frame: là một cờ bao gồm 7 bit lặn.

CAN – Cấu trúc bản tin



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

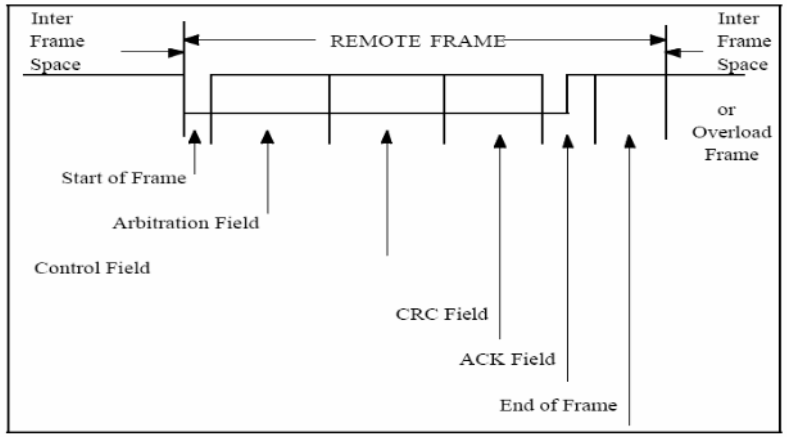
CAN – Cấu trúc bản tin

Remote Frame: bao gồm 6 trường bit.

- START OF FRAME.
- ARBITRATION FIELD
- CONTROL FIELD
- CRC FIELD
- ACK FIELD
- END OF FRAME

(Tương tự như Data Frame)

CAN – Cấu trúc bản tin



CAN – Cấu trúc bản tin

Error Frame bao gồm Error Flag + Error Delimiter.

Error Flag : bao gồm hai loại

ACTIVE ERROR FLAG bao gồm 6 bit trội

PASSIVE ERROR FLAG bao gồm 6 bit lặn

- Một trạm ở trạng thái lỗi tích cực nếu phát hiện ra lỗi sẽ truyền đi một cờ lỗi tích cực. Luật của cờ lỗi sẽ phá huỷ luật chèn bit hay định dạng cố định của bản tin.
- Một trạm ở trạng thái lỗi bị động nếu phát hiện ra lỗi thi sẽ truyền di cờ lỗi bị động.

CAN – Cấu trúc bản tin

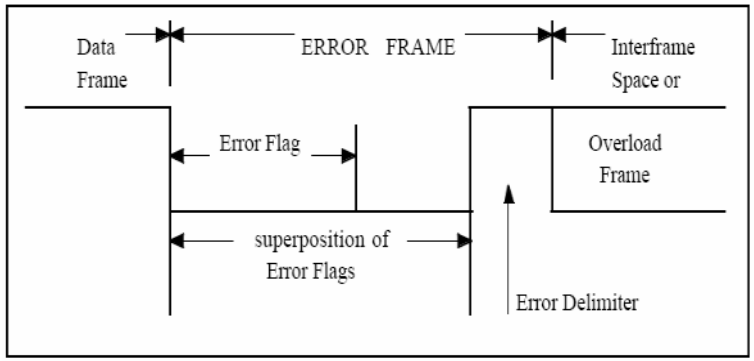
- ERROR DELIMITER: bao gồm 8 bit lặn.

Sau khi truyền đi cờ lỗi trạm sẽ truyền đi các bit lặn sau đó sẽ giám sát Bus cho tới khi nhận được các bit lặn thì truyền thêm 7 hay nhiều hơn các bit lặn.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN – Cấu trúc bản tin



CAN – Cấu trúc bản tin

OVERLOAD FRAME bao gồm hai trường OVERLOAD FLAG và OVERLOAD DELIMITER.

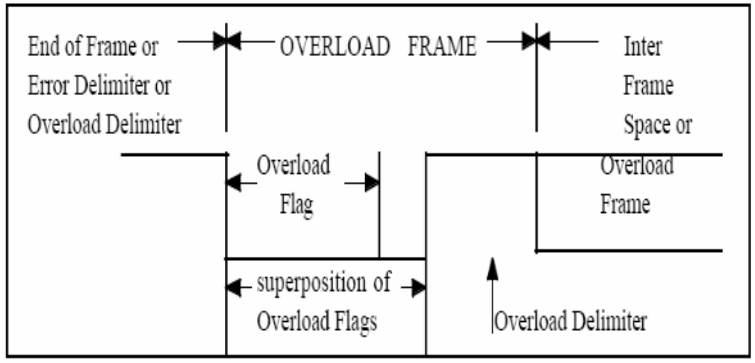
Có hai điều kiện quá tải:

- Điều kiện bên trong bộ nhận mà nó yêu cầu một thời gian trễ cho Data Frame và Remote Frame tiếp theo.
- Phát hiện ra một bit trội trong quá trình Intermission.

OVERLOAD FLAG bao gồm 6 bit trội.

OVERLOAD DELIMITER bao gồm 8 bit lặn

CAN – Cấu trúc bản tin



CAN – Cấu trúc bản tin

INTERFRAME SPACE bao gồm hai trường INTERMISSION và BUS IDLE

INTERMISSION bao gồm 3 bit lặn.

Các Data Frame và Remote Frame được cách nhau bởi INTERFRAME SPACE.

INTERMISSION các trạm không được truyền các Data hay Remote Frame mà chỉ có thể truyền điều khiện quá tải.

BUS IDLE trạm có thể truyền các bản tin.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN

- Mã hoá thông tin: Các trường START OF FRAME, ARBITRATION FIELD, CONTROL FIELD,DATA FIELD and CRC SEQUENCE của bản tin được mã hoá bởi PP nhồi bit.
- Nếu bên phát phát hiện ra 5 bit liền nhau giông nhau trong chuỗi bit đã truyền đi thì nó sẽ tự động chèn một bit đảo vào.
- Các trường còn lại của DATA FRAME hay REMOTE FRAME (CRC DELIMITER, ACK FIELD and END OF FRAME) không bị chèn. ERROR FRAME và OVERLOAD FRAME duy trì định dạng cố định không bị chèn.
- các bit mã hoá NRZ.

CAN - Kiểm soát lỗi.

Phát hiện lỗi: Có 5 kiểu lỗi khác nhau

- BIT ERROR
- STUFF ERROR
- CRC ERROR
- FORM ERROR
- ACKNOWLEDGEMENT ERROR

CAN - Kiểm soát lỗi.

Báo lỗi: Một trạm khi phát hiện ra lỗi thì sẽ gửi đi mộ cờ lỗi. Các lỗi như BIT ERROR, STUFF ERROR, FORM ERROR ACKNOWLEDGEMENT ERROR phát hiện ra ở bit nao thì cờ lỗi sẽ truyền ở bit tiếp theo.

Nếu lỗi CRC được phát hiện thì cò lỗi sẽ truyền sau ACK delimiter.

CAN - Hạn chế lỗi

Hạn chế lỗi các nút CAN chia ra làm 3 trạng thái:

- error active
- error passive
- bus off

Mỗi một nút có hai bộ đếm:

TRANSMIT ERROR COUNT
RECEIVE ERROR COUNT

CAN - Hạn chế lỗi

Các luật thay đổi giá trị bộ đếm:

1. Khi bộ nhận phát hiện ra 1 lỗi thì bộ đếm lỗi nhận tăng lên 1. Ngoại trừ trường hợp lỗi bit khi truyền cò lỗi tích cực hay quá tải.

2.Khi bộ nhận phát hiện ra một bit trội là bít đầu tiên sau khi truyền đi cờ lỗi thì bộ đếm lỗi nhận tăng lên 8.

3. Khi bộ truyền phát hiện ra một lỗi thì nó sẽ truyền đi 1 cờ lỗi và bộ đếm lỗi truyền sẽ tăng lên 8.

4. Nếu bộ truyền phát hiện ra lỗi bit khi truyền cờ lỗi tích cực hay cờ quá tải thì bộ đếm lỗi truyền tăng lên 8.

5. Nếu bộ nhận phát hiện ra lỗi bit khi truyền cờ lỗi tích cực hay cờ quá tải thì bộ đếm lỗi nhận tăng lên 8.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CAN - Hạn chế lỗi

6. Nếu bất kỳ nút nào chịu 7 bit trội liền nhau sau khi truyền đi cờ lỗi tích cực, bị động, cờ quá tải thì bbộ đếm lỗi truyền và nhận tăng lên 8.

7. Sau khi truyền tốt một bản tin thì bộ đếm lỗi truyền giảm đi 1 ngoại trừ nó đã =0.

8. Sau khi nhận tốt một bản tin bộ đếm lỗi nhận:

- giảm đi 1 nếu nó <127
- =0 nếu =0.
- 119-127 nếu nó >128.

9. Nút lỗi bị động nếu có BĐLT hay BĐLN >127.

CAN - Hạn chế lỗi

10. Bus off nếu có BĐLT>256

11. Lỗi bị đông nếu BĐLT và BĐLN <=127 .

12. Bus off-> lỗi tích cực vơi BĐLT=BĐLN=0 sau 128 sự kiện 11 bit lặn liện nhau được ghi nhận trên Bus.

CAN - Hạn chế lỗi

Mạng máy tính
&
Hệ thống thông tin công nghiệ

Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Lịch sử phát triển

Sự xuất hiện của nhiều hệ bus trường khác nhau dẫn đến việc ra đời của hai tổ chức ISP và Worldfip vào năm 1993, với cùng mục đích là xây dựng một chuẩn bus trường thống nhất. Trong khi ISP về cơ bản dựa trên nền tảng là PROFIBUS, Worldfip đại diện cho giới sản xuất và sử dụng các sản phẩm FIP. Cuối năm 1994, các thành phần đại diện phía Bắc Mỹ trong hai tổ chức này đi tới thống nhất thành lập hiệp hội mang tên Fieldbus *Foundation* (FF) nhằm chấm dứt sự phân nhánh trong việc xây dựng chuẩn. Tuy nhiên, các tư tưởng đại diện trong tổ chức mới này không dựa hẳn vào PROFIBUS hay FIP, mà hướng tới một hệ bus twờng mới sử dụng lớp vật lý theo IEC 1158-2.

Foundation Fieldbus – Lịch sử phát triển

Điều này dẫn tới việc các thành phần đại diện châu âu đã rút lui và quay trở lại với hệ thống của họ trong khuôn khổ PNO *(PROFIBUS Nutzerorganisation)* cũng như Worldfip.

Hiện nay Fieldbus Foundation có hơn 130 công ty thành viên trên khắp thế giới. chiếm đại đa số các nhà cung cấp thiết bị đo lường và điều khiển. Hệ thống bus trường wợc phát triển trong khuôn khổ của FF được gọi là *Foundation fieldbus.*

Foundation Fieldbus – Lịch sử phát triển

Tương tự như PROFIBUS-PA, phạm vi ứng dụng tiêu biểu của H1 là các ngành công nghiệp chế biến. Các công ty lớn như ABB,Fisher-rosemount (Emerson Process Management), Honeywell, National Instruments, Endress+hauser và Yokogawa đều có hàng loạt sản phẩm hỗ trợ .

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Tại sao?

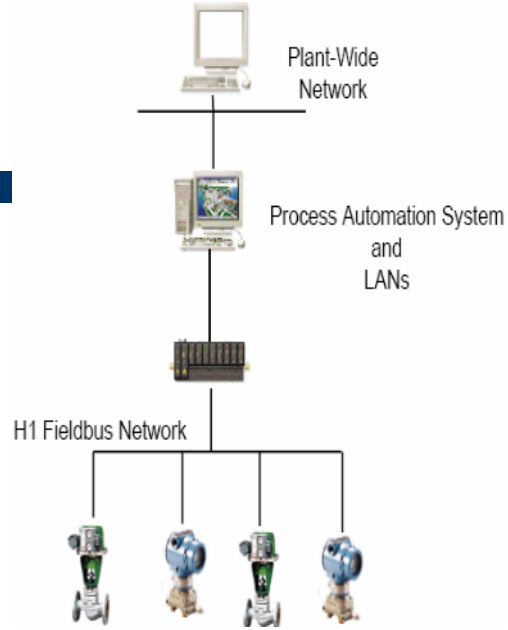
FF không chỉ là một giao thức thông tin mà nó còn có các đặc điểm sau:

- Thay thế hoàn toàn cho hệ thống cũ 4-20 mA.
- Các chức năng điều khiển, cảnh báo, theo dõi quá trình...được phân tán tới các thiết bị trong hệ.
- Cho phép các nhà thiết bị của các nhà SX nhau.
- Hệ thống mở

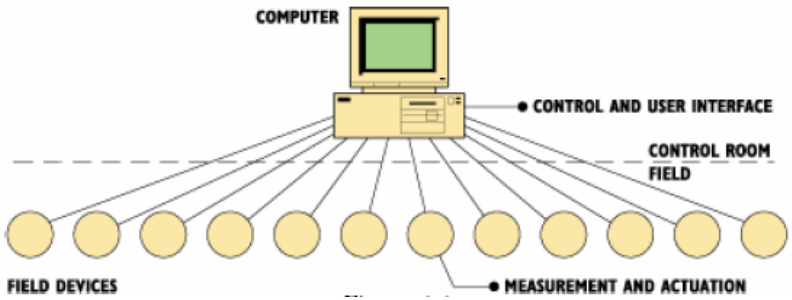
Các thiết bị FF là các thiết bị thông minh.

FF là hệ đầy đủ với các chức năng điều khiển phân tán ở các thiết bị nhưng nó vẫn cho phép hoạt động và điều khiển từ phòng điều khiển trung tâm

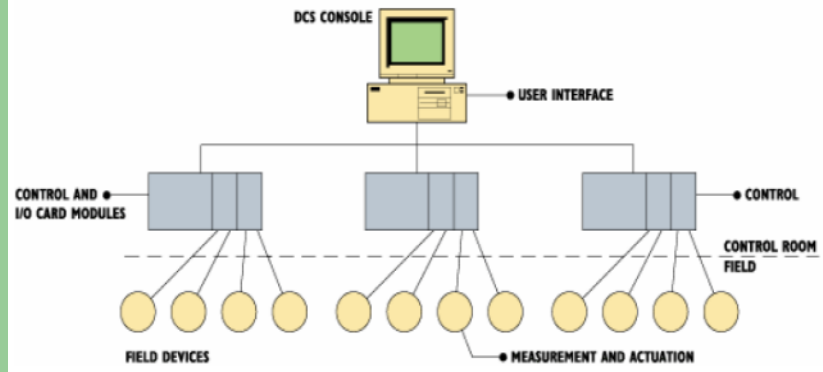
Foundation Fieldbus – Tại sao?



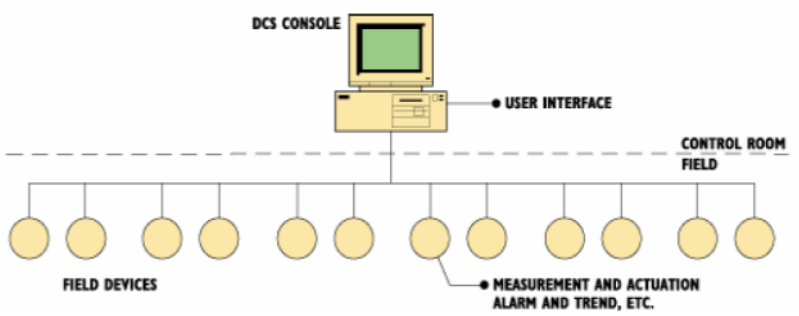
Foundation Fieldbus – Tại sao?



Foundation Fieldbus – Tại sao?



Foundation Fieldbus – Tại sao?



Foundation Fieldbus – Tại sao?

Các ưu điểm của hệ thống thông tin số so với chuẩn 4-20 mA:

- Độ chính xác cao, độ đảm bảo dữ liệu cao.
- Cho phép đa biến.
- Có thể đặt cấu hình và chuẩn đoán từ xa.
- Giảm đấu dây.
- .....

Foundation Fieldbus – Tại sao?

Các nhược điểm của hệ thống thông tin số so với chuẩn 4-20 mA:

- Tốc độ thông tin chậm so với để điều vòng kín.
- Không có giao tiếp của các nhà SX khác nhau.
- Phải kiểm tra trạng thái theo kiểu hỏi vòng.
- .....

Foundation Fieldbus – Lợi ích

- Hoạt động với độ tin cậy cao hơn.
- Độ mềm dẻo hầu như không có giới hạn.
- Giảm giá thành thiết bị.
- Giảm giá thành lắp đặt.
- Lượng thông tin lớn.

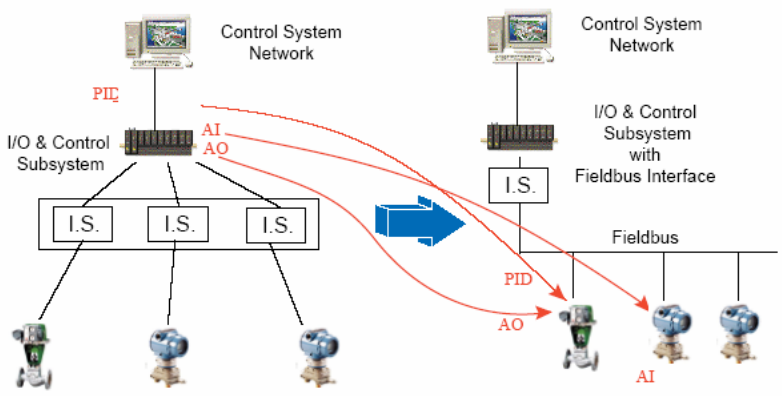
Hệ tương tự dễ hiểu hơn (người dùng chỉ cần 1 screwdriver, và 1 đồng hồ đo dòng có thể kiểm tra, cấu hình các thiết bị).

FF báo các vấn đề một cách trực tiếp, thậm trí trước khi nó xảy ra.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

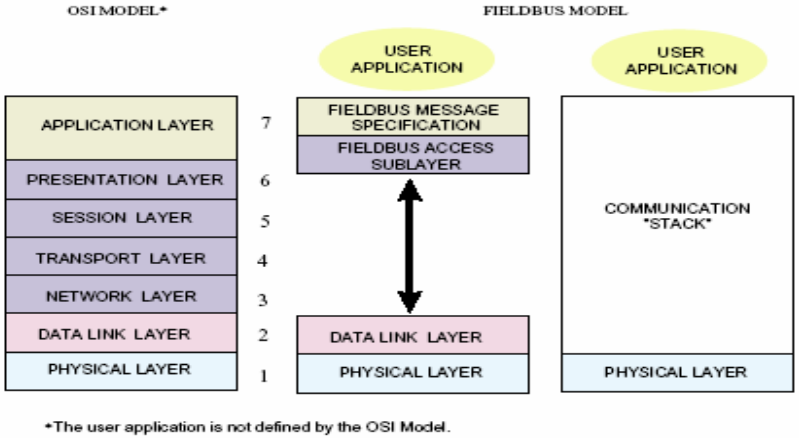
Foundation Fieldbus – Lợi ích



Foundation Fieldbus – Lợi ích

- Khả năng đa biến có thể cho phép kết hợp các bộ điều khiển và các bộ xử lý tín hiệu.
- FF cho phép kết nối vài trăm thiết bị, khoảng cách vài km với 1 đôi dây.
- FF có các khối chức năng phần mềm thay thế cho các khối phần cứng-> thay đổi hệ điều khiển mà không cần đi lại dây hay thay đổi phần cứng.
- Các kết nối có thể thay đổi, các khối chức năng có thể thêm vào hay bớt đi, ta có thực hiện gần như là vô hạn các khối chức năng, hệ có thể mở rông với phần cứng tối thiểu.

Foundation Fieldbus – Kiến trúc giao thức



Foundation Fieldbus – Physical

Môi trường truyền dẫn:

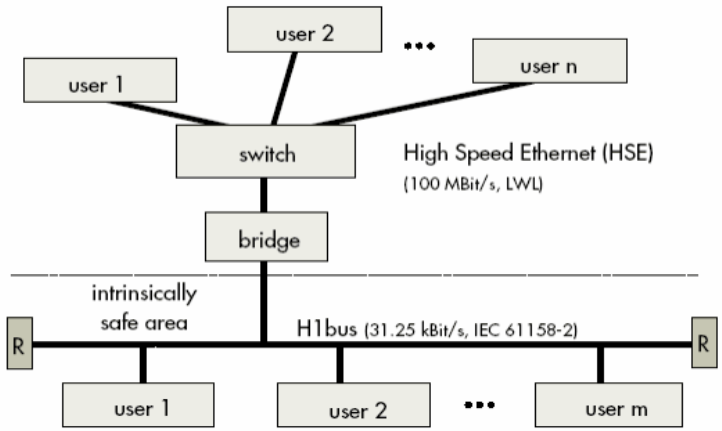
- Cáp điện (hay dùng cáp xoắn)
- Cáp quang.

Tốc độ truyền:

- 31,25 kbps (H1)
-1 Mbps (H2)
-2,5 Mbps (H2)

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Physical

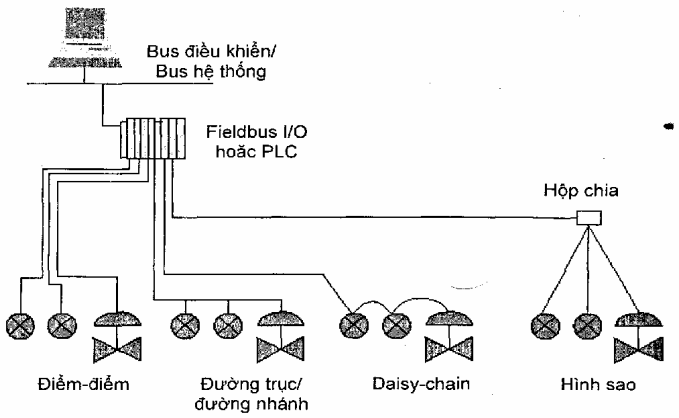


Foundation Fieldbus – Physical

Cấu trúc mạng:

- Cấu trúc dạng Bus (Đường trục/nhánh, Daisy-chain)
- P-to-P.
- Cây.

Foundation Fieldbus – Physical



Foundation Fieldbus – Physical

Số thiết bị	Chiều dài lớn nhất của nhánh
25 – 32	1 m (3.28 ft)
19 – 24	30 m (98.42 ft)
15 – 18	60 m (196.8 ft)
13 – 14	90 m (295.2 ft)
1 – 12	120 m (393.6 ft)

Quan hệ giữa số lượng thiết bị và chiều dài cực đại của nhánh

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus - Physical

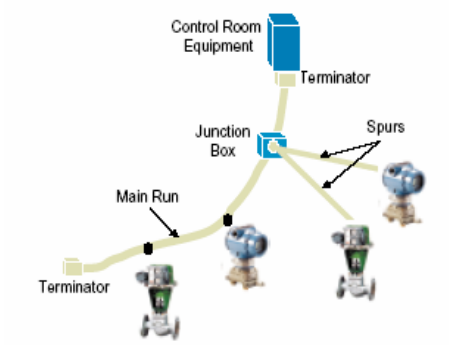
Khoảng cách truyền tối đa phụ thuộc vào tốc độ truyền:

- 31,25 kbps - 1900 m
- 1 Mbps - 750 m
- 2,5 Mbps - 500 m

Số trạm trên 1 đoạn mạng phụ thuộc vào công suất nguồn, loại cáp. tuy nhiên tối đa là 32 trạm.

Sử dung 4 Repaeter : 9500 m, 240 trạm

Foundation Fieldbus - Physical



Foundation Fieldbus - Physical

Characteristics	Data Rate		
Type	**31.25 kbit/s Voltage**	**31.25 kbit/s Voltage**	**31.25 kbits Voltage**
Topology	Bus/tree	Bus/tree	Bus/tree
Power	none	DC	DC
Classification		Intrinsically Safe	
Number of Devices	2-32	2-32	2-32
Cable Length	1900 m	1900 m	1900 m
Spur Length	120 m	120 m	120 m

Foundation Fieldbus - Physical

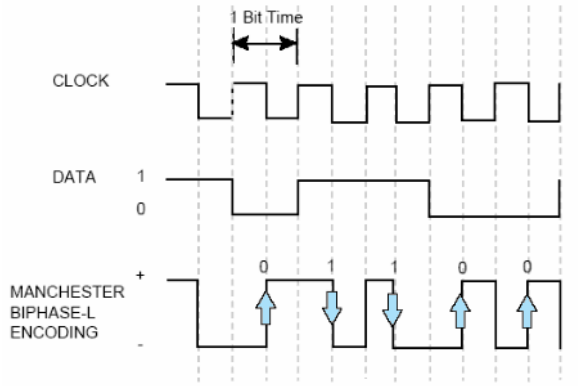
Số liệu được trao đổi theo phương thức truyền đồng bộ, bán song công sử dụng mã Manchester.

Có khả năng đồng tải nguồn trên đường truyền.
Nguồn từ 9-32 VDC

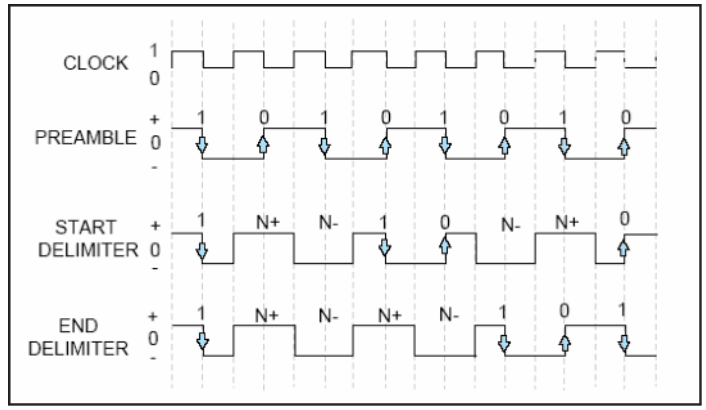
Terminator có dạng R-C

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

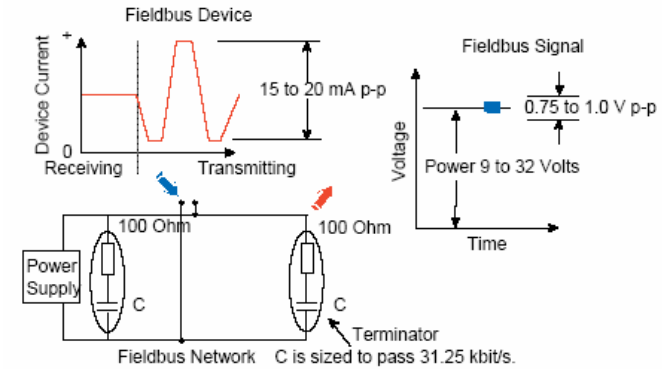
Foundation Fieldbus - Physical



Foundation Fieldbus - Physical



Foundation Fieldbus - Physical



NOTE: As an option, one of the terminators may be center-tapped and grounded to prevent voltage buildup on the fieldbus.

Foundation Fieldbus - Data Link

FMAC:

- Là sự kết hợp của các phương pháp Master/Slave, Token Passing và TDMA.
- Một thiết bị đóng vai trò trạm chủ gọi lá LAS (Link Active Scheduler) phân chia và kiểm soát quyền truy nhập cho toàn mạng.
- Các thiết bị FF chia 2 loại Basic Device, Link Master. Chỉ có Link Master mới có thể trở thành LAS.

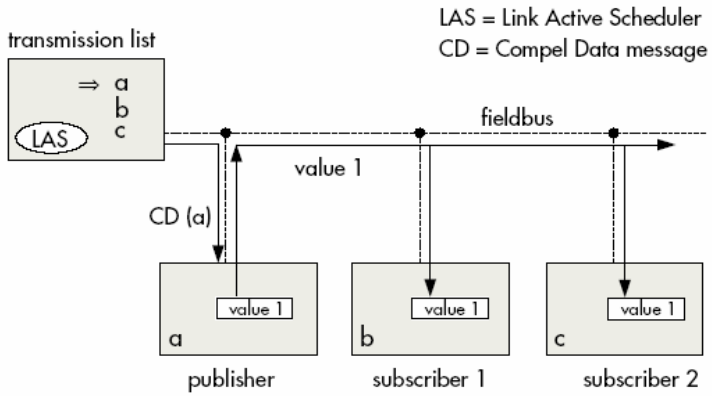
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Data Link

FDLC: Có hai cơ chế giao tiếp là lập lịch và không lập lịch.

- LAS có một danh sách các thời điểm truyền cho tất cả các vùng đệm dữ liệu trong tất cả các thiết bị điều này cần thiết đẻ việc truyền dữ liệu có chu kỳ . Khi đến thời điểm truyền dữ liệu của một thiết bị nào đó,LAS cấp cho thiết bị đó một bản tin cưỡng bức.Sau khi nhân bản tin này thiết bị truyền thông tin trong vùng đệm của mình tới toàn bộ thiết bị trên bus thiết bị có nhu cầu nhận bản tin đó gọi là Người thuê bao.Việc truyền dữ liệu tiền định đực sử dụng trong các trường hợp bình thường,chu kỳ truyền của vòng dữ liệu được kiểm soát giữa thiết bị và Bus.

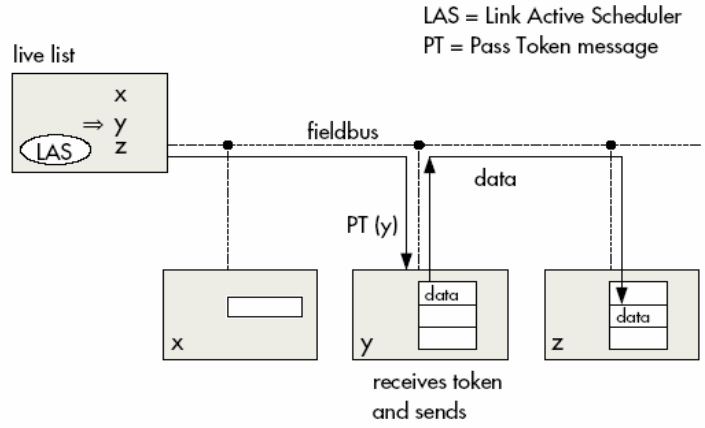
Foundation Fieldbus – Data Link



Foundation Fieldbus – Data Link

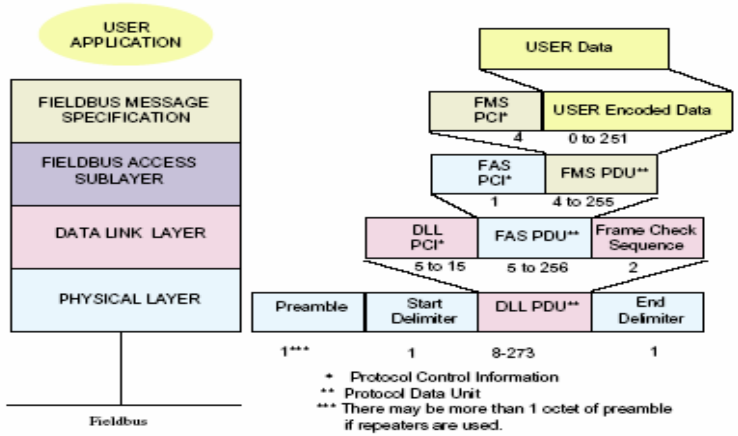
Tất cả các thiết bị trên bus đều có cơ hội gửi các bản tin không định trước giữa các bản tin tiền định. LAS cấp quyền truy cập cho một thiết bị trên bus bằng việc cấp cho thiết bị đó một thẻ bài. Khi thiết bị nhận được thẻ bài nó được phép gửi bản tin cho tới khi vượt quá thời gian giữ thẻ bài lớn nhất có thể.Bản tin có thể đực gửi tới một đich hoặc nhiều đích khác nhau.

Foundation Fieldbus – Data Link



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Cấu trúc bức điện



Foundation Fieldbus – Các dịch vụ giao tiếp

- Khi giải quyết một bài toán phức tạp như FF người ta cần phải phân tích bài toán ra thành các phần, thậm chí thành các phần tử cơ bản.
- Sử dụng thiết kế hướng đối tượng để thiết kế quá trình ứng dụng và các khối chức năng của QTƯD.
- Object là một thực thể có thể thực hiện 1 công việc nào đó.
- Phần mềm trên cơ sở các Obj sẽ thực thi công việc khi có bản tin gửi nhiệm vụ tới chúng -> TKHĐT không có Angorithm.
- Đối tượng được chia thành các lớp phù hợp

Foundation Fieldbus – Các dịch vụ giao tiếp

Fieldbus Access Sublayer (FAS)

Lớp con FAS sử dụng hai cơ chế giao tiếp ở lớp 2 để cung cấp các dịch vụ cho lớp FMS. Kiểu dịch vụ FAS được mô tả bởi các quan hệ giao tiếp ảo VCR *(Vtrtual* Communication *Relationships)*. Ba kiểu VCR được định nghĩa nh sau:

- Kiểu *Client/server:* Giao tiếp không lập lịch giữa một trạm gửi *(server)* và một trạm nhận (client). các thông báo được xếp trong hàng đợi theo thứ tự có ưu tiên. Kiểu VCR này thường được sử dụng trong việc nạp chương trình lên xuống, thay đổi các tham số điều khiển hoặc xác nhận báo cáo.

Foundation Fieldbus – Các dịch vụ giao tiếp

- Kiểu phân phối báo cáo *(Report* Distribution): Giao tiếp không lập lịch giữa một trạm gửi và một nhóm trạm nhận, thường được sử dụng trong việc gửi các thông .báo báo động.
- Kiểu *Publisher/subscriber:* Giao tiếp lập lịch giữa một trạm gửi *(publisher)* và nhiều trạm nhận *(subscriber),* dữ liệu được cập nhật mang tính toàn cục như nằm trong một vùng nhớ chung cho toàn bộ mạng.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Các dịch vụ giao tiếp

Fieldbus Message Specification (FMS)

Các dịch vụ FMS cho phép các chương trình ứng dụng gửi thông báo cho nhau trên bus theo một chuẩn thống nhất về tập dịch vụ cũng như cấu trúc thông báo. Ngoại trừ một số dịch vụ báo cáo thông tin và sự kiện, hầu hết các dịch vụ FMS khác đều sử dụng kiểu VCR Client/server.

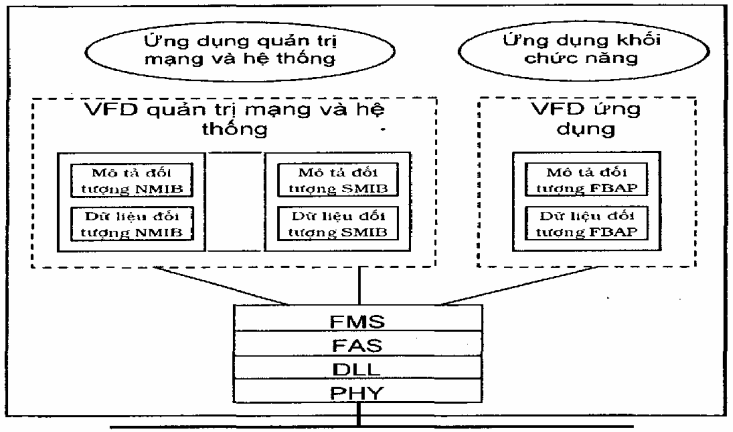
Dữ liệu cần trao đổi qua bus được biểu diễn qua một "Mô tả đối tượng *(object description)*. Các mô tả đối tượng wợc tập hợp thành một cấu trúc gọi là danh mục đối tượng *(object dictionary,* OD). Mỗi mô tả đối tượng được phân biệt qua chỉ số trong danh mục đối tượng.

Foundation Fieldbus – Các dịch vụ giao tiếp

Trong FMS, mô hình thiết bị trường ảo *(Virtual Field Device,* VFDđóng vai trò trung tâm. Một VFD là một đối tượng mang tính chất logic được sứ dụng để quan sát dữ liệu từ xa mô tả trong danh mục đối tượng

Một thiết bị thông thường có ít nhất hai VFD,

Foundation Fieldbus – Các dịch vụ giao tiếp



Foundation Fieldbus – Các dịch vụ giao tiếp

Quản lý mạng và hệ thống: Gồm hai phần

- Phần chính: cung cấp các chức năng cơ bản mà từ đó CT ứng dụng có thể xây dưng lên.
- Phần tiện ích: Cung cấp các dịch vụ tối ưu hoá hoạt động và chuẩn đoán các vấn đề xảy ra với mạng.

Foundation Fieldbus – Các dịch vụ giao tiếp

Phần chính bao gồm:

- Gán tên vật lý cho thiết bị
- Phân địa chỉ cho thiết bị.
- Các khối chức năng có liên quan.
- Đồng bộ hoá đồng hồ hệ thống.
- Lập danh mục các quá trình điều khiển phân tán

Foundation Fieldbus – Các dịch vụ giao tiếp

Các khối chức năng: các khối chức năng này dùng để cấu trúc nên các ứng dụng đo và điều khiển.

+ Khối chức năng;
  - Khối chức năng vào.
  - Khối chức năng ra.
  - Khối chức năng điều khiển.
  - Khối chức năng tính toán

+ Các bộ biến đổi;
  - Khối chức năng bộ biến đổi vào.
  - Khối chức năng bộ biến đổi ra.
  - Khối chức năng bộ biến đổi hiển thị.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Foundation Fieldbus – Các dịch vụ giao tiếp

+ Khối đối tượng vật lý;
  - Cảnh báo
  - Sự kiện.
  - Trend
  -Danh sách hiển thị

Foundation Fieldbus – Các dịch vụ giao tiếp

- Mô hình khối cho phép người dùng sử dụng các khối để cấu trúc nên các ứng dụng.
- Trong FF nó là các khối phần mềm nằm trong thiết bị.
- Trong FF các khối cung cấp phần lớn các chức năng cho các hệ thống điều khiển.
- Người dùng có thể cấu trúc nên hệ ĐK bằng cách liên kết các khối chức năng.

Mạng máy tính
&
Hệ thống thông tin công nghiệ

Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

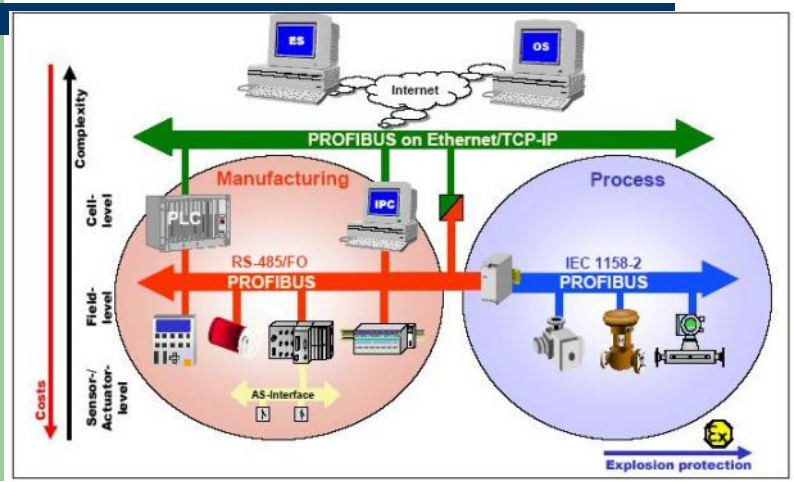
Profibus – Lịch sử phát triển

- PROFIBUS (*Process Field Bus)* là một hệ thống bus trường được phát triển tại Đức từ năm 1987 do 21 công ty và cơ quan nghiên cứu hợp tác. Sau khi được chuẩn hóa quốc gia với DIN 19245, PROFIBUS đã trở thành chuẩn châu âu EN 50 170 trong năm 1996 và chuẩn quốc tế IEC 61158 vào cuối năm 1999. Bên cạnh đó, PROFIBUS còn được đưa vào trong chuẩn IEC 61784 - một chuẩn mở rộng trên cơ sở IEC 61158 cho các hệ thống sản xuất công nghiệp. Với sự ra đời của các chuẩn mới IEC 61158 và IEC 61784 cũng như với các phát triển mới gần đây, PROFIBUS không chỉ dừng lại là một hệ *thống truyền* thông, mà còn được coi là một *công* nghệ *tự động hóa.*

Profibus – Lịch sử phát triển

- Với mục đích quảng bá cũng như hỗ trợ việc phát triển và sử dụng các sản phẩm tương thích PROFIBUS, một tổ chức người sử dụng đã được thành lập, mang tên *PROFIBUS Nutzerorgamsation* (PNO). Từ năm. 1995, tổ chức này nằm trong một hiệp hội lớn mang tên *PROFIBUS International* (PI) với hơn 1.100 thành viên trên toàn thế giới.

Profibus – Kỹ thuật

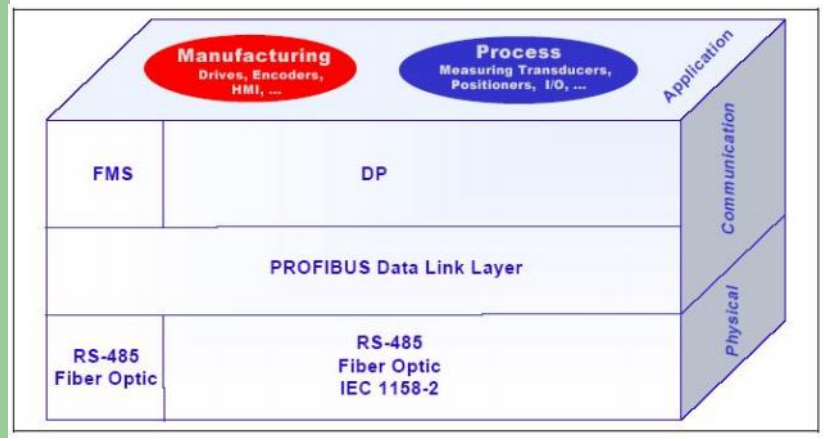


Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Kỹ thuật

- Profibus là một chuẩn bus trường mở, không phụ thuộc vào nhà cung cấp, nó được sử dụng trong một phạm vi rộng các ứng dụng trong tự động hoá sản xuất và tự động hoá quá trình.
- Sự không phụ thuộc vào các nhà cung cấp và tính chất mở được đảm bảo bởi tiêu chuẩn quốc tế EN 50170 và EN 50254. PROFIBUS cho phép giao tiếp giữa các thiết bị của các hãng sản xuất khác nhau mà không cần sự điều chỉnh đặc biệt nào về giao diện.
- PROFIBUS có thể dùng cho cả ứng dụng đòi hỏi tính năng thời gian với tốc độ cao và các nhiệm vụ truyền thông phức tạp.
- Qua sự tiếp tục phát triển về kỹ thuật, PROFIBUS sẽ vẫn là hệ thống giao thức công nghiệp được dùng trong tương lai.

Profibus – Kỹ thuật



Profibus – Kỹ thuật

- PROFIBUS định nghĩa ba loại giao thức là PROFIBUS-FMS, PROFIBUS-DP và PROFIBUS-PA.
- FMS là profile giao tiếp đa năng cho tất cả các đòi hỏi về giao tiếp cấp cao. FMS đưa ra nhiều chức năng ứng dụng tinh vi cho sự giao tiếp giữa các thiết bị thông minh. Tuy nhiên gần đây, vai trò của PROFIBUS-FMS ngày càng mờ nhạt bởi sự cạnh tranh của các hệ dựa trên nền Ethernet (Ethernet/IP, PROFINET, High-speed Ethernet).

Profibus – Kỹ thuật

- DP là giao thức truyền thông được sử dụng thường xuyên nhất. Nó được dùng tối ưu cho tốc độ, hiệu quả và chi phí kết nối thấp, được thiết kế đặc biệt cho sự giao tiếp giữa hệ thống điều khiển và các ngoại vi phân tán. DP thích hợp để thay thế cách truyền tín hiệu song song kiểu thông thường với điện áp 24 V trong tự động hoá sản xuất cũng như cho tín hiệu tương tự truyền với 4…20 mA hoặc Hart trong điều khiển quá trình.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

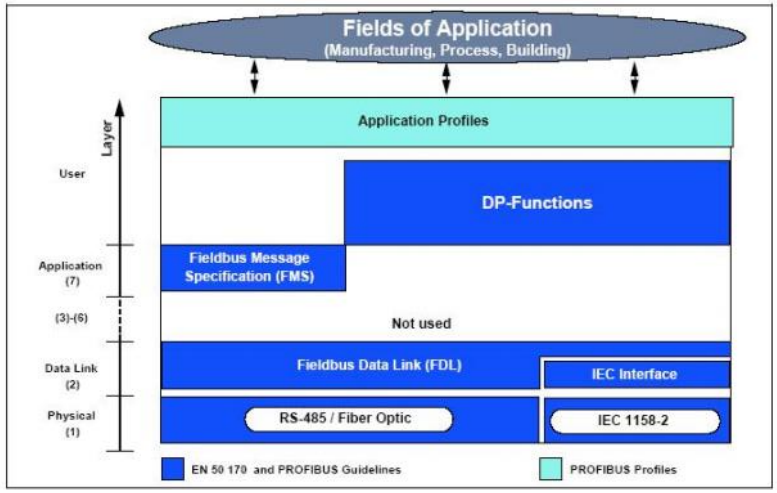
Profibus – Kỹ thuật

- PROFIBUS-PA là kiểu đặc biệt được sử dụng ghép nối trực tiếp các thiết bị trường trong các lĩnh vực tự động hóa các quá trình có môi trường dễ cháy nổ, đặc biệt trong công nghiệp chế biến. Thực chất, PROFIBUS-PA chính là sự mở rộng của PROFIBUS-DP xuống cấp trường cho lĩnh vực công nghiệp chế biến.

Profibus – Kỹ thuật

- Ngày nay. PROFIBUS là hệ bus trường hàng đầu thế giới với hơn 20% thị phần và với hơn 5 triệu thiết bị lắp đặt trong khoảng 500.000 ứng dụng. Có thể nói. PROFIBUS là giải pháp chuẩn, đáng tin cậy cho nhiều phạm vi ứng dụng khác nhau, đặc biệt là các ứng dụng có yêu cầu cao về tính năng thời gian.

Profibus – Kiến trúc giao thức



Profibus – Kiến trúc giao thức

- DP và PA, đây là giao thức giao tiếp có hiệu suất cao, sử dụng các lớp 1 và 2 cũng như lớp giao diện sử dụng. Các lớp từ 3 đến 7 không được sử dụng. Kiến trúc tổ chức hợp lý này đảm bảo truyền dữ liệu nhanh và hiệu quả. Lớp ánh xạ ở trên lớp 7 liên kết với lớp 2 (DDLM) cung cấp giao diện sử dụng dễ dàng truy nhập vào lớp 2. Các chức năng ứng dụng có sẵn cho người sử dụng, cũng như hành vi thiết bị và hệ thống của các kiểu thiết bị DP khác nhau được định rõ trong lớp giao diện sử dụng.

Profibus – Kiến trúc giao thức

- Trong giao thức truyền thông đa chức năng FMS, sự quan trọng đặc biệt nằm ở các lớp 1, 2 và 7. Lớp application (7) bao gồm hai lớp con là FMS (Fieldbus Message Specification) và LLI (Lower Layer Interface). Lớp FMS đảm nhiệm việc sử lý giao thức sử dụng và các dịch vụ truyền thông cho các giao tiếp chủ-chủ và chủ-tớ. Lớp LLI có vai trò trung gian cho FMS kết nối với lớp 2 mà không phụ thuộc vào các thiết bị riêng biệt.

Profibus – Lớp vật lý

- Truyền dẫn RS-485 là công nghệ truyền dẫn được sử dụng thông dụng nhất trong PROFIBUS. Phạm vi ứng dụng bao gồm tất cả các phạm vi truyền dẫn yêu cầu tốc độ truyền cao và lắp đặt đơn giản, giá thành rẻ. Cáp dẫn được sử dụng là đôi dây xoắn có bảo vệ.
- Công nghệ truyền dẫn với RS-485 dễ sử dụng. Việc lắp đặt các cáp xoắn không yêu cầu hiểu biết nhiều về chuyên môn. Cấu trúc của bus cho phép việc thêm và bớt các trạm hoặc từng bước đưa hệ thống hoạt động mà không bị ảnh hưởng của các trạm khác. Sự mở rộng sau không làm ảnh hưởng đến các trạm đang hoạt động.

Profibus – Lớp vật lý

- Tốc độ truyền từ 9,6 kbps đến 12 Mbps.
- Chiều dài tối đa 1200m và phụ thuộc vào tốc độ truyền

Tốc độ truyền phụ thuộc vào độ dài cáp:

Baud rate (kbit/s)	9.6	19.2	93.75	187.5	500	1500	12000
Range/Segment	1200 m	1200 m	1200 m	1000 m	400 m	200 m	100 m

Table 2: Range based on transmission speed for type-A cable

Profibus – Lớp vật lý

- Cấu trúc mạng dạng Bus (Trunk-line/Drop-line, daisy-chain).
- Cáp dùng đôi dây soắn có vỏ bảo vệ (PI khuyến cáo dùng cáp loại A có các thông số như sau:

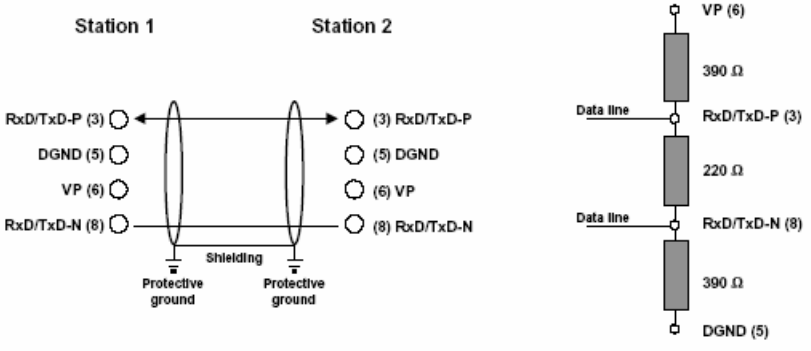
* Trở kháng : 125 đến 165 Ω
* Điện dung : < 30 pf/m
* Trở vòng : 110 Ω/km
* Tiết diện dây dẫn : > 0.34 mm2)

SimpoPDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Lớp vật lý

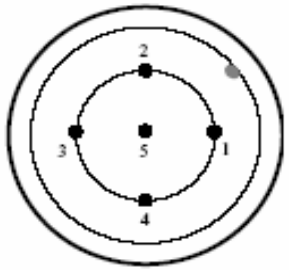
- Điện trở kết thúc dạng Fail-safe Biasing:



Profibus – Lớp vật lý

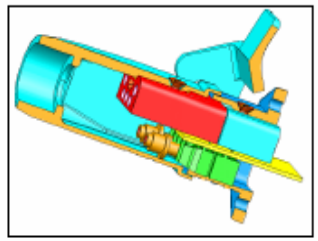
- Số trạm tối đa trên 1 đoạn mạng là 32. Có thể dùng tối đa 9 bộ Repeater -> 10 đoạn mạng. Tổng số trạm là 126.
- Chế độ truyền không đồng bộ, Half-duplex.
- Sử dụng mã NRZ.
- Không định nghĩa đầu nối cơ học.

Profibus – Lớp vật lý



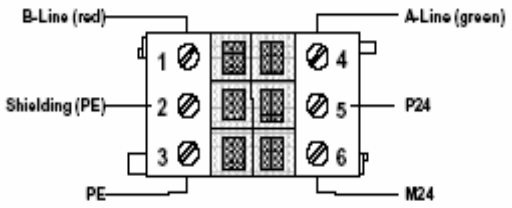
M12 Connector for RS-485 in IP65/67
Pin assignment: 1: VP, 2: RxD/TxD-N
3: DGND; 4: RxD/TxD-P; 5: Shield

Profibus – Lớp vật lý



Han-Brid Connector in Cu-Fo Version
for transmission of data via the fibers and 24 volt power supply for the peripherials in a single connector. This connector is also available in Cu/Cu version.

Profibus – Lớp vật lý



Siemens-Hybrid-Connector for transmission of both 24 volt power supply and PROFIBUS data via copper wires for devices with IP 65 protection.

Profibus – Lớp vật lý

- Cáp quang thích hợp đặc biệt trong các lĩnh vực ứng dụng có môi trường làm việc nhiễu mạnh, hoặc đòi hỏi tốc độ truyền dẫn cực cao và phạm vi phủ mạng lớn.
- Có nhiều loại cáp quang khác nhau cùng với các đặc tính khác nhau đựa trên khoảng cách, giá thành và ứng dụng.
- Hai loại cáp quang có thể sử dụng ở đây: loại sợi thuỷ tinh với chiều dài tối đa 2-3 km và loại sợi nhân tạo với chiều dài tối đa 50m không cần khuyếch đại.

Profibus – Lớp vật lý

- Do đặc điểm liên kết điểm-điểm ở cáp quang, cấu trúc mạng chỉ có thể là hình sao hoặc hạn hữu là mạch vòng. Trong thực tế, cáp quang thường được sử dụng hỗn hợp với RS-485 nên cấu trúc mạng phức tạp hơn. Các bộ chuyển đổi giữa 485 và cáp quang cho phép việc kết nối hỗn hợp.

Profibus – Lớp vật lý

- Truyền đồng bộ trong IEC 1158-2 (MPB) với tốc độ bốt 31.25 kbít/s được sử dụng trong các hệ thống xử lý tự động. Nó thỏa mãn các yêu cầu quan trọng trong công nghệ hoá học và công nghệ hoá dầu: sự an toàn bên trong và cấp nguồn qua bus sử dụng hai dây. Do đó PROFIBUS có thể được sử dụng trong các các khu vực nguy hiểm.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Lớp vật lý

Các nguyên tắc kết nối với IEC1158-2 :

- Mỗi đoạn mạng chỉ được phép có một bộ nguồn cung cấp điện
- Không có năng lượng được cung cấp cho bus khi các trạm đang gửi tin
- Mọi thiết bị trường tiêu thụ dòng không đổi tại trạng thái tĩnh
- Mỗi thiết bị trường hoạt động như một bộ tiêu hao dòng bị động.
- Mỗi đầu cuối được kết thúc bằng một trở đầu cuối bị động.
- Cấu trúc mạng ở đây là cấu trúc đường thẳng, cây hoặc sao.

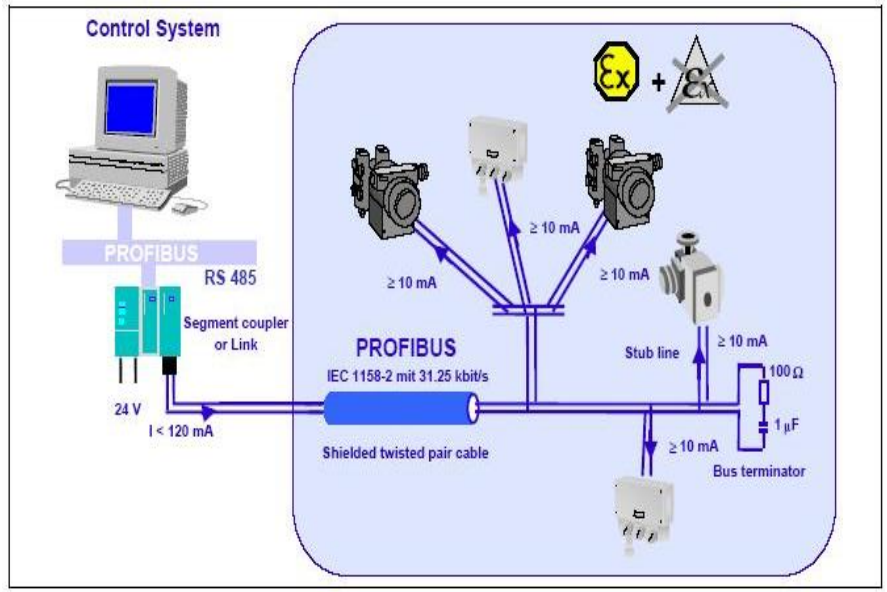
Profibus – Lớp vật lý

- Chế độ truyền	Số, đồng bộ bit, dùng mã Manchester
- Tốc độ truyền	31,25 kbit/s
- Cáp truyền	Hai đôi dây xoắn
- Cung cấp nguồn từ xa dữ liệu	Tuỳ chọn, sử dụng đường dây tải
- Mức bảo vệ cháy nổ	EEX ia/ib và EEX d/m/p/q
Cấu trúc mạng	Đường thẳng, cây hoặc phối hợp
Số trạm	Tối đa 32 trong một đoạn mạng, tổng tối đa 126
Số bộ lặp	Tối đa là 4 bộ lặp
Độ dài 1 đoạn mạng	1900m tổng 9500m

Profibus – Lớp vật lý

- Cáp bus chính được lắp ráp tại hai đầu cùng với trở đầu cuối bị động, bao gồm một bộ RC nối tiếp nhau với R=100 Ω và C = 1µF.
- Các bộ nối đoạn mạng là các bộ chuyển đổi tín hiệu, nó thích ứng các tín hiệu RS-485 với tín hiệu IEC 1158-2. Nếu các bộ nối được sử dụng, tốc độ trong đoạn mạng của RS-485 được giới hạn lớn nhất là 93.75 kbit/s
- Các bộ liên kết, chúng miêu tả tất cả các thiết bị trường kết nối trong đoạn mạng IEC 1158-2 như là một trạm tớ trong đoạn mạng RS-485. Không có giới hạn về tốc độ trong đoạn mạng RS-485 khi sử dụng để liên kết. Điều đó có nghĩa rằng nó cũng có thể thực hiện với các mạng nhanh, với chức năng điều khiển, bao gồm các cả thiết bị trường kết nối bằng IEC 1158-2.

Profibus – Lớp vật lý



Profibus – Điều khiển truy nhập Bus

- Điều khiển truy nhập trung gian (MAC) xác định thủ tục khi một trạm cho phép truyền dữ liệu. MAC phải chắc chắn rằng chỉ có một trạm có quyền truyền dữ liệu tại mỗi thời điểm.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Điều khiển truy nhập Bus

- Giao thức PROFIBUS được thiết kế để thoả mãn hai yêu cầu của MAC, đó là:

  * Trong lúc giao tiếp giữa các hệ thống tự động hoá phức tạp (các trạm chủ – masters) mỗi trạm phải có đủ thời gian để thực hiện công việc truyền thông trong một khoảng thời gian chính xác nhất định.

  * Mặt khác, đối với truyền thông giữa một bộ điều khiển khả trình phức tạp và thiết bị ngoại vi đơn giản được chỉ định (các trạm tớ – slaves) thì việc truyền dữ liệu một cách tuần hoàn và tính năng thời gian thực cần phải được thực hiện càng nhanh và càng đơn giản càng tốt.

Profibus – Điều khiển truy nhập Bus

- Do vậy, giao thức truy nhập trung gian_PROFIBUS bao gồm cả truy cập kiểu thẻ bài khi các trạm chủ giao tiếp với nhau và truy cập kiểu chủ/tớ khi trạm chủ giao tiếp với thiết bị ngoại vi đơn giản.

- Truy cập kiểu thẻ bài đảm bảo quyền truy nhập bus (thẻ bài) được chỉ định cho mỗi trạm chủ trong một khung thời gian xác định chính xác. Thông tin trong thẻ bài, một bản tin đặc biệt để việc chuyển thẻ bài từ trạm chủ này sang trạm chủ khác phải đi theo một vòng logic một lần tới tất cả các trạm chủ trong một thời gian luân chuyển thẻ bài cực đại. Trong PROFIBUS truy cập kiểu thẻ bài chỉ dùng cho giao tiếp giữa các trạm chủ.

Profibus – Điều khiển truy nhập Bus

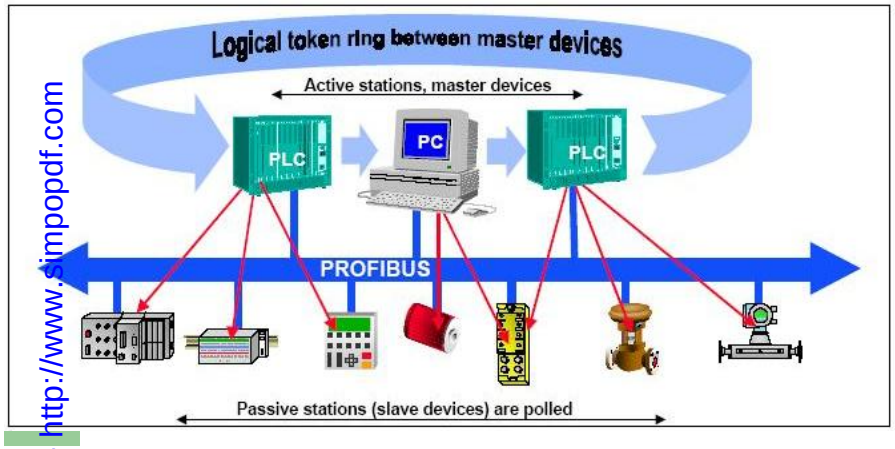
- Truy cập kiểu chủ/tớ cho phép trạm chủ (trạm tích cực) đang giữ thẻ bài được quyền truy nhập các trạm tớ được chỉ định (các trạm bị động). Nó cho phép trạm chủ có thể gửi bản tin hay khôi phục bản tin từ các trạm tớ.

- Phương pháp truy nhập này cho phép thực hiện với các kiểu cấu trúc hệ thống sau:

  Hệ thống một trạm chủ

  Hệ thống nhiều chủ

Profibus – Điều khiển truy nhập Bus



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Dịch vụ truyền số liệu

- Một nhiệm vụ quan trọng của lớp thứ 2 là bảo vệ dữ liệu. Định dạng khung lớp thứ 2 PROFIBUS đảm bảo độ toàn vẹn dữ liệu rất cao. Tất cả các bản tin đều có một khoảng cách Hamming HD=4, thông qua các dấu tách bắt đầu và kết thúc bản tin, bit chẵn lẻ và kiểm tra byte.

Profibus – Dịch vụ truyền số liệu

- Lớp thứ 2 của PROFIBUS hoạt động trong một chế độ không nối. Ngoài việc truyền dữ liệu cùng cấp, nó cung cấp giao tiếp nhiều đích (Broadcast và Multicast).
- Giao tiếp Broadcast có nghĩa rằng một tram chủ tích cực gửi một tin không phản hồi tới tất cả các trạm (chủ và tớ).
- Giao tiếp Multicast có nghĩa rằng một trạm chủ tích cực gửi một tin không phản hồi tới một nhóm các trạm đã được định trước (chủ và tớ).

Profibus – Dịch vụ truyền số liệu

- Mỗi Profile giao tiếp trong profibus sử dụng một tập hợp riêng biệt các dịch vụ của lớp 2. Các dịch vụ này được gọi bởi các lớp cao hơn thông qua các điểm truy nhập dịch vụ (SAPs). Trong FMS các điểm truy nhập dịch vụ được dùng để đánh địa chỉ cho các quan hệ giao tiếp logic.

SDA-Send Data With Acknowledge

SRD-Send Data Request Data With Reply

SDN-Send Data With No Acknowledge

CSRD- Cyclic Send And Request Data With Reply

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – Cấu trúc bức điện

- Khung với chiều dài thông tin cố định, không mang dữ liệu:

SD1	DA	SA	FC	FCS	ED

- Khung với chiều dài thông tin cố định, mang 8 byte dữ liệu:

SD3	DA	SA	FC	DU	FCS	ED

- Khung với chiều dài thông tin khác nhau, với 1-246 byte dữ liệu:

SD2	LE	LEr	SD2	DA	SA	FC	DU	FCS	ED

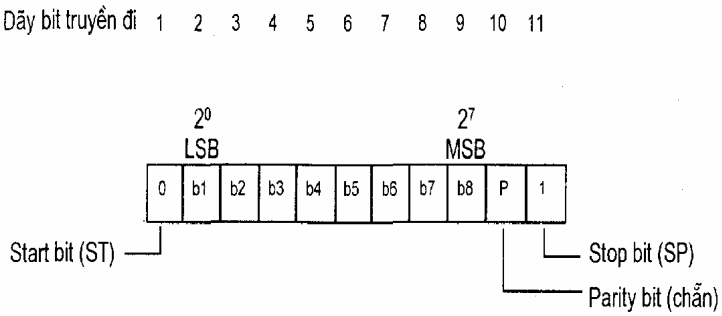
- Token:

SD4	DA	SA

Profibus – Cấu trúc bức điện

Ký hiệu	Tên đầy đủ	Ý nghĩa
SD1... SD4	Start Delimiter	Byte khởi đầu, phân biệt giữa các loại khung SD1 = 10H, SD2=68H, SD3 = A2H, SD4=DCH
LE	Length	Chiều dài thông tin (4-249 byte)
LEr	Length repeated	Chiều dài thông tin nhắc lại vì lý do an toàn
DA	Destination Address	Địa chỉ đích (trạm nhận), từ 0-127
SA	Source Address	Địa chỉ nguồn (trạm gửi), từ 0-126
DU	Data Unit	Khối dữ liệu sử dụng
FC	Frame Control	Byte điều khiển khung
FCS	Frame Check Sequence	Byte kiểm soát lỗi, HD = 4
ED	End Delimiter	Byte kết thúc, ED = 16H

Profibus – Cấu trúc bức điện



Profibus – Cấu trúc bức điện

Việc thực hiện truyền tuân thủ theo các nguyên tắc sau đây:

- Trạng thái bus rỗi tương ứng với mức tín hiệu của bit 1, tức mức tín hiệu thấp theo phương pháp mã hóa bit NRZ (0 ứng với mức cao).
- Trước một khung yêu cầu (*request frame*) cần một thời gian rỗi tối thiểu là 33 bit phục vụ mục đích đồng bộ hóa giữa hai bên gửi và nhận.
- Không cho phép thời gian rỗi giữa các ký tự UART của một khung.
- Với mỗi ký tự UART, bên nhận kiểm tra các bit khởi đầu, bit cuối và bit chẵn lẻ (parity chẵn). Với mỗi khung, bên nhận kiểm tra các byte SD, DA, SA, FCS, ED, LE/LEr (nếu có) cũng như thời gian rỗi trước mỗi khung yêu cầu. Nếu có lỗi, toàn bộ khung phải hủy bỏ.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - Cấu trúc bức điện

Trong trường hợp gửi dữ liệu với xác nhận (SDA), bên nhận có thể dùng một ký tự duy nhất SC=E5H để xác nhận. Ký tự duy nhất SC này cũng được sử dụng để trả lời yêu cầu dữ liệu (SRD) trong trường hợp bên được yêu cầu không có dữ liệu đáp ứng.

Profibus - FMS

Mặc dù PROFIBUS-FMS không được chuẩn hóa trong IEC 6158 và một phần vì thế vai trò của nó cũng mờ nhạt dần trong các phát triển tiếp theo, ứng dụng của nó đã có một vai trò nhất định trong một số lĩnh vực công nghiệp chế tạo, lắp ráp. Sử dụng PROFIBUS-FMS là bus hệ thống, các máy tính điều khiển có thể được ghép nối theo cấu hình nhiều chủ để giao tiếp với nhau và với các thiết bị trường thông minh dưới hình thức gửi các thông báo. ở đây, phạm vi chức năng, dịch vụ cao cấp là tính năng đợc coi trọng hơn so với thời gian phản ứng của hệ thống.

Profibus - FMS

Lớp ứng dụng của PROFIBUS-FMS bao gồm hai lớp con là FMS và LLI *(Lower Layer Interface)*. Bởi các lớp từ 3 đến 6 không xuất hiện ở đây. lớp LLI có vai trò thích ứng, chuyển dịch các dịch vụ giữa lớp FMS và lớp FDL lớp 2. Giao diện giữa FMS với các quá trình ứng dụng được thực hiện bởi lớp ALI *(Application Layer Interface)*.

Profibus - FMS

Giao tiếp hướng đối tượng

- PROFIBUS-FMS cho phép thực hiện các hoạt động giao tiếp hướng đối tượng theo cơ chế Client/server. Ở đây, ý nghĩa của phương thức hướng đối tượng là quan điểm thống nhất trong giao tiếp dữ liệu, không phụ thuộc vào các đặc điểm của nhà sản xuất thiết bị hay của lĩnh vực ứng dụng cụ thể.
- Các phần tử có thể truy nhập được từ một trạm trong mạng, đại diện cho các đối tượng thực hay các biến quá trình được gọi là các đối tượng giao tiếp. Các thành viên trong mạng giao tiếp thông qua các đỗi tượng này.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - FMS

- Việc truy nhập các đối tượng có thể thực hiện theo nhiều cách khác nhau. Phương pháp hiệu quả nhất là sử dụng chỉ số đối tượng *(object* index), còn gọi là phương pháp định địa chỉ logic. Chỉ số có thể coi là căn cước của một đối tượng nội trong một thành viên của mạng, được biểu diễn bằng một số thứ tự 16 bịt. Nhờ vậy, các khung thông báo sẽ có chiều dài ngắn nhất so với các phương pháp khác.

- Một khả năng thứ hai là truy nhập thông qua tên hình thức của đối tượng, hay còn gọi là *tag.*

Profibus - FMS

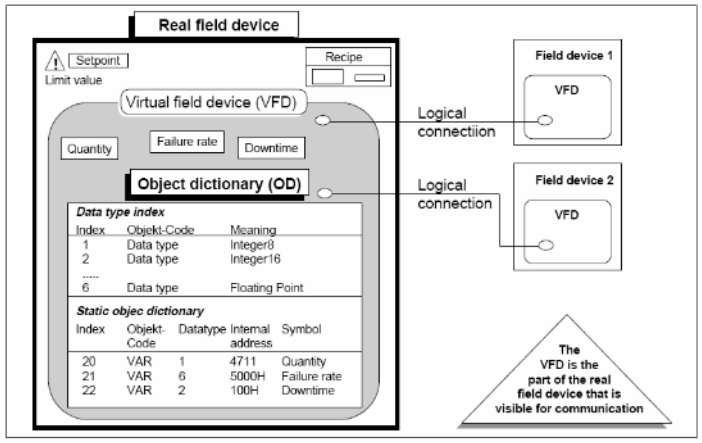
Thiết bị trường ảo (VFD)

- Thiết bị trường ảo *(virtual Field Device,* VFD) là một mô hình trừu tượng, mô tả các dữ liệu, cấu trúc dữ liệu và đặc tính của một thiết bị tự động hóa dưới giác độ của một đối tác giao tiếp.

- Một đối tượng VFD chứa tất cả các đối tượng giao tiếp và danh mục mô tả các đồi tượng mà các đối tác giao tiếp có thể truy nhập qua các dịch vụ. Một đối tượng VFD được sắp xếp tương ứng với đúng một quá trình ứng dụng.

- Một thiết bị thực có thể chứa nhiều đối tượng VFD, trong đó địa chỉ của mỗi đối tượng VFD được xác định qua các điểm đầu cuối giao tiếp của nó.

Profibus - FMS



Profibus - FMS

Đối tượng truyền thông

Đối tượng được mô tả thông qua các thuộc tính của đối tượng.

- Những đối tượng truyền thông tĩnh được đưa vào danh mục đối tượng tĩnh. Chúng được định dạng một lần và không bị thay đổi trong khi vận hành. FMS ghi nhận các kiểu đối tượng truyền thông tĩnh:

  * Biến đơn,
  * Ma trận (dãy các biến đơn của cùng một kiểu)
  * Bản ghi (dãy các biến đơn của các kiểu khác nhau)
  * Vùng nhớ (Domain):chỉ vùng nhớ có liên kết logic chứa chương trình hay dữ liệu.
  * Sự kiện (event) các thông báo, cảnh báo

Simpo PDF Merge and Split Unregistered Version - http://www.simopdf.com

Profibus - FMS

- Những đối tượng truyền thông động được đưa vào phần động của danh mục đối tượng và có thể bị thay đổi khi vận hành.
  * Danh sách biến (Variable List).
  * Program invocation.
- Định địa chỉ logic là một phương pháp được ưa dùng hơn để đánh địa chỉ cho các đối tượng. Việc truy cập được thực hiện bởi một địa chỉ ngắn (chỉ số) là một số kiểu không dấu hexa. Mỗi đối tượng chỉ có một chỉ số. Một mục chọn được thêm vào để định địa chỉ cho các đối tượng bằng tên.
- Các đối tượng truyền thông có thể được bảo vệ khỏi bị truy cập bởi những đối tượng không có quyền truy nhập thông qua sự bảo vệ truy cập, hay những dịch vụ được cho phép để truy cập một đối tượng (ví dụ chỉ được đọc) bị hạn chế.

Profibus - FMS

Quan hệ giao tiếp

- Ngoại trừ các hình thức gửi đồng loạt *(broadcast* và *multicast)*. việc trao đổi thông tin trong FMS luôn được thực hiện giữa hai đối tác truyền thông dới hình thức có nối theo cơ chế Client/server. Một client được hiểu là một chương trình ứng dụng (nói chính xác hơn là một quá trình ứng dụng) gửi yêu cầu để truy nhập các đối tượng. Còn một server chính là một chương trình cung cấp các dịch vụ truyền thông thông qua các đối tượng.

Profibus - FMS

Quan hệ giao tiếp

- Mối quan hệ giao tiếp giữa một client và một server được gọi là một kênh logic. Về nguyên tắc, một chương trình ứng dụng có thể đóng cả hai vai trò là client và server.
- Mỗi thành viên trong mạng có thể đồng thời có nhiều quan hệ giao tiếp với cùng một thành viên khác. hoặc với các thành viên khác nhau. Mỗi quan hệ giao tiếp được mô tả bởi một số các thông số trong một *communication reference (CR)*, bao gồm địa chỉ trạm đối tác *(remote addresss),* điểm truy nhập dịch vụ *(service access point,* SAP), các loại dịch vụ được hỗ trợ và chiều dài các bộ nhớ đệm.

Profibus - FMS

Các dịch vụ của FMS

- Các dịch vụ FMS là một tập con của các dịch vụ MMS (Manufacturing Message Specification, ISO9506) được tối ưu hoá cho các ứng dụng của bus trường và được mở rộng cho quản lí đối tượng truyền thông và quản lí mạng.

Các dịch vụ có xác nhận chỉ có thể được sử dụng cho các mối quan hệ truyền thông có kết nối định hướng.

Các dịch vụ không xác nhận chỉ được dùng trong các mối quan hệ truyền thông không kết nối (truyền broadcast và multicast). Chúng có thể được truyền với mức ưu tiên cao hoặc thấp.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - FMS

Các dịch vụ trong FMS được chia thành các nhóm sau:

- Dịch vụ **Variable Access** được dùng cho truy cập biến, bản ghi, ma trận hay danh sách biến.
- Dịch vụ **Domain Management** được dùng để truyền những vùng nhớ lớn. Dữ liệu phải được người dùng chia thành các phần nhỏ.
- Dịch vụ **Program Invocation Management** được dùng để điều khiển theo chương trình.
- Dịch vụ **Event Management** được dùng để truyền thông tin cảnh báo. Những thông tin này được gửi theo chế độ broadcast hay multicast.

Profibus - FMS

- Dịch vụ **VFD Support** được dùng để xác minh và thăm dò trạng thái. Chúng có thể được gửi đồng thời khi yêu cầu theo chế độ truyền multicast hay broadcast.
- Dịch vụ **OD Management** được dùng để đọc hay ghi khi truy cập vào danh mục đối tượng.
- Dịch vụ **Context Management** phục vụ cho việc thiết lập và kết thúc các kết nối logic.

Profibus - FMS

LLI (Lower Layer Interface)

- Liên hệ của lớp thứ 7 với lớp thứ 2 được thực hiện bởi LLI. Nhiệm vụ bao gồm điều khiển luồng dữ liệu và giám sát kết nối. Người dùng giao tiếp với các quá trình thông qua các kênh logic được gọi là các **mối quan hệ truyền thông.** LLI cung cấp nhiều kiểu mối quan hệ truyền thông để thực hiện FMS và các dịch vụ quản lí. Các mối quan hệ truyền thông có những khả năng kết nối khác nhau (ví dụ: quan sát, truyền dẫn, yêu cầu đối với đối tác truyền thông).

Profibus - FMS

- Các mối quan hệ truyền thông có kết nối thể hiện kết nối logic cùng cấp (peer-to-peer) giữa hai quá trình ứng dụng. Trước hết kết nối phải được thiết lập bởi dịch vụ khởi đầu trước khi có thể sử dụng cho truyền dữ liệu. Sau khi được thiết lập thành công, kết nối được bảo vệ khỏi bị truy cập bởi những đối tác không có quyền truy cập và sẵn sàng để truyền dữ liệu. Khi một kết nối không còn cần nữa thì nó được giải phóng. LLI cho phép giám sát kết nối điều khiển theo thời gian cho các mối quan hệ truyền thông có kết nối.

Profibus - FMS

- Các mối quan hệ truyền thông không kết nối cho phép một thiết bị giao tiếp đồng thời với một vài trạm sử dụng các dịch vụ không xác nhận. Trong các mối quan hệ truyền thông broadcast, một dịch vụ không xác nhận FMS được gửi đồng thời tới tất cả các trạm khác. Trong mối quan hệ truyền thông multicast, một dịch vụ không xác nhận FMS được gửi đồng thời tới một nhóm trạm đã định trước.Tất cả các mối quan hệ truyền thông của một thiết bị FMS được đưa vào một CRL.

Simpo PDF Merge and Split Unregistered Version - http://www.simopdf.com

Profibus - FMS

Quản lí mạng

- Ngoài các dịch vụ của FMS ra, các hàm quản lí mạng (**F**ieldbus **MA**nagement Layer 7 = FMA7) cũng có sẵn. Các hàm FMA7 là không bắt buộc và tuỳ theo sự cấu hình hoá trung tâm. Chúng có thể được khởi đầu riêng hay từ xa.

- **Quản lí theo ngữ cảnh** được dùng để thiết lập hay ngắt các kết nối FMA7.

- **Quản lí cấu hình hoá** được dùng để truy cập các CRL, biến, các bộ đếm số và các tham số của lớp1/2. Nó cũng được dùng cho sự xác minh và đăng kí của các trạm trên bus

Profibus - FMS

- **Quản lí lỗi** được dùng để biểu diễn lỗi/sự kiện và reset các thiết bị.

Nguyên tắc truy cập của các thiết bị cấu hình hoá đạt được bằng sự xác định của quản lí nối mặc định. Một kết nối quản lí mặc định phải được đưa vào với CREF=1 trong CRL đối với mọi thiết bị có hỗ trợ dịch vụ FMA7 như một bộ đáp ứng.

Profibus - DP

PROFIBUS DP được thiết kế hiệu quả cho việc trao đổi dữ liệu ở cấp trường. Những thiết bị điều khiển trung tâm, như PLC/PC hay các hệ thống điều khiển quá trình, giao tiếp thông qua một chuỗi kết nối với các thiết bị cấp trường phân tán như thiết bị vào ra, thiết bị truyền dẫn và các van, cũng như các bộ biến năng đo lường. Dữ liệu trao đổi với các thiết bị phân tán chủ yếu là tuần hoàn. Các chức năng truyền thông đòi hỏi ở đây được xác định bởi các chức năng DP cơ bản phù hợp với EN50-170. Ngoài các chức năng cơ bản, DP cũng đưa ra các dịch vụ truyền thông không tuần hoàn mở rộng để phục vụ cho việc tham số hoá, vận hành, theo dõi và cảnh báo của các thiết bị trường thông minh.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - DP

- DP cho phép hệ thống một trạm chủ hay nhiều trạm chủ. Nó cho độ mềm dẻo cao trong suốt cấu trúc hệ thống. Số trạm cực đại (cả trạm chủ và trạm tớ) nối trong mỗi bus là 126. Xác lập cấu hình hệ thống xác định số trạm, đánh địa chỉ cho mỗi trạm cho các địa chỉ I/O, tính nhất quán của dữ liệu I/O, định dạng bản tin chuẩn đoán lỗi và tham số bus sử dụng.
- Trong cấu hình nhiều trạm chủ các trạm chủ đếu có thể đọc dữ liệu ảnh đầu I/O từ trạm tớ nhưng chỉ có 1 trạm có quyền ghi dữ liệu đầu ra.

Profibus - DP

- Có 3 loại thiết bị khác nhau:

* DP master class 1 (DPM1)

Đây là một bộ điều khiển trung tâm trao đổi thông tin tuần hoàn với các trạm tớ theo một chu trình thông tin đã định trước. Thiết bị loại này điển hình là các PLC hay PC.

* DP master class 2 (DPM2)

Thiết bị loại này là các máy lập trình, công cụ cấu hình và thiết bị vận hành. Ngoài các dịch vụ của DPM1, các thiết bị này còn cung cấp các hàm đặc biệt phục vụ cho việc đặt cấu hình hệ thống, chuẩn đoán trạng thái, truyền nạp chương trình…

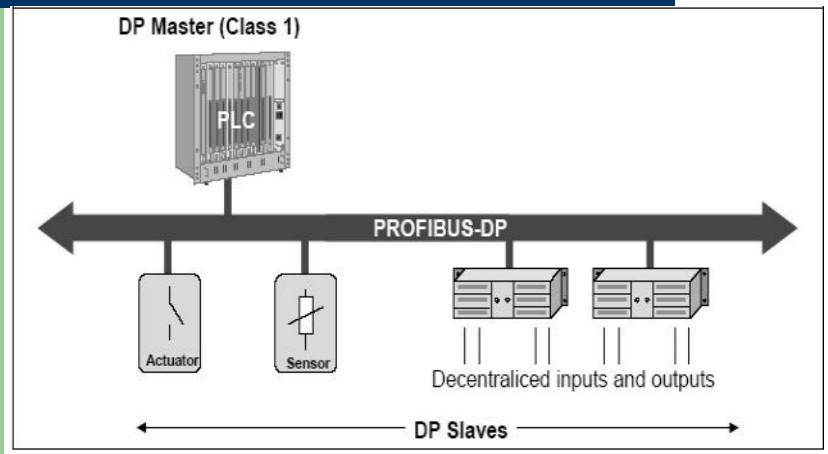
Profibus - DP

* DP slave:

Một DP slave là một thiết bị trường (thiết bị vào/ra, kích thích, van, bộ biến đổi đo lường…) thu thập thông tin ở đầu vào và gửi thông tin ra đầu ra tới các thiết bị ngoại vi. Cũng có những thiết bị chỉ cung cấp thông tin đầu vào hoặc đầu ra.

Lượng thông tin đầu vào và đầu ra phụ thuộc vào kiểu thiết bị. Cả thông tin đầu vào và thông tin đầu ra cực đại cho phép đều là 246 bytes

Profibus - DP



Profibus – DP

Đặc tính vận hành của hệ thống:

- Chuẩn DP bao gồm cả việc mô tả chi tiết hành vi của hệ thống để đảm bảo tính tương thích và khả năng thay thế lẫn nhau của các thiết bị. Hành vi của hệ thống được xác định trước hết qua trạng thái vận hành của các thiết bị DPM1.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – DP

- DPM1 có thể được điều khiển cục bộ hay qua bus bởi thiết bị cấu hình. Có 3 trạng thái chính:

*Stop: Trong trạng thái này, không có sự truyền dữ liệu giữa DPM1 và các trạm tớ. Chỉ có thể chuẩn đoán và tham số hoá.

*Clear: Trong trạng thái này, DPM1 đọc thông tin đầu vào của các trạm tớ và giữ các đầu ra ở giá trị an toàn.

*Operate: Trong trạng thái này, DPM1 ở chế độ trao đổi dữ liệu đầu vào và đầu ra tuần hoàn với các trạm tớ. Thông tin đầu vào của các trạm tớ được đọc và thông tin đầu ra được ghi cho các trạm tớ.

Profibus – DP

- DPM1 gửi một cách tuần hoàn thông tin về tình trạng của nó tới tất cả các trạm tớ đã chỉ định cho nó có sử dụng lệnh gửi đồng loạt vào các khoảng thời gian đặt trước.

- Phản ứng của hệ thống đối với một lỗi trong khi DPM1 ở chế độ trao đổi dữ liệu được xác định bằng tham số cấu hình auto-clear. Nếu tham số này được đặt là TRUE thì DPM1 sẽ khoá đầu ra của tất cả các trạm tớ được chỉ định tới trạng thái an toàn ngay khi một trạm tớ không sẵn sàng cho việc truyền dữ liệu sử dụng. Sau đó DPM1 chuyển sang trạng thái Clear.

Nếu tham số này là FALSE thì DPM1 giữ nguyên ở trạng thái Operate ngay cả khi có lỗi xảy ra và người dùng có thể xác định được phản ứng của hệ thống.

Profibus – DP

Trao đổi dữ liệu tuần hoàn

- Trao đổi dữ liệu giữa trạm chủ và các trạm tớ gán cho nó được thực hiện tự động theo một trình tự qui đinh sẵn. Khi đặt cấu hình hệ thống bus, người sử dụng định nghĩa các trạm tớ cho một thiết bị DPM1, qui định các trạm tớ tham gia và các trạm tớ không tham gia trao đổi dữ liệu tuần hoàn.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

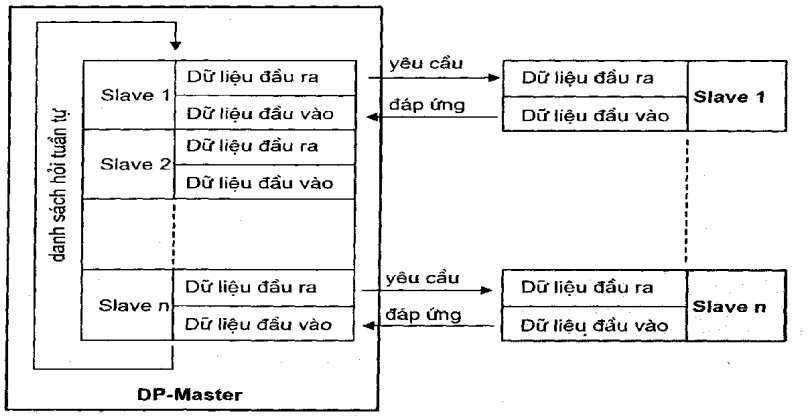
Profibus - DP

- Trước khi thực hiện trao đổi dữ liệu tuần hoàn, trạm chủ chuyển thông tin cấu hình và các tham số đã được đặt xuống các trạm tớ. Mỗi trạm tớ sẽ kiểm tra các thông tin về kiểu thiết bị, khuôn dạng và chiều dài dữ liệu, số lượng các đầu vào/ra. Chỉ khi thông tin cấu hình đúng với cấu hình thực của thiết bị và các tham số hợp lệ thì nó mới bắt đầu thực hiện trao đổi dữ liệu tuần hoàn với trạm chủ.

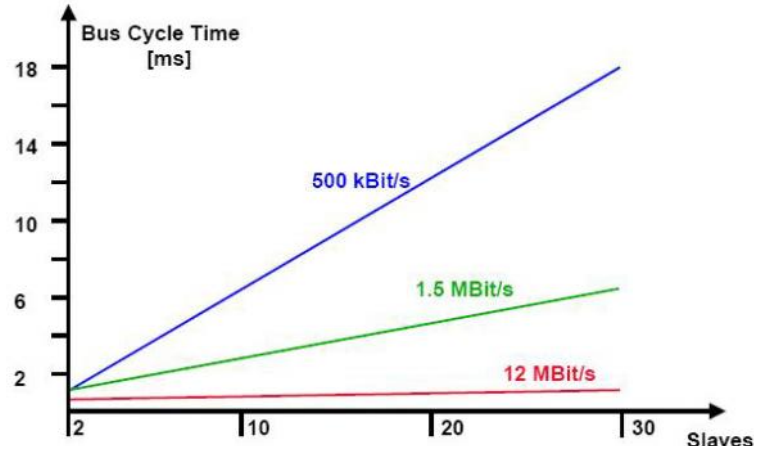
Profibus - DP

- Trong mỗi chu kỳ, trạm chủ đọc các thông tin đầu vào lần lượt từ các trạm tớ lên bộ nhớ đệm cũng như đưa các thông tin đầu ra từ bộ nhớ đệm xuống lần lượt các trạm tớ theo một trình tự qui định sẵn trong danh sách (polling list). Mỗi trạm tớ cho phép truyền tối đa 246 Byte dữ liệu đầu vào và 246 Byte dữ liệu đầu ra.
- Với mỗi trạm tớ, trạm chủ gửi một khung yêu cầu và chờ đợi một khung đáp ứng (bức điện trả lời hoặc xác nhận). Thời gian trạm chủ cần để xử lý một lượt danh sách hỏi tuần tự chính là chu kỳ bus.

Profibus - DP

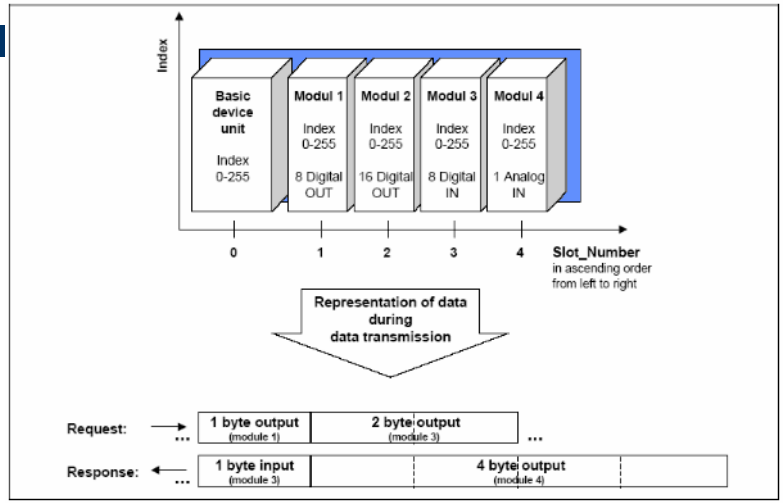


Profibus - DP



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - DP



Profibus - DP

Đồng bộ hoá dữ liệu vào ra:

- Trong truyền dữ liệu sử dụng do trạm chủ thực hiện tự động, trạm chủ có thể gửi lệnh điều khiển tới một trạm tớ đơn lẻ, một nhóm trạm tớ hay tới tất cả các trạm tớ một cách đồng thời. Những lệnh điều khiển này được truyền như một lệnh gửi đồng loạt. Chúng cho phép sử dụng chế độ Sync và Freeze cho việc đồng bộ hoá điều khiển theo sự kiện của các trạm tớ.

Profibus - DP

- Đầu ra của tất cả các trạm tớ đã đánh địa chỉ được giữ nguyên ở trạng thái hiện tại. Trong suốt quá trình truyền dữ liệu tiếp theo, dữ liệu đầu ra được lưu trữ tại các trạm tớ, nhưng các trạng thái đầu ra được giữ không đổi. Dữ liệu đầu ra được lưu trữ sẽ không được gửi tới các đầu ra cho tới khi nhận được một lệnh Sync tiếp theo. Chế độ Sync kết thúc bằng một lệnh Unsync.

- Tương tự, một lệnh điều khiển Freeze sẽ chuyển các trạm tớ sang chế độ Freeze. Trong chế độ này, trạng thái của các đầu vào được giữ nguyên tại giá trị hiện tại. Dữ liệu đầu vào không được cập nhật cho đến khi trạm chủ gửi một lệnh Freeze tiếp theo. Chế độ Freeze kết thúc bằng một lệnh Unfreeze.

Profibus - DP

Chuẩn đoán hệ thống:

- Chức năng chuẩn đoán lỗi của DP cho phép định vị lỗi nhanh. Thông tin chuẩn đoán lỗi được truyền qua bus và được tập trung tại trạm chủ. Thông tin này được chia làm 3 cấp:

1. Mức trạm: Thông tin này có liên quan đến tình trạng vận hành chung của trạm, ví dụ như: quá nhiệt độ, điện áp thấp…

2. Chuẩn đoán module: Thông tin này thể hiện rằng trong một phạm vi nhất định của đầu vào/ra ví dụ như module đầu ra 8bit.. của một trạm, chuẩn đoán là chưa xác định.

3. Chuẩn đoán kênh: Trong trường hợp này, nguyên nhân lỗi được xác định trong mối liên hệ với từng bit vào/ra riêng biệt (kênh) ví dụ như ngắn mạch đầu ra 7.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – DP

Các chức năng sau đây có sẵn khi truyền dữ liệu không tuần hoàn giữa hệ thống điều khiển trung tâm (DPM1) và các trạm tớ:

MSAC1_Read: Trạm chủ đọc một khối dữ liệu từ trạm tớ.

MSAC1_Write: Trạm chủ ghi một khối dữ liệu tới trạm tớ.

MSAC1_Alarm: Thông tin cảnh báo truyền từ trạm tớ tới trạm chủ. Có bản tin báo nhận của trạm chủ. Chỉ sau khi thông tin báo nhận sự cảnh báo đã được nhận thì trạm tớ mới có thể gửi một thông tin cảnh báo mới. Điều đó có nghĩa rằng thông tin cảnh báo không bao giờ bị ghi đè.

Profibus – DP

MSAC1 nhận thông tin cảnh báo: Trạm chủ nhận gửi một thông tin cho trạm tớ với thông báo đã nhận được thông tin cảnh báo.

Trạng thái MSAC1: Thông tin về trạng thái được gửi từ trạm tớ tới trạm chủ. Và thông tin này không được báo nhận. Do vậy thông tin về trạng thái có thể bị ghi đè.

Profibus – DP

Các chức năng sau đây có sẵn để truyền dữ liệu không tuần hoàn giữa những công cụ lập trình và vận hành (DPM2) và các trạm tớ:

MSAC2_khởi đầu và MSAC2_Kết thuc: Thiết lập và kết thúc của các nối cho việc truyền dữ liệu không tuần hoàn giữa DPM2 và trạm tớ.

MSAC2_đọc: Trạm chủ đọc một khối dữ liệu từ trạm tớ.

MSAC2_ghi: Trạm chủ ghi một khối dữ liệu tới trạm tớ.

MSAC2_vận chuyển dữ liệu: Với dịch vụ này, trạm chủ có thể ghi dữ liệu một cách không tuần hoàn tới cáo trạm tớ và nếu yêu cầu thì cũng có thể đọc dữ liệu từ trạm tớ trong cùng một chu trình dịch vụ.

Profibus – DP

Giao tiếp trực tiếp giữa các trạm tớ

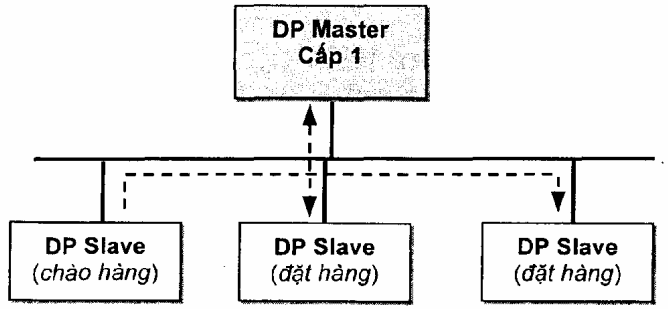
Trao đổi dữ liệu giữa các trạm tớ là một yêu cầu thiết thực đối với cấu trúc điều khiển phân tán thực sự sử dụng các thiết bị trường thông minh. Như ta đã biết, cơ chế giao tiếp chủ-tớ thuần túy làm giảm hiệu suất trao đổi dữ liệu cho trờng hợp này. Chính vì thế, phiên bản DP-V2 đã bổ sung một cơ chế trao đổi dữ liệu trực tiếp theo kiểu chào hàng đặt hàng giữa các trạm tớ.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - DP

Một trạm tớ (ví dụ một cảm biến) có thể đóng vai trò là "nhà xuất bản" hay "nhà cung cấp" dữ liệu. Khối dữ liệu sẽ được gửi đồng loạt tới tất cả các trạm tớ (ví dụ một van điều khiển, một biến tần) đã đăng ký với vai trò "người đặt hàng" mà không cần đi qua trạm chủ. Với cơ chế này, không những hiệu suất sử dụng đường truyền được nâng cao, mà tính năng đáp ứng của hệ thống còn được cải thiện rõ rệt.

Profibus - DP



Hình 4.8: Giao tiếp trực tiếp giữa các trạm tớ

Profibus - DP

Chế độ đẳng thời

Đối với một số ứng dụng như điều khiển truyền động điện, điều khiển chuyển động, cơ chế giao tiếp theo kiểu hỏi tuần tự hoặc giao tiếp trực tiếp tớ-tớ chưa thể đáp ứng được đòi hỏi cao về tính năng thời gian thực. Vì vậy, phiên bản DP-V2 bổ sung chế độ đẳng thời *(isochronous* mode), cho phép thực hiện giao tiếp theo cơ chế chủ/tớ kết hợp với TDMA.

Profibus - DP

Nhờ một thông báo điều khiển toàn cục gửi đồng loạt, toàn bộ các trạm trong mạng được đồng bộ hóa thời gian với độ chính xác tới micro-giây. Việc giaotiếp được thực hiện theo một lịch trình đặt trước, không phụ thuộc vào tải tức thời trên bus. Cơ chế này cho phép phối hợp hoạt động một cách chặt chẽ và nhịp nhàng giữa các trạm trên bus.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

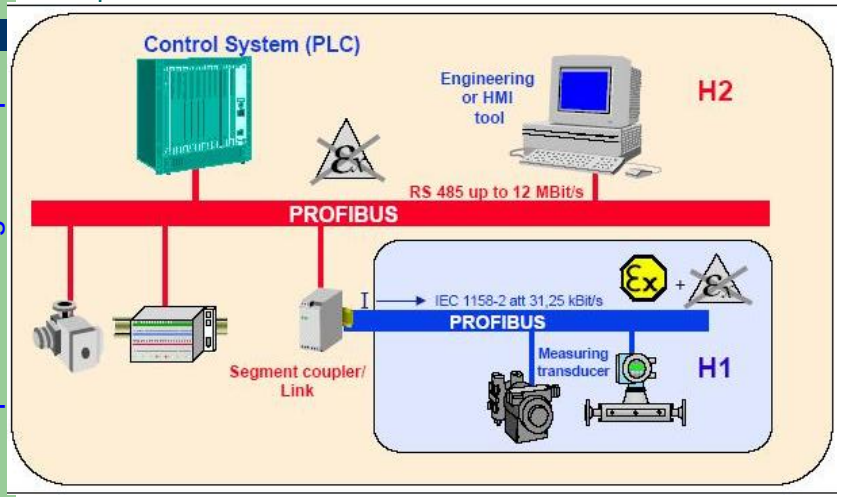
Profibus - PA

- PROFIBUS PA thực chất được xây dựng trên mô hình giao thức PROFIBUS DP với chuẩn truyền dẫn IEC 1158-2(MBP). và một số các thông số- đặc tính cho các thiết bị trường.
- Ưu điểm của nó là cho phép các thiết bị của những nhà sản xuất khác nhau có thể tương tác với nhau hoặc thay thế lẫn nhau.
- Những đặc tính hữu dụng của PROFIBUS PA khiến giao thức này có thể thay thế phương thức truyền tín hiệu với 4..20 mA hoặc HART.

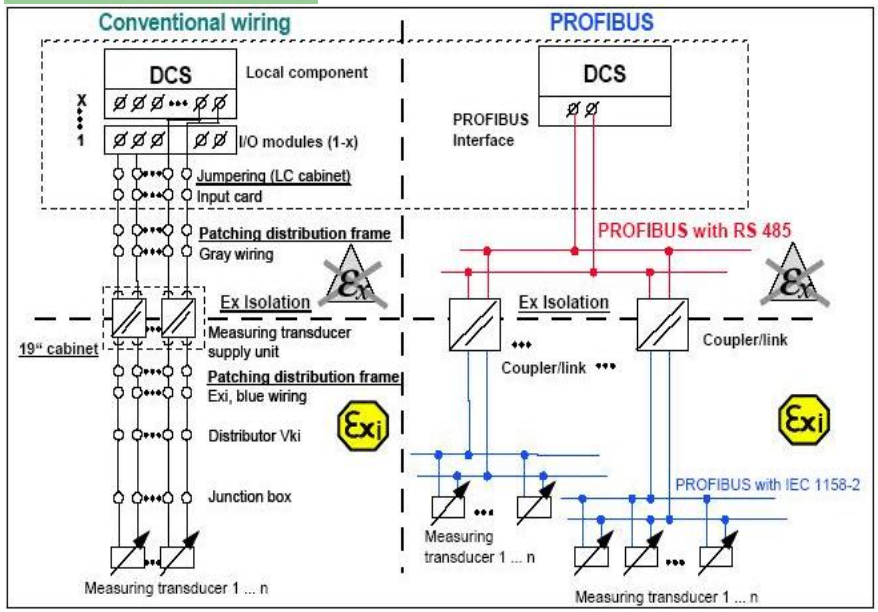
Profibus - PA

- Xét về mặt đặc tính hoạt động-chức năng, PROFIBUS PA dựa trên mô hình khối hàm chức năng(Function Block Model).
- PROFIBUS PA cho phép kết nối các hệ thống thiết bị trường bằng cáp đôi dây xoắn đơn giản, với tốc độ truyền cố định 31,5kbps.
- Khi sử dụng giao thức này, ta có thể bảo dưỡng, thay thế một số thiết bị nếu cần thiết trong khi đang vận hành. Đặc biệt, nó còn rất hữu ích khi sử dụng ở những khu vực nguy hiểm, dễ cháy nổ với phương thức bảo vệ kiểu "an toàn riêng" (EEX ia/ib) hoặc kiểu "đóng kín" (EEXd).

Profibus - PA



Profibus - PA



Profibus – PA

Giao diện Bus an toàn riêng:

- Trường thiết bị trong những vùng nguy hiểm được kết nối với công nghệ truyền dẫn sử dụng chuẩn 1158-2. Chuẩn này cho phép truyền dẫn thông tin cũng như năng lượng giữa các trường thiết bị chỉ với 2 dây dẫn.
- Khác với những phương thức truyền dẫn quen thuộc trước đây, PROFIBUS PA chỉ cần dùng một đường dây truyền dẫn tín hiệu từ những điểm đo đạc tới bộ I/O của hệ thống điều khiển. Với một nguồn công suất (nguồn chống nổ - nếu cần, ở những khu vực nguy hiểm), tín hiệu sẽ được truyền khắp mạng PROFIBUS tới những nơi yêu cầu.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus – PA

- Tuỳ vào mức độ cháy nổ của khu vực và sự tiêu tốn năng lượng của thiết bị, có từ 9 tới 32 bộ truyền tín hiệu đo đạc được kết nối trong mạng truyền thông.

Profibus – PA

Các yêu cầu cụ thể cho BUS an toàn riêng:

- Một đoạn mạng chỉ có một nguồn nuôi tích cực.
- Mỗi một trạm tiêu thụ dòng cố định(>=10mA) ở trạng thái xác lập.
- Mỗi một trạm được coi là tải tiêu thụ dòng thụ động(bỏ qua thành phần dung, cảm kháng).
- Một trạm khi phát tín hiệu 0 được nạp thêm nguồn.

Profibus – PA

- Các giá trị đo, biến trạng thái và biến điều khiển của giao thức PROFIBUS PA được truyền dẫn tuần hoàn với quyền ưu tiên cao tới DP Master (DPM1) thông qua các bộ DP cơ sở. Điều này đảm bảo các giá trị đo cũng như các biến trạng thái luôn luôn được cập nhật và luân chuyển kịp thời đến DP Master.
- Mặt khác, các thông số thiết bị không tuần hoàn như thông tin bảo dưỡng- chuẩn đoán, chế độ vận hành …được trao đổi tuần hoàn tới công cụ phát triển - engineering tool (DPM2) thông qua các hàm DP mở rộng kết nối với quyền ưu tiên thấp.

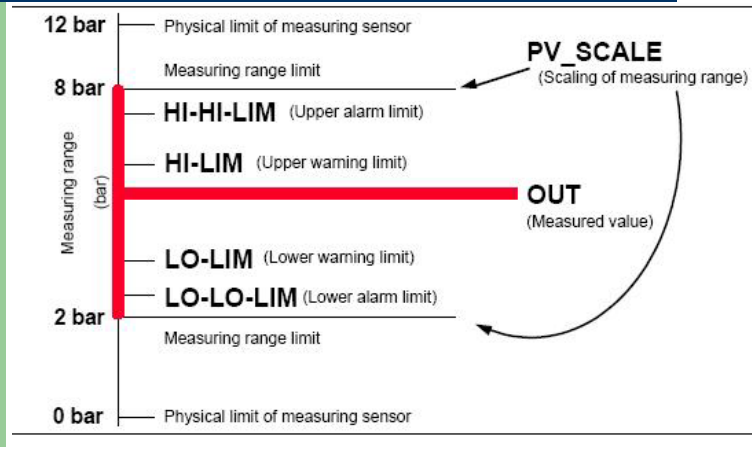
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Profibus - PA

Các thiết bị trường PA tuỳ theo các đặc tính (profile) có thể chia thành 2 nhóm :

- Đặc tính nhóm A (Profile class A) : quy định đặc tính và chức năng cho các thiết bị đơn giản như : bộ cảm biến áp suất, nhiệt độ các cơ cấu truyền động. Ta cũng có thể truy nhập các thông số hệ thống như tốc độ, thời gian trễ, ngưỡng cảnh báo vào mạng thông tin.
- Đặc tính nhóm B (Profile Class B): quy định chức năng, đặc tính cho các thiết bị phức hợp, còn gọi là các thiết bị thông minh. Các chức năng này của giao thức cho phép gán địa chỉ tự động, đồng bộ hoá thời gian , phân tán dữ liệu tới các bộ I/O phân tán, mô tả thiết bị qua ngôn ngữ DDL (Device Discription Language) cũng như lập lịch khối hàm (Function Block).

Profibus - PA



Profibus - PA

Giao thức PROFIBUS PA cho phép tương tác, thay thế lẫn nhau giữa các thiết bị của những nhà sản xuất khác nhau. Nó sử dụng mô hình khối hàm để mô tả chức năng tham số thiết bị. Mỗi khối hàm mô tả một chức năng sử dụng, chẳng hạn vào hoặc ra tương tự. Trên thực tế, hai khối hàm - tuỳ theo chức năng cụ thể, có thể một khối hàm vật lý và một khối chuyển đổi, được kết nối với nhau thông qua mối liên kết truyền thông của hệ tạo thành chương trình điều khiển.

Profibus - PA

Thường có những loại khối hàm sau đây :

- Khối hàm vật lý (Physical Block) : chứa những thông tin chung về thiết bị như tên, nhà sản xuất, chủng loại, mã số.
- Khối hàm chuyển đổi (Transducer Block) chứa những thông số kết nối giữa các thiết bị
- Khối đầu vào tương tự (Analog Input) : cung cấp giá trị đo được bởi cảm biến như các thông số trạng thái, nhiệt độ, áp suất.
-Khối đầu ra tương tự (AO) : cung cấp các giá trị tương tự ở đầu ra hệ thống điều khiển.

Profibus - PA

- Khối đầu vào số (Digital Input) : chứa giá trị đầu vào ở dạng số
- Khối đầu ra số (DO) : đưa ra thông số đầu ra của hệ thống điều khiển ở dạng số.

Trong mạng truyền thông, nhiều khối hàm được các nhà sản xuất tích hợp với nhau thông qua thiết bị trường, do đó ta có thể truy nhập vào hệ thống lấy ra các thông số, kết nối các khối hàm tạo nên trình ứng dụng giao thức PROFIBUS PA.

Thiết bị Profibus

Các thiết bị PROFIBUS có những đặc điểm cấu trúc khác nhau. Sự khác nhau về cấu trúc giữa chúng tuỳ thuộc vào chức năng của từng thiết bị (ví dụ như thiết bị truyền tín hiệu đầu vào/đầu ra, các bộ chẩn đoán) và phụ thuộc vào các tham số đường truyền như tốc độ truyền dữ liệu, các giá trị thời gian giám sát. Những tham số này thay đổi tuỳ theo từng loại thiết bị và hệ thống điều khiển. Để mạng truyền thông với giao thức PROFIBUS có cấu trúc đơn giản, các thiết bị thường sử dụng GSD files.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Thiết bị Profibus

- Tất cả các nhà sản xuất đều phải cung cấp file GSD trong các thiết bị PROFIBUS của mình.
- GSD files được ứng dụng rộng rãi, từ hệ thống truyền tin mở đến các hệ thống điều khiển vận hành. GSD files được dùng trên mọi cấu hình từ loại đơn giản nhất đến loại phức tạp nhất. Điều này có nghĩa là tích hợp giữa các thiết bị thuộc những nhà sản xuất khác nhau trong mạng PROFIBUS không là vấn đề khó khăn.

Thiết bị Profibus

- GSD files chứa những đặc điểm đặc trưng cơ bản giống nhau giữa các thiết bị PROFIBUS, đó chính là lý do vì sao GSD files tương thích được với nhiều loại thiết bị. Thông qua những file này, kỹ sư dự án không phải nắm bắt các thông số kỹ thuật theo cách đo đạc bằng tay thông thường như trước nữa. Thời gian được tiết kiệm đáng kể và trong suốt quá trình, hệ thống điều khiển sẽ tự động kiểm tra (check) các sai số đầu vào, sai số truyền dữ liệu và nhiều loại sai số khác.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Thiết bị Profibus

- Nhìn chung một file GSD bao gồm ba khu vực (sections) sau:

* Khu vực chứa thông tin chung: những thông tin chung chẳng hạn như tên thiết bị, mã đăng ký phần mềm, phần cứng, tốc độ truyền dữ liệu của đường truyền, thời gian giám sát.

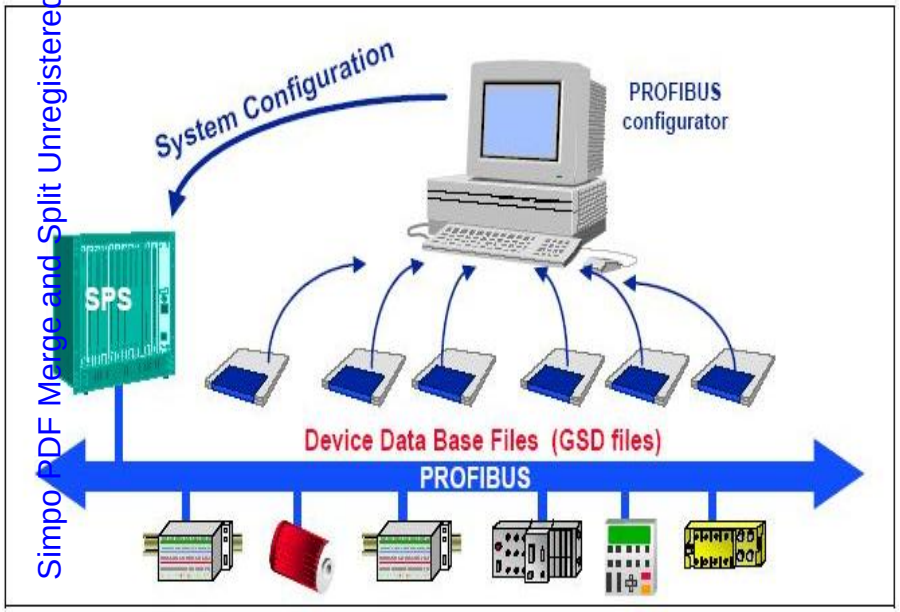
* Khu vực liên kết với trạm chủ (Master - related): chứa những tham số liên kết với trạm chủ, ví dụ số lượng lớn nhất các slave có thể kết nối… Khu vực này không có trong các thiết bị slave.

* Khu vực liên kết với trạm tớ (Slave - related): khu vực này chứa những thông tin liên quan đến trạm slave như số lượng và chủng loại đầu vào/ đầu ra, các thông tin chẩn đoán về các module thiết bị.

Thiết bị Profibus

- Các tham số của từng khu vực riêng biệt được tách biệt ra bởi các từ khoá - key words. Từ khoá A chỉ những tham số uỷ nhiệm (ví dụ tên hãng sản xuất), tham số lựa chọn (options) như mã số đồng bộ. Sự khác biệt từng nhóm tham số cho phép ta lựa chọn options được hiệu quả. Ngoài ra, các file bit map với những biểu tượng của thiết bị có thể được tích hợp. Dung lượng thông tin các file GSD có thể chứa rất lớn. Nó chứa thông số về tốc độ truyền dữ liệu cũng như cả không gian mô tả các module hữu ích trong các module thiết bị. Nó còn chứa cả các thông tin chẩn đoán.

Thiết bị Profibus



Thiết bị Profibus

Nhà cung cấp	Chip	Kiểu	Đặc tính	FMS	DP	Vi xử lý	Giao thức SW	Tốc độ (Mbit/s)
AGE	Agent-PB	Chủ/tớ	Chip giao thức đa chức năng dựa trên FPGA	•	•	•	•	12
IAM	PBM	Chủ	Chip giao thức ngoại vi	•	•	•	•	3
M2C	IX1	Chủ/tớ	Chip đơn hoặc chip giao thức ngoại vi	•	•	•	•	3
Siemens	SPC4	Tớ	Chip giao thức ngoại vi	•	•	•	•	12
Siemens	SPC3	Tớ	Chip giao thức ngoại vi	–	•	•	•	12
Siemens	DPC31	Tớ	Chip giao thức có vi xử lý	–	•	•	•	12
Siemens	ASPC2	Chủ	Chip giao thức ngoại vi	•	•	•	•	12
Siemens	SPM2	Tớ	Chip đơn có kết nối 64 I/O bits	–	•	–	–	12
Siemens	LSPM2	Tớ	Giá rẻ, Chip đơn có kết nối 32 I/O bits	–	•	–	–	12
PROFICHIP	VPC3+	Tớ	Chip giao thức ngoại vi		•	•	•	12
PROFICHIP	VPC LS	Tớ	Giá rẻ, Chip đơn có kết nối 32 I/O bits	–	•	–	–	12

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Thiết bị Profibus

Phương thức làm việc của bộ slave đơn giản

Đối với các thiết bị đầu vào/ đầu ra đơn giản, giải pháp PROFIBUS với AICs đơn chíp là một giải pháp thực tế. Tất cả các chức năng giao thức đã được tích hợp sẵn trong ASICs. Vì vậy không cần bộ vi xử lý hoặc phần mềm mà chỉ cần mạch giao diện truyền tin, tinh thể thạch anh và các thiết bị điện tử công suất đóng vai trò như các thiết bị ngoại vi. Thí dụ bộ slave điển hình bao gồm SPM2 ASIC của Siemens, chip IX1 của M2C và CHIP vpcls-asic của profichip.

Thiết bị Profibus

Phương thức làm việc của bộ slave thông minh

- Đối với bộ slave thông minh, bộ phận thời gian tới hạn của nó sẽ làm việc trên chip giao thức, các phần còn lại làm việc như phần mềm trong bộ vi điều khiển.

- Chip DDC31 của hãng Siemens là sự kết hợp của chip giao thức và bộ vi điều khiển. Còn những chip cơ sở khác, ví dụ như ASICs SPC3 (Siemens), VPC3+ (PROFICHIP) hay IX1 (M2C) thì đã được chế sẵn, chỉ cần lắp ráp. Những con chip ASIC cung cấp giao diện dùng chung với các bộ vi điều khiển. Ngoài ra, ở bộ slave thông minh ta còn có thể dùng các bộ vi xử lý với lõi đã được tích hợp giao thức PROFIBUS.

Thiết bị Profibus

Phương thức làm việc của bộ master phức hợp

- Cũng giống như bộ slave thông minh, ở bộ master phức hợp, bộ phận thời gian tới hạn của nó cũng làm việc trên chip giao thức, còn những phần còn lại làm việc như phần mềm trong bộ vi điều khiển. Các chip ASICs ASPC2 (Siemens), IX1 (M2C) hay VAF PBM (IAM) đều đã được chế sẵn để hỗ trợ các thiết bị master phức hợp hoạt động. Chúng cũng được kết hợp và cùng vận hành với các bộ vi xử lý.

Thiết bị Profibus

Phương thức làm việc theo chuẩn IEC 1158-2

- Đối với các thiết bị trường truyền công suất tuân theo chuẩn IEC 1158-2, một vấn đề cần lưu ý là công suất tiêu tốn phải thấp (vì đây là những thiết bị truyền công suất). Đối với những thiết bị loại này thường thường chỉ dùng nguồn dòng cỡ 10 mA là phù hợp cho việc nuôi các thiết bị đo đạc, cung cấp năng lượng truyền tin.

- Để đáp ứng được các yêu cầu trên, hãng Siemens và hãng Smar đã chế tạo ra những con chip đặc biệt phù hợp. Những con chip này sẽ lấy năng lượng cần thiết để vận hành toàn bộ thiết bị từ đường bus truyền theo chuẩn IEC 1158-2 và khiến đường truyền là nguồn cung cấp điện áp cho các thiết bị trường PA khác trong hệ thống

AS-Interface

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

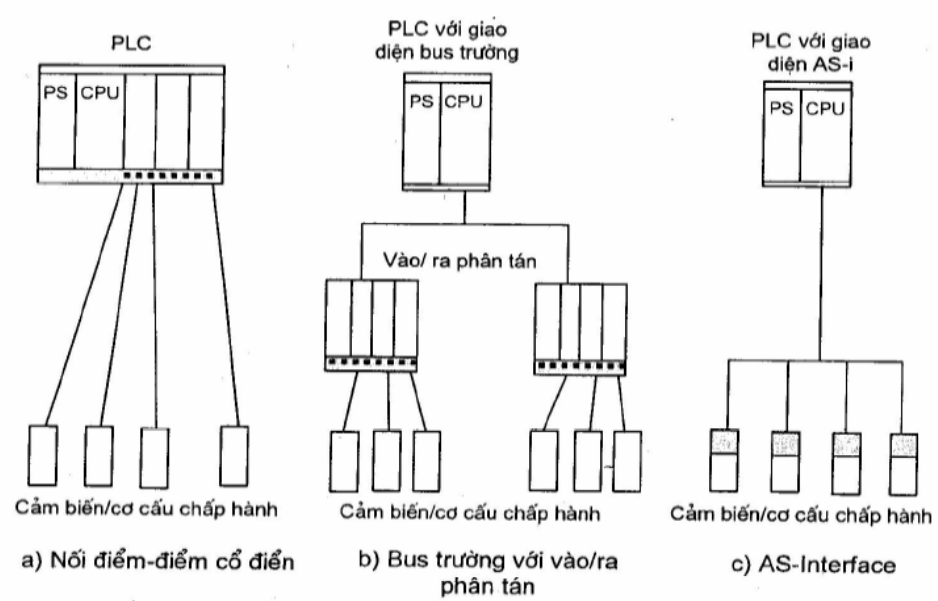
AS-i

- Khả năng đồng tải nguồn.
- Khả năng chống nhiễu cao nhưng không đòi hỏi chất lượng đường truyền dẫn tốt.
- Cấu trúc mạng đường thẳng hay cây.
- Các thành phần giao diện thực hiện với giá cả thấp.
- Bộ nối phải nhỏ , gọn, đơn giản, giá thành hợp lý.

AS-i

- AS-i : Actuator Sensor Interface
- Trong CN 80% các cảm biến và cơ cấu chấp hành làm việc với các biến logic.
- Nối mạng giữa chúng phải đảm bảo yêu cầu về giá thành, lắp đặt, vận hành, bảo dưỡng.

AS-i



AS-i Kiến trúc giao thức

- Giao tiếp giữa các bộ điều khiển và các thiết bị cảm biến, cơ cấu chấp hành.
- Trao đổi dữ liệu thuần tuý, số lượng nhỏ.
- Nâng cao hiệu suất, đơn giản thực hiện -> sử dụng lớp vật lý trong mô hình hệ mở.

AS-i

- Sử dụng PP mã hoá bit mới để tương thích với truyền trên hai dây, đồng tải nguồn.
- ĐK truy nhập mạng và bảo toàn dữ liệu cũng thực hiện ở lớp vật lý.
- Master/Slave.
- Chẵn lẻ + mã hoá bit

AS-i Cấu trúc mạng

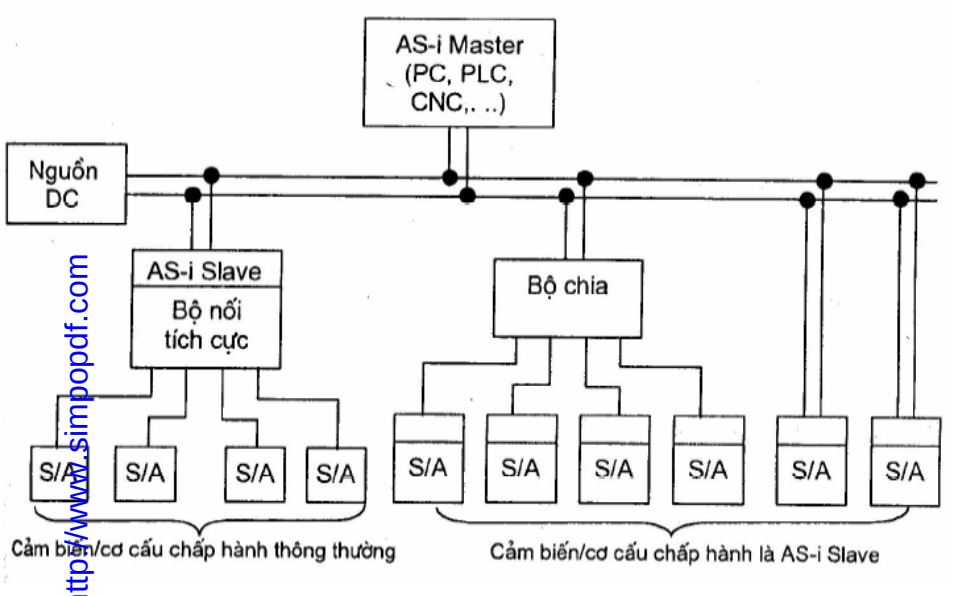
- Tuý chọn theo yêu cầu, vị trí, phạm vi đi dây.
- Daisy-chain, Trunk-line/drop-line.
- Cấu trúc hình cây.
- Có thể phân bố đều hay tham gia theo nhóm.
- Không yêu cầu điện trở đầu cuối (terminator).

AS-i

- Trạm chủ duy nhất có thể là: PLC, PC, hay bộ ĐK CNC, bộ nối field bus
- Trạm tớ là bộ nối tích cực ghép với 4 AS thông thường hay các AS có AS-i.
- Nối trực tiếp hay qua các bộ chia.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

AS-i

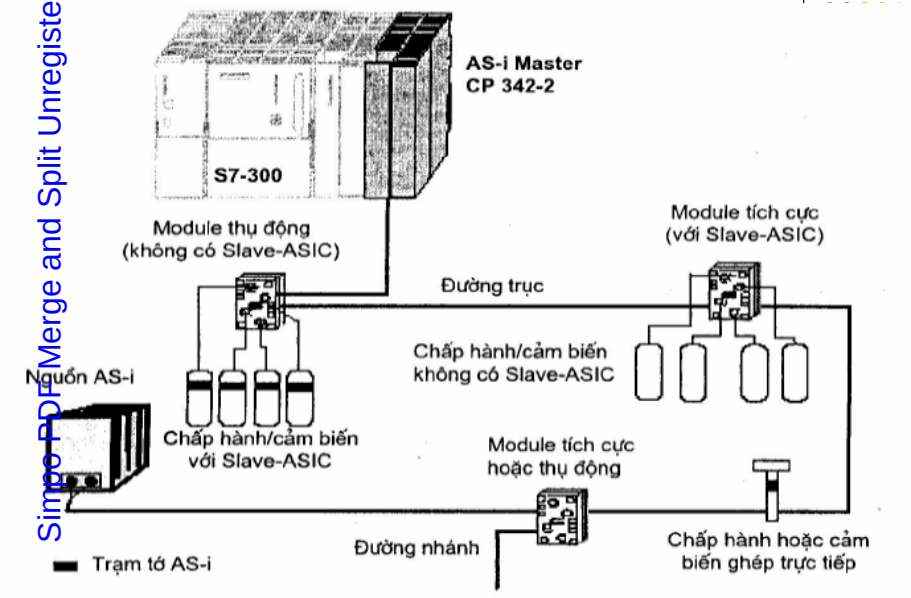


Simple PDF Merge and Split Unregistered Version - http://www.simplopdf.com

AS-i

- Chiều dài tổng là 100m. Mở rộng bằng các bộ repeater.
- Trạm tớ tối đa 31 trạm ->124 thiết bị.
- Thực hiện truyền hai chiều cho phép trạm chủ quản lý tối đa 124 kênh vào số và 124 kênh ra số.
- Tốc độ truyền qui định 167kbps (6us/1 bit).

AS-i



AS-i

- Cáp truyền qui định là cáp thông thường (cáp tròn), cáp đặc biệt (cáp dẹt).
- Lõi 1.5mm đáp ứng dòng tối thiểu 2A (24V)

AS-i Cơ chế giao tiếp

- Theo cơ chế Master/Slave.
- Trong một chu kỳ bus Master trao đổi với slave một lần theo phương pháp hỏi tuần tự (polling).
- Trạm chủ gửi bức điện có chiều dài 14 bit, 5bit địa chỉ và 5 bit dữ liệu.
- Nhận tín hiệu trả lờ trong t/g định trước.
- Bản tin trả lời có chiều dài 7 bit, 4 bit dữ liệu.

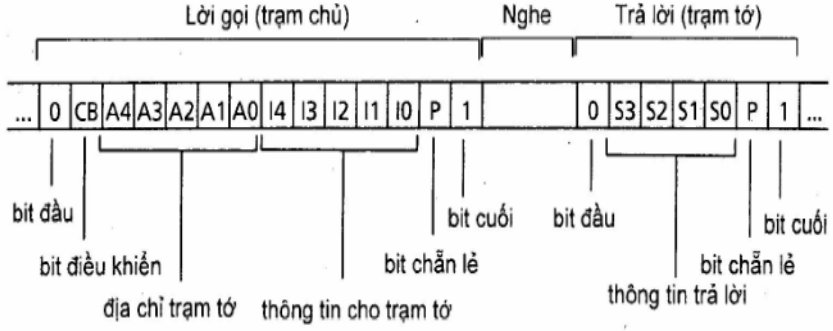
AS-i

- Chu kỳ bus phụ thuộc vào số trạm.
- Chu kỳ bus tối đa là 5ms với 31 trạm.
- M/S cho mạch ghép nối trạm tớ đơn giản, giá thành thực hiện thấp.
- Hệ thống linh hoạt.
- Có thể gửi riêng tín hiệu không trả lời không cần lặp lại cả chu kỳ.

AS-i

- Có thể gửi các thông báo khác mà không ảnh hưởng đến chu kỳ bus.
- 9 loại thông báo: hai phục vụ truyền số liệu và tham số, hai dùng để đặt địa chỉ, năm sử dụng để nhận dạng và xác định trạng thái trạm Slave.

AS-i Cấu trúc bức điện



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

AS-i

Trao đổi dữ liệu	0	0	A4	A3	A2	A1	A0	0	D3	D2	D1	D0	P	1
Đặt tham số	0	0	A4	A3	A2	A1	A0	0	P3	P2	P1	P0	P	1
Đặt địa chỉ	0	0	0	0	0	0	0	A4	A3	A2	A1	A0	P	1
Reset trạm tớ	0	1	A4	A3	A2	A1	A0	1	1	1	0	0	P	1
Xóa địa chỉ mặc định	0	1	A4	A3	A2	A1	A0	0	0	0	0	0	P	1
Đọc cấu hình vào/ra	0	1	A4	A3	A2	A1	A0	1	0	0	0	0	P	1
Đọc mã căn cước	0	1	A4	A3	A2	A1	A0	1	0	0	0	1	P	1
Đọc trạng thái	0	1	A4	A3	A2	A1	A0	1	1	1	1	0	P	1
Đọc và xóa trạng thái	0	1	A4	A3	A2	A1	A0	1	1	1	1	1	P	1

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

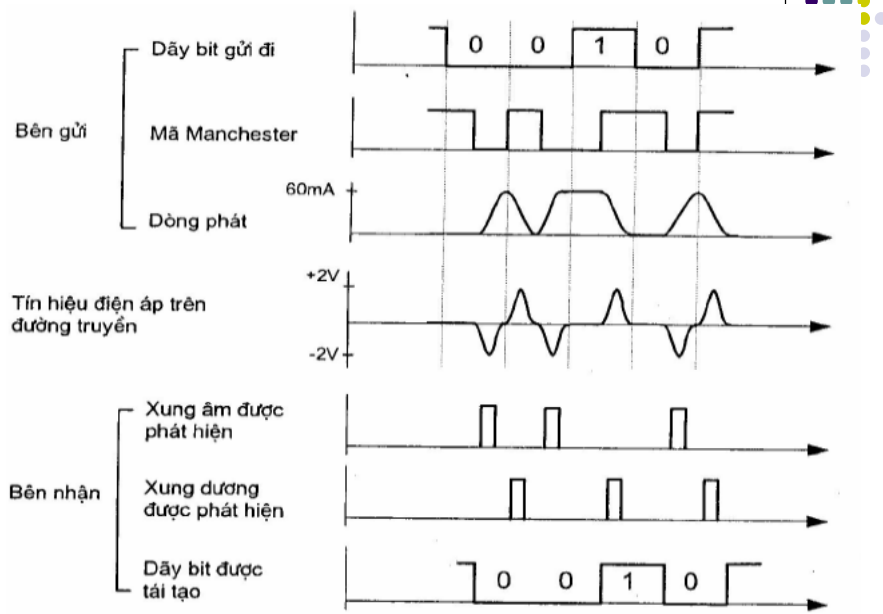
AS-i

- Giữa yêu cầu và trả lời cần thời gian nghỉ từ 3-8 bit.

AS-i Mã hoá bit

- Cáp hai dây thường suy hao mạnh khi tần số tăng, nhiễu CN nhiều -> hạn chế dải tần của tín hiệu.
- Đơn giản, hiệu suất cao, khả năng đồng bộ nhịp, phối hợp phát hiện lỗi.
- APM (Alternate Pulse Modulation)

AS-i



AS-i

- Xung áp gần hình sin -> dải tần hẹp tần số tín hiệu tương đương với tần số xung nhịp.
- Thay đổi tuần tự xung dương, âm triệt tiêu dòng một chiều.

AS-i

- Chẵn lẻ có HD=2, nhưng tỉ lệ lỗi bit còn lại thấp.
- Khi p=0,0012 (200lần lỗi/s) thì Ttmbf=10 năm

AS-i Bảo toàn dữ liệu

- Bản tin ngắn, hiệu suất cáo.
- Kiểm tra chẵn lẻ và mã hoá bit hợp lý.
- Trong một chu kỳ bit (6uS) bên thu lấy mẫu tín hiệu 16 lần, mỗi chu kỳ phải có một hay hai xung, các xung kế tiếp phải đảo chiều -> chỉ có tín hiệu dạng này mới được nhận.
- Bức điện có chiều dài cố định, ngăn bởi các t/g nghỉ-> sai lệch cũng được phát hiện.
- Kiểm tra chẵn lẻ.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Mạng máy tính
&
Hệ thống thông tin công nghiệ

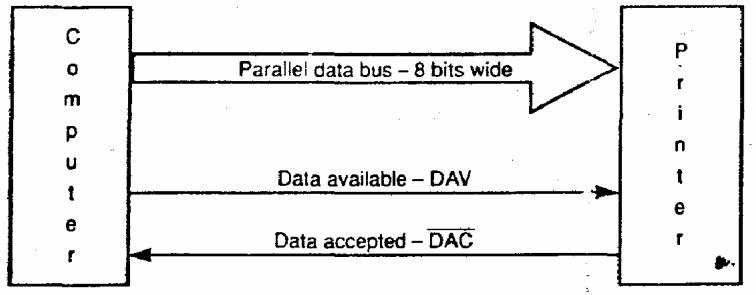
Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Truyền song song.

- Máy tính lưu và xử lý số liệu theo từng từ (có độ dài -8,-16,-32,-64 bit).
- Dữ liệu sẽ được cấp theo dạng song song mỗi lần một từ, mỗi một bit có một đường dẫn riêng.
- Ta có 8 (16,32,64) dây dẫn song song nối giữa 2 điểm truyền đồng thời 8 (16,32,64) mức điện áp (0/1).
- Như vậy truyền song song là truyền từng byte (từ có độ dài 8,16,32,64 bit).
- Phương pháp truyền song song có tốc độ truyền cao, nó thường được sử dụng khi truyền bên trong các thiết bị hay giữa các linh kiện trên cùng một tấm mạch in,
- Tuy nhiên khi truyền ở khảng cách xa phương pháp này có nhược điểm là tốn dây dẫn và có sự sai khác về mặt thời gian của các tín hiệu

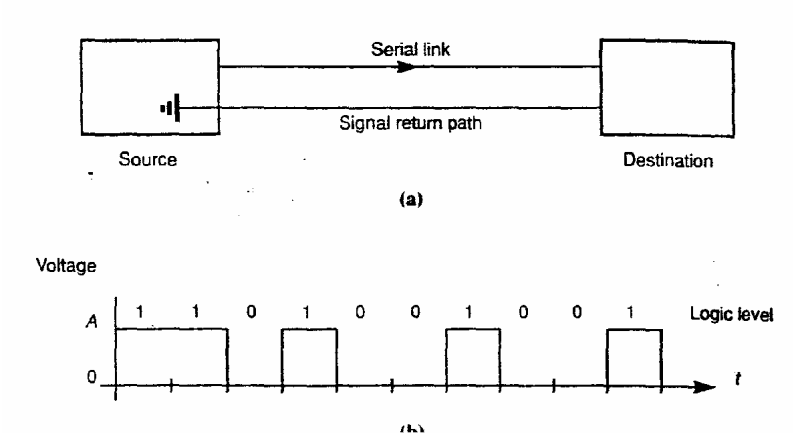
Truyền song song.



Truyền nối tiếp.

- Sử dụng hai dây dẫn nối giữa hai điểm. Các mức điện áp ON/OFF sẽ được truyền tuần tự với một chuẩn thời gian theo hai dây dẫn.
- Như vậy truyền nối tiếp là truyền từng bit
- Phương pháp này tuy có tốc độ thấp hơn phương pháp song song nhưng nó khắc phục được các hạn chế của phương pháp song song khi truyền ở khoảng cách xa.

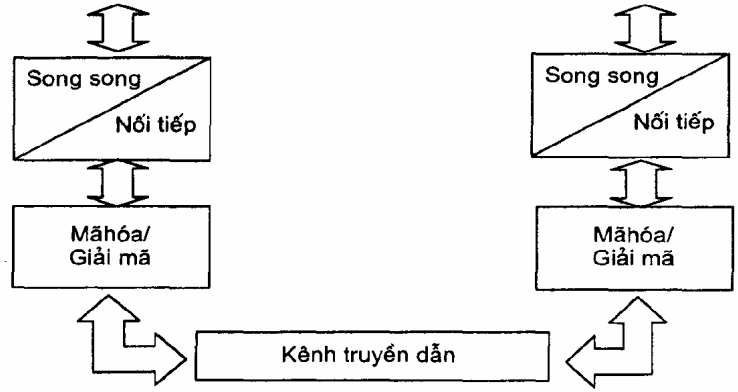
Truyền nối tiếp.



Truyền nối tiếp.

- Máy tính dù tồn tại ở dạng nào đều có các bộ VXL và bus song song và xử lý số liệu song song. Vì vậy, để có thể dùng phương pháp truyền nối tiếp, ta cần có các bộ chuyển đổi song song và nối tiếp

Truyền nối tiếp.

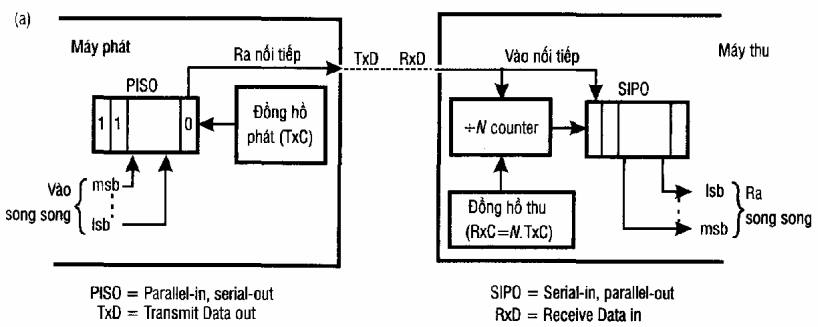


Truyền không đồng bộ

- Có thời gian nghỉ giữa các khung bản không cố định.
- Việc truyền được bắt đầu bởi 1 Start bit, các bit được truyền với một thông số định trước.
- Một đặc điểm quan trọng là thông số của cổng truyền phải giống nhau ở bên phát và bên thu để đảm bảo độ dài của chuỗi bit dữ liệu là như nhau.
- Trong truyền bất đồng bộ, đồng hồ thu chạy một cách bất đồng bộ với tín hiệu thu. Đê xử lý thu hiệu quả, cần phải có kế hoạch dùng đồng hồ thu để lấy mẫu tín hiệu đến, ngay điểm giữa thời bit của dữ liệu
- Đề đạt được điều này, tín hiệu đồng hồ thu nhanh gấp N lần đồng hồ phát và mỗi bit được dịch vào SIPO sau N chu kỳ xung đồng hồ

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Truyền không đồng bộ



Truyền không đồng bộ

Nguyên tắc đồng bộ ký tự

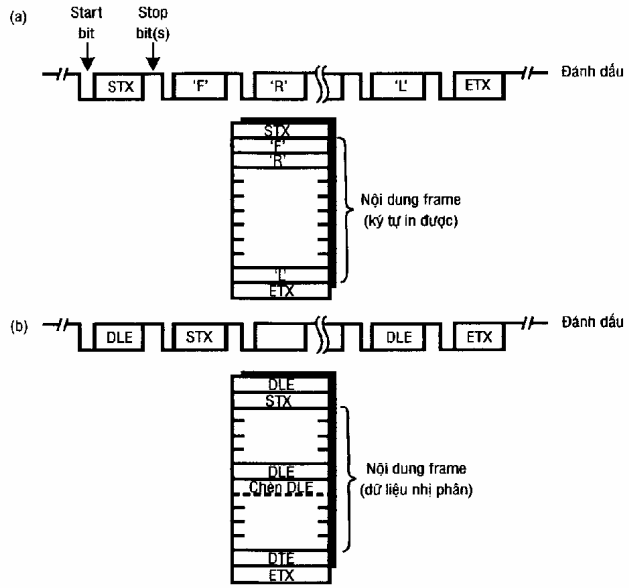
Mạch điều khiển truyền nhận được lập trình để hoạt động với số bit bằng nhau trong một ký tự kể cả số stop bit, start bit và bit kiểm tra giữa thu và phát. Sau khi phát hiện và nhận start bit, việc đồng bộ ký tự đạt được tại đầu thu rất đơn giản, chỉ việc đếm đúng số bit đã được lập trình. Sau đó sẽ chuyển ký tự nhận được vào thanh ghi đệm thu nội bộ và phát tín hiệu thông báo với thiết bị điều khiển (CPU) rằng đã nhận được một ký tự mới. Và sẽ đợi cho đến khi phát hiện một start bit kế tiếp.

Truyền không đồng bộ

Nguyên tắc đồng bộ frame

Khi thông điệp gồm một khối các ký tự thường xem như một frame thông tin (information frame) được truyền, bên cạnh việc đồng bộ bit và đồng bộ ký tự, máy thu còn phải xác định được điểm bắt đầu và điểm kết thúc một frame. Điều này được gọi là sự đồng bộ frame

Truyền không đồng bộ



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

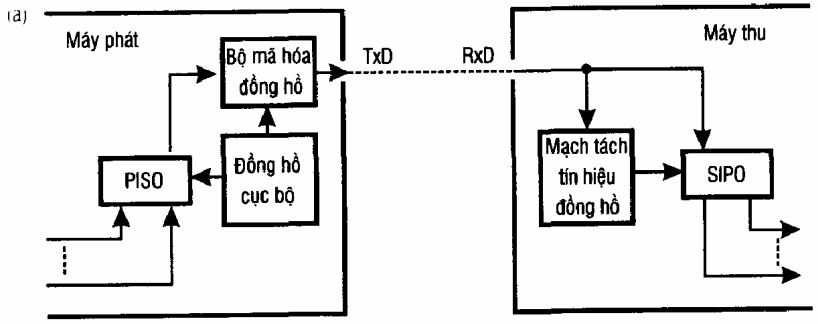
Truyền đồng bộ

So với truyền không đồng bộ, truyền đồng bộ có một số khác biệt như là:

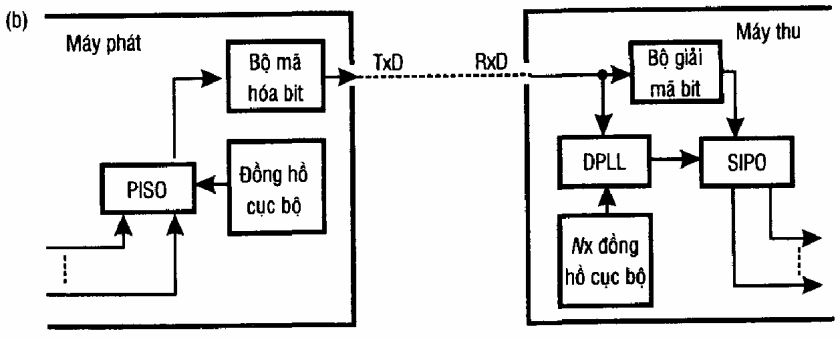
- Số liệu được truyền liên tục, không có các bit start, stop
- Có khung bản tin lớn hơn
- Cần có giao thức để điều khiển và ổn định luồng dữ liệu.
- Tuy nhiên, cũng giống như truyền bất đồng bộ chúng ta chí chấp nhận phương pháp nào cho phép máy thu đạt được sự đồng bộ bit, đồng bộ ký tự và đồng bộ frame.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

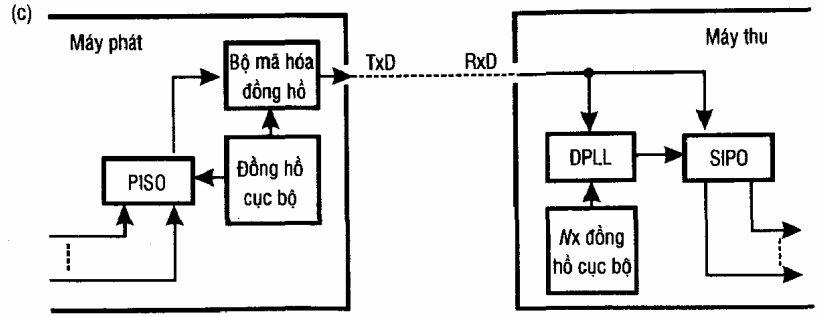
Truyền đồng bộ



Truyền đồng bộ



Truyền đồng bộ



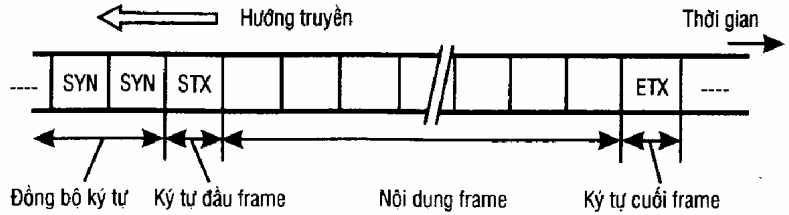
Truyền đồng bộ

Truyền đồng bộ hướng ký tự:

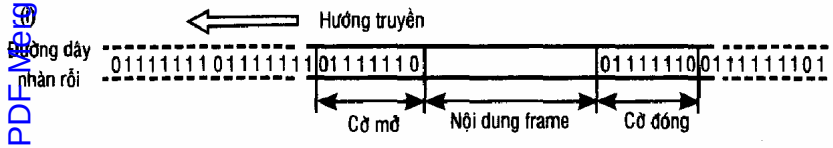
Đê thực hiện việc đồng bộ này, máy phát thêm vào các ký tự điều khiển truyền, gọi là các ký tự đồng bộ SYN, ngay trước các khối ký tự truyền. Các ký tự điều khiển này phải có hai chức năng: trước hết, chúng cho phép máy thu duy trì đồng bộ bit. Thứ hai, khi điều này đã được thực hiện, chúng cho phép máy thu bắt đầu biên dịch luồng bit thu theo các ranh giới ký tự chính xác *sự đồng bộ ký tự.*

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

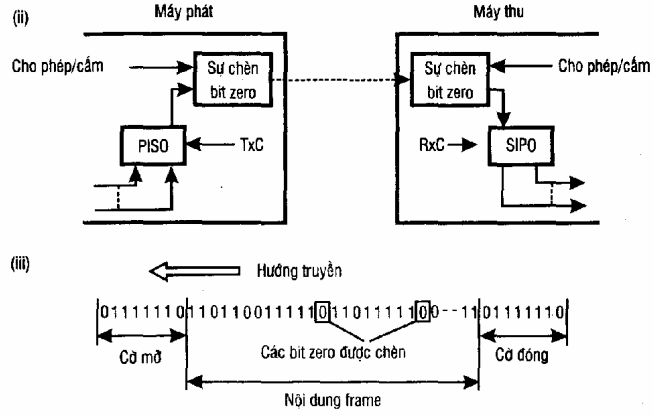
Truyền đồng bộ-Hướng bit



Truyền đồng bộ

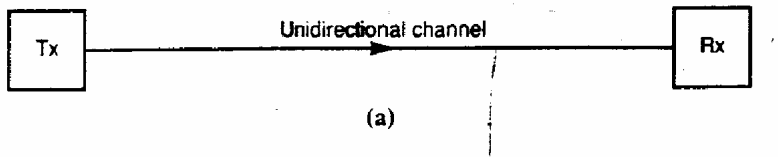


Truyền đồng bộ



Truyền 1 chiều (simplex)

Truyền đơn công (Simplex): Là hệ được thiết kế để truyền số liệu theo một chiều không cung cấp chiều ngược lại.

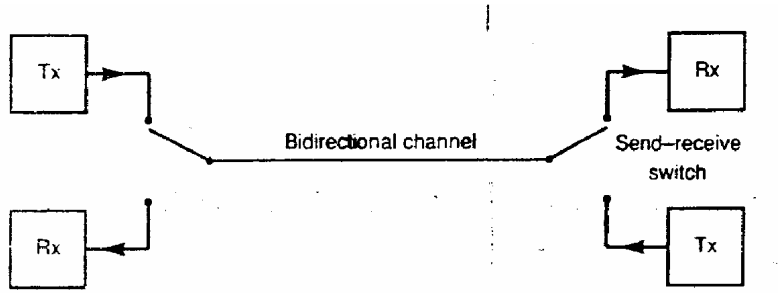


Truyền 2 chiều (duplex)

Hệ song công (Duplex): Là hệ được thiết kế để truyền số liệu theo cả hai chiều

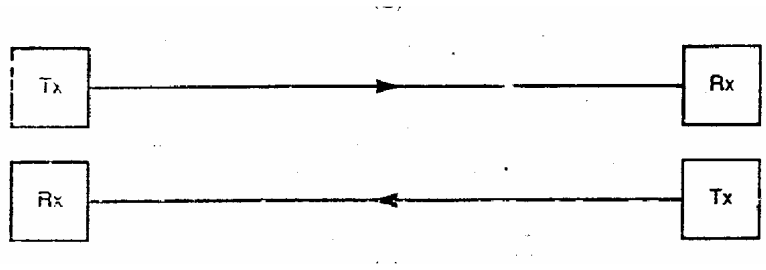
Truyền Half-duplex)

Bán song công (Half Duplex): Là hệ có thể truyền số liệu theo cả hai chiều nhưng tại mỗi thời điểm chỉ thực hiện một chiều.



Truyền Full-duplex)

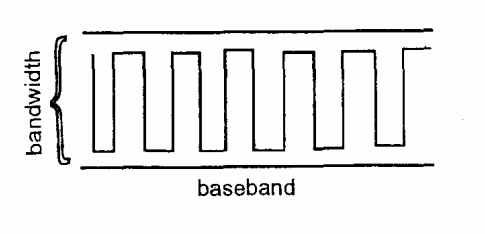
Hệ song công (Full Duplex): Là hệ có thể truyền số liệu hai chiều một cách đồng thời



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Truyền tải cơ sở

Một tín hiệu mang một nguồn thông tin có thể biểu diễn bằng tổng của nhiều dao động có tần sỗ khác nhau nằm trong một phạm vi hẹp, được gọi là dải tần cơ sở (Base Band) hay dải hẹp. Tín hiệu được truyền đi cũng chính là tín hiệu được tạo ra sau khi mã hóa bit, nên có tần số cố định hoặc nằm trong một khoảng hẹp nào đó, tùy thuộc vào phương pháp mã hóa bit



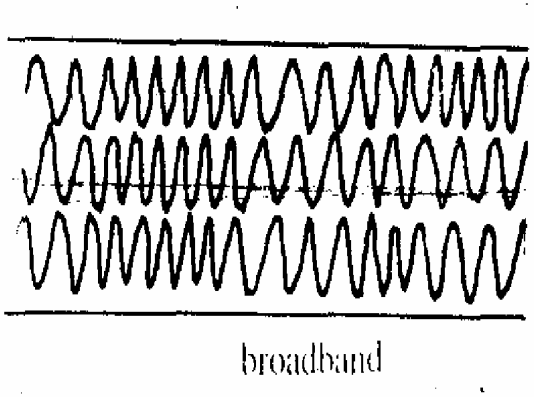
Truyền tải dải mang

Trong một sồ trường hợp, dải tần cơ sở không tương thích trong môi trường làm việc. Để khắc phục tình trạng này, người ta sử một tín hiệu khác - gọi là tín hiệu mang, có tần số nằm trong một dải tần thích hợp - gọi là dải mang. Dải tần này thường lớn hơn nhiều so với tần số nhịp. Dữ liệu cần truyền tải sẽ dùng để điều chế tần số, biên độ hoặc pha của tín hiệu mang

Truyền tải dải rộng

Một tín hiệu có thể chứa đựng nhiều nguồn thông tin khác nhau bằng cách sử dụng kết hợp một cách thông minh nhiều thông số thông tin. Sau khi nhiều nguồn thông tin khác nhau đã được mã hoá bịt, mỗi tín hiệu được tạo ra sẽ dùng để điều biến một tín hiệu khác, thường có tần số lớn hơn nhiều, gọi là tín hiệu *mang.* Các tín hiệu mang đã được điều biến có tần sồ khác nhau, nên có thể pha trộn, xếp chồng thành một tín hiệu duy nhất có phổ tần trải rộng. Đây chính là kỹ thuật dồn kênh phân tần trong truyền tải thông tin, nhằm mục đích sử dụng hiệu quả hơn đường truyền. Phía bên nhận sẽ thực hiện việc giải điều biến và phân kênh, hồi phục các tín hiệu mang các nguồn thông tin khác nhau.

Truyền tải dải rộng



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Liên kết

** Liên kết: Liên kết (link) là mối quan hệ vật lý hoặc logic giữa hai hoặc nhiều đối tác truyền thông. Đối với liên kết vật lý, các đối tác chính là các trạm truyền thông được liên kết với nhau qua một môi trường vật lý.*

Có thể phân biệt các kiểu liên kết sau đây:

-Liên kết điểm-điểm (potnt-to-point): Một mối nên kết chỉ có hai đối tác tham gia.

- *Liên kết điểm-nhiều điểm (mu/ti-drop):* Trong một mối liên kết có nhiều đối tác tham gia, tuy nhiên chỉ một đối tác cố định duy nhất (trạm chủ) có khả năng phát trong khi nhiều đối các còn lại (các trạm tớ) thu nhận thông tin cùng một lúc.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Liên kết

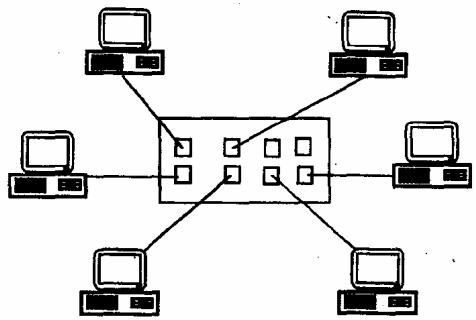
- Liên kết nhiều điểm (Multipoint): Trong một mối liên kết có nhiều đối tác tham gia và có thể trao đổi thông tin qua lại tự do theo bất kỳ hướng nào. Bất cứ một đối tác nào cũng có quyền phát và bất cứ trạm nào cũng nghe được. Cũng như kiểu liên kết điểm-nhiều điểm có thể sử dụng một cáp dẫn duy nhất để nối mạng giữa các đối tác.

Topology

** Topology:*

Topology là cấu trúc liên kết của một mạng, hay nói cách khác chính là tổng hợp của các liên kết. Topology có thể hiểu là cách sắp xếp, tổ chức về mặt vật lý của mạng, nhưng cũng có thể là cách sắp xếp logic của các nút mạng, cách định nghĩa về tổ chức logic các mối liên kết giữa các nút mạng.

Cấu trúc mạng dạng sao



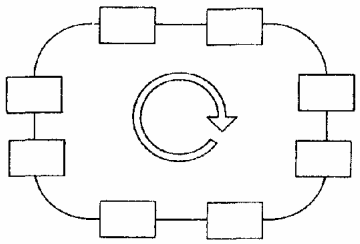
Cấu trúc mạng dạng sao

- Cấu trúc hình sao là một cấu trúc mạng tất cả các trạm được nối vào một trạm trung tâm
- Trạm trung tâm này sẽ điều khiển hoạt động truyền thông của toàn mạng.
- Tuỳ theo yêu cầu truyền thông trạm trung tâm có thể là một bộ chuyển mạch (switch), một bộ chọn đường (router) hay đơn giản là một bộ phân kênh.
-Ta có thể nhận thấy ở đây kiểu liên kết về mặt vật lý là điểm-điểm. Tuy nhiên, liên kết về mặt logic vẫn có thể là nhiều điểm. Nếu trạm trung tâm đóng vai trò tích cực, nó có thể đảm đương nhiệm vụ kiểm soát toàn bộ việc truyền thông của mạng, còn nếu không sẽ chỉ như một bộ chuyển mạch.

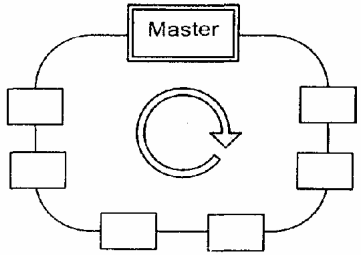
Cấu trúc mạng dạng sao

- Nhược điểm của cấu trúc hình sao là sự cố ở trạm trung tâm sẽ làm tê liệt toàn bộ các hoạt động truyền thông trong mạng, tốn dây dẫn và độ dài của dây nối với trạm trung tâm hạn chế.
- Ưu điểm của cấu trúc hìng sao là lắp đặt đơn giản, dễ kiểm soát và khắc phục sự cố. Do sử dụng liên kết điểm-điểm do vây có thể tận dụng tối đa tốc độ truyền của đường truyền vậy lý.

Cấu trúc mạng dạng vòng



a) Không có điều khiển trung tâm



b) Có điều khiển trung tâm

Cấu trúc mạng dạng vòng

- Cấu trúc mạch vòng được thiết kế sao cho các thành viên trong mạng được nối từ điểm này đến điểm kia một cách tuần tự theo một mạch vòng khép kín.
- Trong vòng, tín hiệu được truyền đi theo một chiều qui định. Mỗi trạm nhận được dữ liệu từ trạm đứng trước và chuyển tiếp sang trạm lân cận đứng sau. Quá trình này được lặp lại tới khi dữ liệu quay trở về trạm đã gửi.
- Ưu điểm cơ bản của mạng cấu trúc theo kiểu này là mỗi một nút đồng thời có thể là một bộ khuếch đại, do vậy khi thiết kế mạng theo kiểu cấu trúc vòng có thể thực hiện với khoảng cách và số trạm rất lớn. Mỗi trạm có khả năng vừa nhận vừa phát tín hiệu cùng một lúc. Bởi mỗi thành viên ngăn cách vòng ra làm hai phần.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cấu trúc mạng dạng vòng

- Với kiểu mạch vòng không có điều khiển trung tâm, các trạm đều bình đẳng như nhau trong quyền nhận và phát tín hiệu. Như vậy việc kiểm soát đường dẫn sẽ do các trạm tự phân chia.

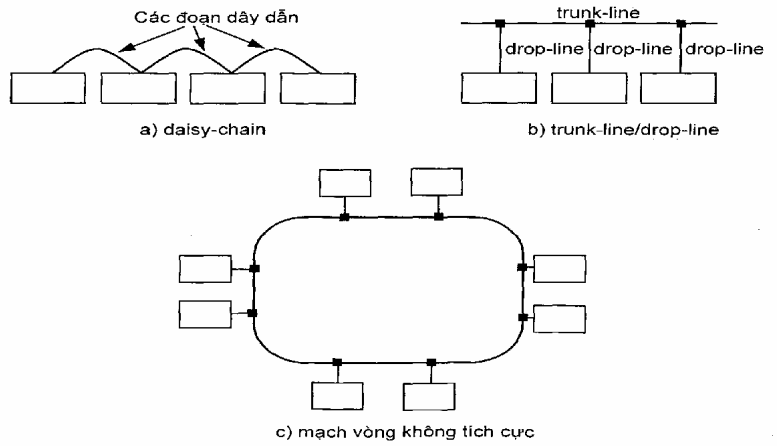
- Với kiểu có điều khiển trung tâm, một trạm chủ sẽ đảm nhiệm vai trò kiểm soát việc truy nhập đường dẫn.

- Cấu trúc mạch vòng thực chất dựa trên cơ sở liên kết điểm-điểm, vì vậy thích hợp cho việc sử dụng các phương tiện truyền tín hiệu hiện đại như cáp quang, tia hồng ngoại, v.v.

- Một ưu điểm tiếp theo của cấu trúc mạch vòng là khả năng xác định vị trí xảy ra sự cố, ví dụ đứt dây hay một trạm ngừng làm việc. Tuy nhiên, sự hoạt động bình thường của mạng còn trong trường hợp này chỉ có thể tiếp tục với một đường dây dự phòng. Mạng dạng vòng đòi hỏi phải có một giao thức điều khiển truy nhập đường truyền khá phức tạp.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cấu trúc mạng dạng Bus



Cấu trúc mạng dạng Bus

- Trong cấu trúc đơn giản này, tất cả các thành viên của mạng đều được nối trực tiếp với một đường dẫn chung.

- Khi một trạm gửi tín hiệu ra Bus thì nó sẽ quảng bá tới tất cả các trạm còn lại.

- Đặc điểm cơ bản của cấu trúc bus là việc sử dụng chung một đường dẫn duy nhất cho tất cả các trạm, vì thế tiết kiệm được cáp dẫn và công lắp đặt

Cấu trúc mạng dạng Bus

-Bên cạnh việc tiết kiệm dây dẫn thì tính đơn giản, dễ thực hiện là những ưu điểm chính của cấu trúc bus,

- Mạng cần phải có một giao thức để điều khiển việc truy nhập đường truyền.

Ngoài việc cần phải kiểm soát truy nhập đường truyền, cấu trúc bus có những nhược điểm sau:

- Một tín hiệu gửi đi có thể tới tất cả các trạm và theo một trình tự không kiểm soát được, vì vậy phải thực hiện phương pháp gán địa chỉ (logic) theo kiểu thủ công cho từng trạm. Trong thực tế, công việc gán địa chỉ này gây ra không ít khó khăn.

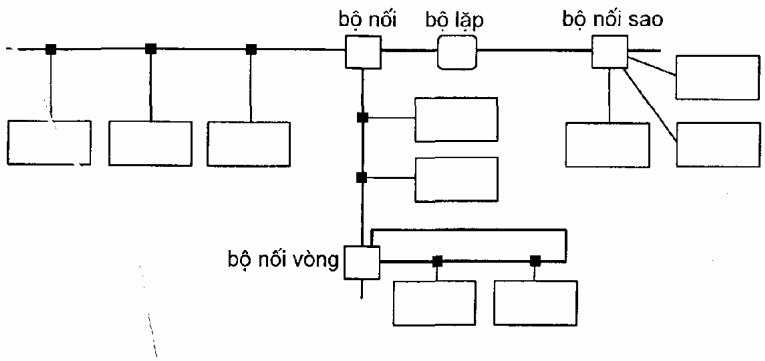
Cấu trúc mạng dạng Bus

- Tất cả các trạm đều có khả năng phát và phải luôn luôn nghe đường dẫn để phát hiện ra một thông tin có phải gửi cho mình hay không, nên phải được thiết kế sao cho đủ tải với số trạm tối đa. Đây chính là lý do phải hạn chế số trạm trong một đoạn mạng. Khi cần mở rộng mạng, phải dùng thêm các bộ lặp.
- Chiều dài dây dẫn thường tương đối dài, vì vậy đối với cấu trúc đường thẳng xảy ra hiện tượng phản xạ tại mỗi đầu dây làm giảm chất lượng của tín hiệu. Để khắc phục vấn đề này người ta chặn hai đầu dây bằng hai trở đầu cuối (Terminator). Việc sử dụng các trở đầu cuối cũng làm tăng tải của hệ thống.

Cấu trúc mạng dạng Bus

- Trường hợp đường dẫn bị đứt, hoặc do ngắn mạch trong phần kết nối bus của một trạm bị hỏng đều dẫn đến ngừng hoạt động của cả hệ thống. Việc định vị lỗi ở đây cũng gặp rất nhiều khó khăn.
- Cấu trúc đường thẳng, liên kết đa điểm gây khó khăn trong việc áp dụng các công nghệ truyền tín hiệu mới như sử dụng cáp quang.

Cấu trúc mạng dạng cây



Điều khiển truy nhập đường truyền

- Trong một mạng có cấu trúc bus,hay dạng vòng, các thành viên phải chia nhau một đường dẫn chung.
- Để tránh sự xung đột về tín hiệu gây ra sai lệnh về thông tin, ở mỗi thời điểm trên một đường dẫn chỉ duy nhất một điện tín được phép truyền đi. Chính vì vậy mạng phải được điều khiển sao cho tại một thời điểm nhất định chỉ một thành viên trong mạng được gửi thông tin đi. Còn số lượng thành viên trong mạng muốn nhận thông tin thì không hạn chế.
- Một trong những vấn đề quan trọng hàng đầu ảnh hưởng tới chất lượng của mỗi hệ thống là phương pháp phân chia thời gian gửi thông tin trên đường dẫn hay phương pháp *truy nhập đường truyền*.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Điều khiển truy nhập đường truyền

- Phương pháp truy nhập đường truyền là một trong những vấn đề cơ bản đối với các hệ thống , bởi mỗi phương pháp có những ảnh hưởng khác nhau tới các tính năng kỹ thuật của hệ thống. Cụ thể, ta phải quan tâm tới ít nhất 3 khía cạnh: độ tin cậy, tính năng thời gian, hiệu xuất sử dụng đường truyền.

Các phương pháp điều khiển truy nhập đường truyền có thể chia thành hai nhóm chính:

- Điều khiển truy nhập ngẫu nhiên: việc truy nhập không được qui định chặt chẽ trước mà xảy ra hoàn toàn ngẫu nhiên theo nhu cầu của các trạm.

- Điều khiển truy nhập có điều khiển: Trình tự truy nhập được xác định rõ ràng từ trước. Việc truy nhậy được kiểm soát chặt chẽ theo cách tập trung hay phân tán bởi các thành viên.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

CSMA/CD

Carrier Sense Multiple Access with Collision Detection-*Phương pháp đa truy nhập sử dụng sóng mang có pháp hiện xung đột*

- Phương pháp truy nhập ngẫu nhiên này được sử dụng cho topo dạng bus, trong đó tất cả các trạm của mạng được nối trực tiếp vào bus.

- Mọi trạm đều có thể truy nhập vào bus chung (đa truy nhập) một cách ngẫu nhiên và do vậy rất có thể dẫn đến xung đột (hai hoặc nhiều trạm đồng thời truyền dữ liệu).

-Dữ liệu được truyền đi theo khuôn dạng chuẩn trong đó có vùng thông tin điều khiển chứa địa chỉ của dữ liệu.

CSMA/CD

- CSMA/CD là phương pháp cải tiến từ phương pháp CSMA, hay còn gọi là LBT (Listen Before Talk - *Nghe trước khi nói)*. Tư tưởng của nó là : một trạm cần truyền dữ liệu trước hết phải "nghe" xem đường truyền xem đang rỗi hay bận. Nếu rỗi thì truyền dữ liệu đi (theo khuôn dạng chuẩn). Ngược lại, nếu đường truyền đang bận (đã có dữ liệu khác) thì trạm phải thực hiện theo một trong 3 giải thuật sau (thường gọi là các giải thuật "kiên nhẫn"-persistent algorithms) :

(1) Trạm tạm "rút lui" chờ đợi trong một thời đoạn ngẫu nhiên nào đó rồi lại bắt đầu "nghe" đường truyền (Non persistent) .

(2) Trạm tiếp tục "nghe" đến khi đường truyền rỗi thì truyền dữ liệu đi với xác suất bằng 1 (1-persistent)

(3) Trạm tiếp tục "nghe" đến khi đường truyền rỗi thì truyền đi với xác suất p xác định trước (0<p<1) (p-persistent).

CSMA/CD

- Việc xẩy ra xung đột thường là do độ trễ truyền dẫn : một trạm truyền dữ liệu (cùng sóng mang) đi rồi nhưng do độ trễ truyền dẫn nên một trạm khác lúc đó đang "nghe" đường truyền sẽ tưởng là rỗi và cứ thế truyền dữ liệu đi. Mấu chốt vấn đề là ở chỗ : vì các trạm chỉ "nghe *trước khi* nói" mà không "nghe *trong khi* nói" nên thực tế có xung đột nhưng các trạm vẫn không hay biết gì và vẫn cứ tiếp tục truyền dữ liệu đi, gây ra việc chiếm dụng đường truyền một cách vô ích.

CSMA/CD

Để có thể phát hiện xung đột, CSMA/CD-hay còn gọi là LWT(Listen While Talk-Nghe trong khi nói) đã bổ sung thêm qui tắc :

- Khi một trạm đang truyền nó vẫn tiếp tục "nghe" đường truyền.Nếu phát hiện thấy xung đột thì nó ngừng ngay việc truyền nhưng vẫn tiếp tục gửi tín hiệu sóng mang thêm một thời gian nữa để đảm bảo rằng tất cả các trạm trên mạng đều có thể "nghe" được sự kiện xung đột đó.
- Sau đó trạm chờ đợi trong một thời đoạn ngẫu nhiên nào đó rồi thử truyền lại theo các qui tắc của CSMA

CSMA/CD

- Rõ ràng với CSMA/CD, thời gian chiếm dụng vô ích đường truyền được giảm xuống bằng thời gian dùng để phát hiện một xung đột.
- Ưu điểm của CSMA/CD là tính chất đơn giản, linh hoạt, việc ghép thêm hay bỏ đi một trạm trong mạng không ảnh hưởng gì tới hoạt động của hệ thống.
- Nhược điểm của CSMA/CD là tính bất định của thời gian phản ứng, hiệu xuất sử dụng đường truyền vì thế cũng thấp.

CSMA/CA

(*Carrier Sense Multiple Access* with *Collislon Avoidance)*

- Sử dụng cho Topo mạng dạng Bus. Tương tự như CSMA/CD, mỗi trạm đều phải nghe đường dẫn trước khi gửi cũng như sau khi gửi thông tin.
- ở đây sử dụng một phương pháp mã hóa bit thích hợp để trong trường hợp xảy ra xung đột, một tín hiệu "trội" (dominant) sẽ lấn át tín hiệu kia "lặn" (recessive).
- Nếu một trạm gửi đi tín hiệu "lặn" mà giám sát về tín hiệu "trội" thì nó sẽ mất quyền ưu tiên và phải dừng truyền. Sau đó trạm sẽ chờ một thời gian ngẫu nhiên nào đó và thử nghe lại đường truyền.

CSMA/CA

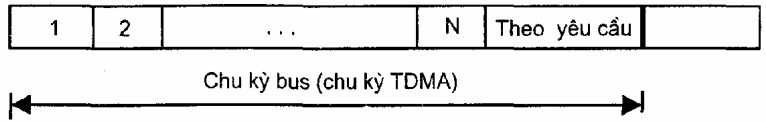
- Mỗi bức điện đều được bắt đầu bằng một dãy bit đặc biệt được gọi là cờ hiệu, sau đó là tới các phần khác như thông tin kiểm soát, địa chỉ,...
- Phương pháp CSMA/CA, có thể sử dụng mức ưu tiên cho mỗi trạm (hoặc theo loại thông tin) và gắn mã ưu tiên vào phần đằng sau cờ hiệu của mỗi bức điện.
- Nhờ có phương pháp sử dụng mức ưu tiên mà tính năng thời gian thực của hệ thống được cải thiện. Có thể thấy rõ, tuy bị hạn chế về tốc độ truyền và chiều dài đây dẫn, hiệu suất sử dụng đường truyền ở phương pháp này rất cao. Các trạm chỉ gửi thông tin đi khi có nhu cầu và nếu xảy ra xung đột thì một trong hai bức điện vẫn tiếp tục được gửi đi.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

TDMA (Time Divsion Multiple Access):

- Sử dụng cho Topo mạng dạng Bus.Trong phương pháp kiểm soát truy nhập phân chia thời gian TDMA, mỗi trạm được phân một thời gian truy nhập bus nhất định. Các trạm có thể lần lượt thay nhau gửi thông tin trong khoảng thời gian cho phép gọi là khe *thời gian* hay *lát* thời gian (time *slot, tim slice*) theo một tuần tự qui định sẵn. Việc phân chia nàyđược thực hiện trước khi hệ thống đi vào hoạt động (tiền định).
- Hệ thống có thể hoạt động không có trạm chủ. Trong trường hợp có một trạm chủ thì vai trò của nó chỉ hạn chế ở mức độ kiểm soát việc tuân thủ đảm bảo giữ đúng lát thời gian của các trạm khác. Mỗi trạm đều có khả năng đảm nhiệm vai trò chủ động trong giao tiếp trực tiếp với các trạm khác.

TDMA (Time Divsion Multiple Access):



TDMA (Time Divsion Multiple Access):

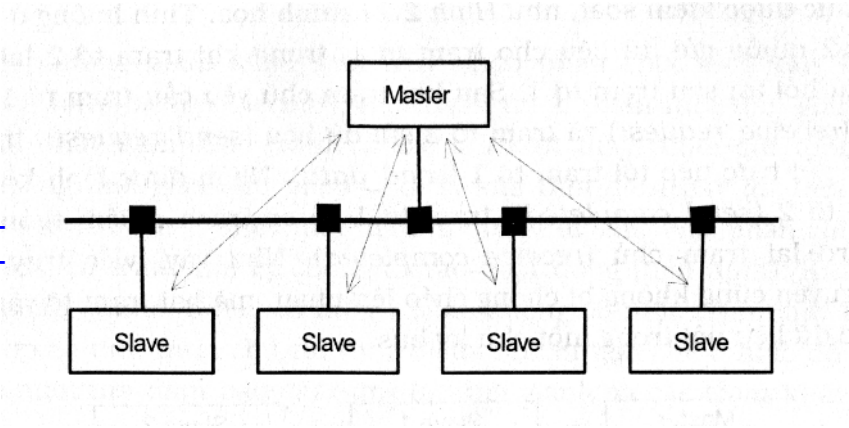
- Ngoài các lát thời gian phân chia cố định cho các trạm dùng để trao đổi dữ liệu định kỳ (đánh số từ 1 tới N), thường còn có một khoảng dự trữ dành cho việc trao đổi dữ liệu bất thường theo yêu cầu, ví dụ gửi thông tin cảnh báo, mệnh lệnh đặt cấu hình, dữ liệu tham số, setpoint..

Master/Slave

- Sử dụng cho cấu trúc mạng dạng Bus. Trong phương pháp chủ/tớ, một trạm chủ (master) có trách nhiệm chủ động phân chia quyền truy nhập bus cho các trạm tớ (slave*).*
- Các trạm tớ đóng vai trò bị động, chỉ có quyền truy nhập bus và gửi tín hiệu đi khi có yêu cầu. Trạm chủ có thể dùng phương pháp hỏi tuần tự *(polling)* theo chu kỳ để kiểm soát toàn bộ hoạt động giao tiếp của cả hệ thống.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Master/Slave



Master/Slave

- Trong một số hệ thống, thậm chí các trạm tớ không có quyền giao tiếp trực tiếp với nhau, mà bất cứ dữ liệu cần trao đổi nào cũng phải qua trạm chủ. Nếu hoạt động giao tiếp diễn ra theo chu kỳ, trạm chủ sẽ có trách nhiệm chủ động yêu cầu dữ liệu từ trạm tớ cần gửi và sau đó sẽ chuyển tới trạm tớ cần nhận. Trong trường hợp một trạm tớ cần trao đổi dữ liệu bất thường với một trạm khác phải thông báo yêu cầu của mình khi được trạm chủ hỏi đến và sau đó chờ được phục vụ. Trình tự tham gia giao tiếp, hay trình tự hỏi/đáp của các trạm tớ có thể do người dùng qui định trước (tiền định) bằng các công cụ đặt cấu hình.

Master/Slave

- Phương pháp chủ/tớ có một ưu điểm là việc kết nối mạng các trạm tớ đơn giản, đỡ tốn kém bởi gần như toàn bộ "trí tuệ" tập trung tại trạm chủ. Một trạm chủ thường là một thiết bị điều khiển, vì vậy việc tích hợp thêm chức năng xử lý truyền thông là điều không khó khăn.

- Một nhược điểm của phương pháp kiểm soát tập trung chủ/tớ là hiệu suất trao đổi thông tin giữa các trạm tớ bị giảm do phải dữ liệu phải đi qua khâu trung gian là trạm chủ, dẫn đến giảm hiệu suất sử dụng đường truyền.

Master/Slave

- Một hạn chế nữa của phương pháp này là độ tin cậy của hệ thống truyền thông phụ thuộc hoàn toàn vào một trạm chủ duy nhất. Trong trường hợp có xảy ra sự cố trên trạm chủ thì toàn bộ hệ thống truyền thông ngừng làm việc. Một cách khắc phục là sử dụng một trạm tớ đóng vai trò giám sát trạm chủ và có khả năng thay thế trạm chủ khi cần thiết.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

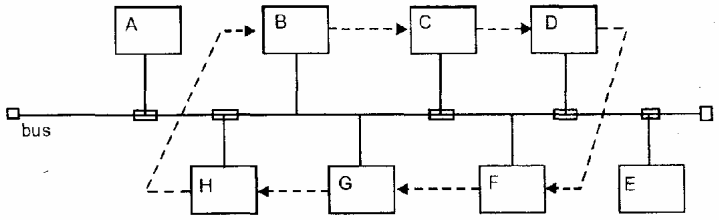
Token Bus

- Phương pháp này sử dụng cho topo mạng dạng Bus.
- Nguyên lý của phương pháp này là : để cấp phát quyền truy nhập đường truyền cho các trạm đang có nhu cầu truyền dữ liệu, một thẻ bài được lưu chuyến trên một vòng lôgic thiết lập bởi các trạm đó.
- Khi một trạm nhận được thẻ bài thì nó có quyền sử dụng đường truyền trong một thời gian xác định trước. Trong thời gian đó nó có thể truyền một hay nhiều đơn vị dữ liệu. Khi đã hết dữ liệu hoặc hết thời gian cho phép, trạm phải chuyển thẻ bài đến trạm tiếp theo trong vòng lôgic.

Token Bus

- Như vậy, công việc phải làm đầu tiên là thiết lập vòng lôgic (hay còn gọi là vòng ảo) bao gồm các trạm đang có nhu cầu truyền dữ liệu được xác định vị trí theo một chuỗi thứ tự mà trạm cuối cùng của chưuỗi sẽ tiếp liều sau bởi trạm đấu tiên. Mỗi trạm được biết địa chỉ của các trạm kề *trước* và *sau* nó.
- Thứ tự của các trạm trên vòng lôgic có thể độc lập với thứ tự vật lý. Các trạm không hoặc chưa có nhu cầu truyền dữ liệu thì không được đưa vào vòng lôgic và chúng chi có thể tiếp nhân dữ liệu.

Token Bus



Đường truyền vật lý

Vòng lôgic

Token Bus

Việc thiết lập vòng lôgic trong chương trình là không khó, nhưng việc duy trì nó theo trạng thái thực tế của mạng mới là khó. Cụ thể phải thực hiện được các chức năng sau :

- Bổ sung một trạm vào vòng lôgic : các trạm nằm ngoài vòng lôgic cần được xem xét định kỳ để nếu có nhu cầu truyền dữ liệu thì bổ sung vào vòng lôgic.
- Loại bỏ một trạm khỏi vòng lôgic : khi một trạm không còn nhu cầu truyền dữ liệu cần loại nó ra khỏi vòng lôgic để tối ưu hóa việc điều khiển truy nhập bằng thẻ bài.
- Quản lý lỗi : một số lỗi có thể xảv ra, chẳng hạn trùng địa chỉ (hai trạm đều nghĩ rằng đến lượt mình) hoặc "đứt vòng" không trạm nào nghĩ tới lượt mình.
- Khởi tạo vòng lôgic : khi cài đặt mạng hoặc sau khi "đứt vòng", cần phải khởi tạo lại vòng.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Token Ring

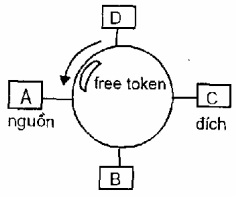
- Phương pháp này áp dụng cho Topo dạng vòng.
- Phương pháp này cũng dựa trên nguyên lý dùng thẻ bài để cấp phát quyền truy nhập đường truyền. Nhưng ở đây thẻ bài lưu chuyển theo vòng vật lý chứ không cần thiết lập vòng lôgic như đối với phương pháp Token Bus.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

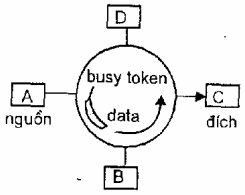
Token Ring

- Thẻ bài là một đơn vị dữ liệu đặc biệt trong đó có một bit biểu diễn trạng thái sử dụng của nó *(bận* hoặc rỗi). Một trạm muốn truyền dữ liệu thì phải đợi đến khi nhận được một *thẻ* bài "rỗi" (free) . Khi đó trạm sẽ đổi bít trạng thái của thẻ bài thành "bận" (busy) và truyền một đơn vị dữ liệu cùng với thẻ bài đi theo chiều của vòng. Giờ đây không còn thẻ bài "rỗi" trên vòng nữa, do đó các trạm có dữ liệu cần. truyền cũng phải đợi. Dữ liệu đến trạm đích sẽ được sao lại, sau đó cùng với thẻ bài đi tiếp cho đến khi quay về trạm nguồn. Trạm nguồn sẽ xóa bỏ dữ liệu và đổi bít trạng thái trở về "rỗi" và cho lưu chuyển tiếp trên vòng để các trạm khác có thể nhận được quyền truyền dữ liệu.

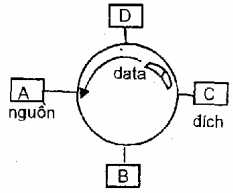
Token Ring



A có dữ liệu cần truyền đến C. Nhận được thẻ bài "rỗi", nó đổi bit trạng thái thành "bận" và truyền dữ liệu đi cùng với thẻ bài.



Trạm đích C sao dữ liệu dành cho nó và chuyển tiếp dữ liệu cùng thẻ bài đi về hướng trạm nguồn A sau khi đã gửi thông tin báo nhận vào đơn vị dữ liệu.



A nhận được dữ liệu cùng thẻ bài quay về, đổi bit trạng thái của thẻ bài thành "rỗi" và chuyển tiếp trên vòng, xóa dữ liệu đã truyền.

Token Ring

- Sự quay về lại trạm nguồn của dữ liệu và thẻ bài nhằm tạo một cơ chế báo nhận (acknowledgment) tự nhiên : trạm đích có thể gửi vào đơn vị dữ liệu phần header các thông tin về kết quả tiếp nhận dữ liệu của mình. Chẳng hạn, các thông tin đó có thể là : (1)trạm đích không tồn tại hoặc không hoạt động ; (2) trạm đích tồn tại nhưng dữ liệu không được sao chép; (3) dữ liệu đã được tiếp nhận; (4) có lỗi.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Token Ring

- Trong phương pháp này cần giải quyết hai vấn đề có thể dẫn đến phá vỡ hệ thống. Một là việc *mất thẻ bài* làm cho trên vòng không còn thể bài lưu chuyển nữa. Hai là một *thẻ bài "bận" lưu chuyển không dừng* trên vòng. Có thể có nhiều giải pháp khác nhau cho hai vấn đề này. Sau đây là một giải pháp được khuyến nghị :

- Đối với vấn đề mất thê bài, có thể qui định trước một trạm điều khiển chủ động (active monitor). Trạm này sẽ phát hiện tình trạng mất thẻ bài bằng cách dùng cơ chế ngưỡng thời gian (time-out) và phục hồi bằng cách phát đi một thẻ bài "rỗi" mới.

Token Ring

- Đối với vấn đề thẻ bài "bận" lưu chuyển không dừng, trạm monitor sử dụng một bit trên thẻ bài (gọi là monitor bit) để "đánh dấu" (đặt giá trị 1) khi gặp một thẻ bài "bận" đi qua nó. Nếu nó gặp lại một thẻ bài "bận" với bit đã đánh dấu đó thì có nghĩa là trạm nguồn đã không nhận lại được đơn vị dữ liệu của mình và thẻ bài "bận" cứ quay vòng mãi. Lúc đó, trạm monitor sẽ đổi bit trạng thái của thẻ bài thành "rỗi" và chuyển tiếp trên vòng. Các trạm còn lại trên vòng sẽ có vai trò bị động : chúng theo dõi phát hiện tình trạng sự cố của trạm monitor chủ động và thay thế vai trò đó.

Mạng máy tính
&
Hệ thống thông tin công nghiệ

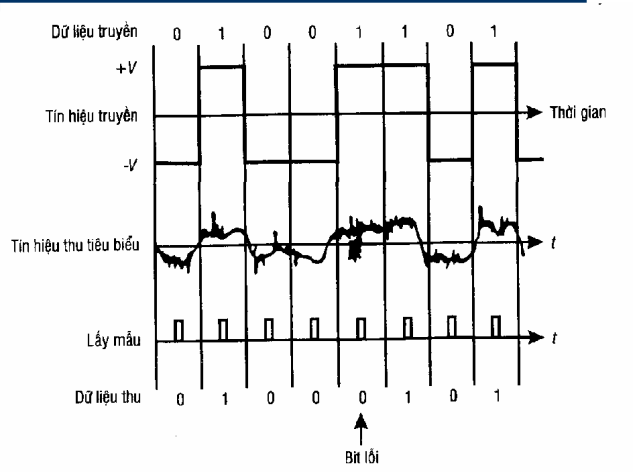
Đào Đức Thịnh
BM Kỹ thuật đo & THCN

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Môi trường truyền dẫn và chuẩn vật lý

* Tổng quan: Để truyền dữ liệu nhị phân qua một đường dây, các bit nhị phân truyền đi phải được chuyển thành các tín hiệu điện. Ví dụ có thể truyền một bit nhị phân 1 bằng cách đặt lên đường dây biên độ điện thế +V và truyền bit nhị phân 0 với mức điện thế -V. Khi nhận các tín hiệu điện này, thiết bị thu sẽ dịch +V thành 1 và -V thành 0. Trong thực tế, các tín hiệu điện được truyền đi bị suy giảm và méo dạng bởi môi trường truyền, đôi khi bộ thu không thể phân tách đâu là tín hiệu 1 và đâu là tín hiệu 0.

Môi trường truyền dẫn và chuẩn vật lý



Môi trường truyền dẫn và chuẩn vật lý

Mức độ suy giảm và méo dạng chịu ảnh hưởng nhiều nhất bởi:

- Loại môi trường truyền
- Tốc độ bit đang truyền
- Cự ly giữa hai thiết bị truyền.

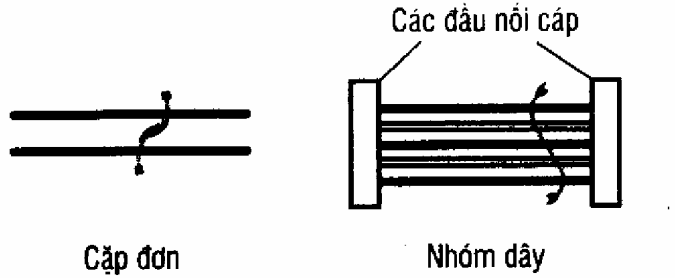
Vì sự suy giảm và méo dạng trong các loại môi trường truyền và các thành phần vật lý khác nhau là khác nhau, nên các tiêu chuẩn quốc tế đã được định nghĩa cho giao tiếp điện giữa hai chủng loại thiết bị truyền dữ liệu.

Môi trường truyền dẫn và chuẩn vật lý

Các chuẩn này không chỉ định nghĩa các mức tín hiệu điện được dùng mà còn chỉ ra cách thức áp dụng và ý nghĩa của bất kỳ tín hiệu điều khiển nào cùng với các tiêu chuẩn được dùng tại giao tiếp vật lý.Trong hầu hết các trường hợp chúng ta sẽ xem xét là giao tiếp của một máy tính với các thành phần giao tiếp truyền số liệu khác nhau, nhưng thường dùng thuật ngữ 'thiết bị đầu cuối' DTE (Da ta Terminal Equipment) thay cho 'máy tính', đó là ngụ ý cho bất kỳ loại thiết bị đầu cuối nào.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cáp hai dây không xoắn



Cáp hai dây không xoắn

Một đường truyền 2 dây không xoắn là môi trường truyền dẫn đơn giản nhất. Mỗi dây cách ly với dây kia và cả hai xuyên tự do (không xoắn nhau) qua môi trường không khí. Loại đường dây này thích hợp cho kết nối hai thiết bị cách xa nhau đến 50m dùng tốc độ bit nhỏ hơn 19,2kbps. Tín hiệu thường là mức điện thế hay cường độ dòng điện dựa vào tham chiếu điện thế đất (Ground, không cân bằng) đặt lên một dây trong khi điện thế đất được đặt vào dây kia.

Cáp hai dây không xoắn

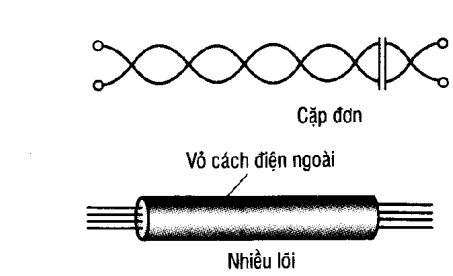
Mặc dù một đường hai dây có thể được dùng để nối hai máy tính một cách trực tiếp, nhưng thường dùng nhất là cho kết nối một DTE đến một thiết bị kết nuối mạch dữ liệu cục bộ DCE (Data Communication Equipment), ví dụ như Modem. Các kết nối như vậy thường dùng dây đa đường, cách tổ chức thông thường là cách ly riêng một dây cho mỗi tín hiệu và một dây nối đất (Ground). Bộ dây hoàn chỉnh được bọc trong một cáp nhiều lõi được bảo vệ hay dưới dạng một hộp cáp.

Cáp hai dây không xoắn

Với loại dây này cần phải cẩn thận tránh can nhiễu giữa các tín hiệu điện trong các dây dẫn kề nhau trong cùng một cáp. Hiện tượng này gọi là nhiễu xuyên âm. Ngoài ra cấu trúc không xoắn khiến chúng dễ bị thâm nhập bởi các tín hiệu nhiễu bắt nguồn từ các nguồn tín hiệu khác do bức xạ điện từ. Các yếu tố ảnh hưởng này đồng thời tạo ra giới hạn về cự ly cũng như tốc độ truyền.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cáp hai dây xoắn



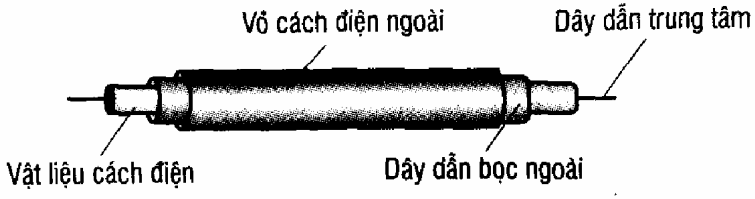
Cáp hai dây xoắn

Chúng ta có thể loại bỏ các tín hiệu nhiễu bằng cách dùng cáp xoắn đôi, trong đó một cặp dây xoắn lại với nhau. Sự xấp xỉ các đường dây tham chiếu đất và dây tín hiệu có ý nghĩa khi bất kỳ tín hiệu nhiễu nào thâm nhập thì sẽ vào cả hai dây, ảnh hưởng của chúng sẽ giảm đi bởi sự triệt lẫn nhau. Hơn nữa, nếu có nhiều cặp xoắn trong cùng một cáp thì sự xoắn của mỗi cặp trong cáp cũng làm giảm nhiễu xuyên âm.

Cáp hai dây xoắn

Các đường dây xoắn đôi cùng với mạch phát và thu thích hợp lợi dụng các ưu điểm có được từ phương pháp hình học sẽ là đường truyền tốc độ xấp xỉ 1 Mbps qua cự ly ngắn (ngắn hơn l00m) và tốc độ thấp hơn qua cự ly dài hơn. Các mạch thu phát phức tạp cho phép tốc độ cao hơn qua cự ly dài hơn. Các đường dây này được gọi là cáp xoắn đôi không bảo vệ UTP (Unshielded Twisted Pair), được dùng rộng rãi trong mạng điện thoại và trong nhiều ứng dụng truyền số liệu. Đối với các cặp xoắn được bảo vệ STP (Shielded Twisted Pair), có dùng thêm một lưới bảo vệ để giảm hơn nữa ảnh hưởng của nhiễu.

Cáp đồng trục



Cáp đồng trục

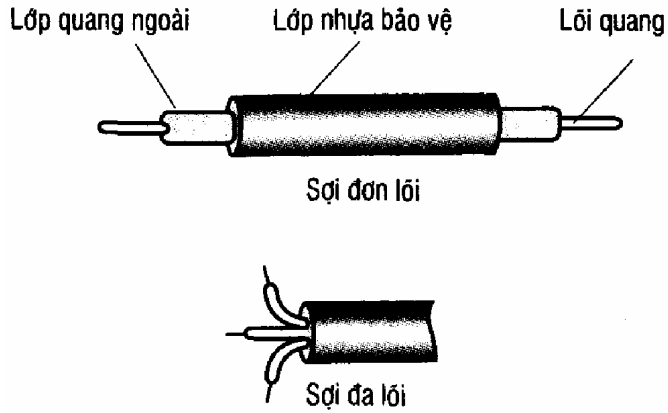
- Các yếu tố giới hạn chính đối với cáp xoắn là khả năng truyền và hiện tượng được gọi là 'hiệu ứng ngoài da'. Khi tốc độ bit truyền gia tăng, dòng điện chạy trên đường dây có khuynh hướng chỉ chạy trên bề mặt ngoài của dây dẫn, do đó dùng rất ít phần dây có sẵn. Điều này lại làm tăng trở kháng của đường dây đối với các tín hiệu có tần số cao, dẫn đến suy hao lớn đối với tín hiệu.
- Ngoài ra, với tần số cao thì năng lượng tín hiệu bị tiêu hao nhiều do ảnh hưởng bức xạ.

Cáp đồng trục tối thiểu được hai ảnh hưởng trên.

Cáp đồng trục

Dây tín hiệu trung tâm được bảo vệ hiệu quả đối với các tín hiệu xuyên nhiễu từ ngoài nhờ lưới dây bao quanh bên ngoài Chỉ suy hao lượng tối thiểu do bức xạ điện từ và hiệu ứng ngoài da do có lớp dây dẫn bao quanh. Cáp đồng trục có thể dùng với một số loại tín hiệu khác nhau, nhưng thông dụng nhất là dùng cho tốc độ 10 Mbps trên cự ly vài trăm mét, nếu dùng điều chế tốt thì có thể đạt được thông số cao hơn.

Cáp quang



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cáp quang

- Mặc dù có nhiều cải tiến nhưng các loại cáp kim loại vẫn bị giới hạn về tốc độ truyền dẫn. Cáp quang khác xa với các loại cáp trước đây, cáp quang mang thông tin dưới dạng các chùm dao động của ánh sáng trong sợi thủy tinh. Sóng ánh sáng có băng thông rộng hơn sóng điện từ, điều này cho phép cáp quang đạt được tốc độ truyền khá cao lên đến hàng trăm Mbps.
- Sóng ánh sáng cũng "miễn dịch" đối với các nhiễu điện từ và nhiễu xuyên âm. Cáp quang cũng cực kỳ hữu dụng trong việc truyền các tín hiệu tốc độ thấp trong môi trường xuyên nhiễu nặng ví dụ như điện cao thế, chuyển mạch.
- Ngoài ra còn dùng trong các nơi có nhu cầu bảo mật, vì rất khó mắc xen rẽ (câu trộm) về mặt vật lý.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cáp quang

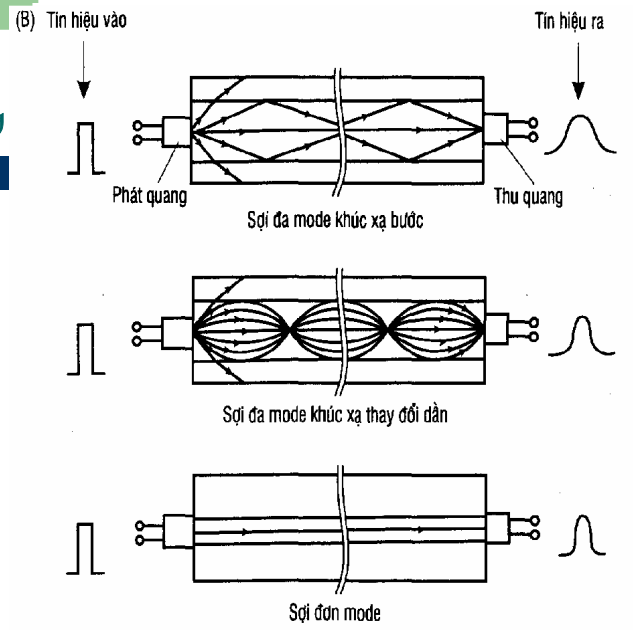
- Một cáp quang bao gồm một sợi thủy tinh cho mỗi tín hiệu được truyền, được bọc bởi một lớp phủ bảo vệ ngăn ngừa bất kỳ nguồn sáng nào từ bên ngoài. Tín hiệu ánh sáng phát ra bởi một bộ phát quang, thiết bị này thực hiện chuyển đổi các tín hiệu điện thông thường từ một đầu cuối dữ liệu thành tín hiệu quang. Một bộ thu quang được dùng để chuyển ngược lại (từ quang sang điện) tại máy thu. Thông thường bộ phát quang là diode phát quang hay laser thực hiện chuyển đổi tín hiệu điện thành tín hiệu quang. Các bộ thu dùng các photodiode cảm quang hay photo transistor.

Cáp quang

Bản thân sợi quang gồm hai phần: lõi thủy tinh và lớp phủ thủy tinh có hệ số khúc xạ thấp. ánh sáng lan truyền dọc theo lõi thủy tinh theo một trong ba cách phụ thuộc loại và bề rộng của vật liệu lõi được dùng.

ánh sáng có thể truyền trên cáp theo ba chế độ truyền:

Cáp quang



Cáp quang

Trong chế độ đa mo de khúc xạ bước-multimode *stepped index* vật liệu phủ và lõi khác nhau nhưng hệ sô khúc xạ ổn định không thay đổi. Tất cả các ánh sáng phát ra bởi diode có góc phát nhỏ hơn góc tới hạn được phản xạ tại giao tiếp giữa lớp phủ và lõi và lan truyền trong lõi Tùy vào góc phát mà ánh sáng sẽ mất một lượng thời gian để lan truyền dọc theo dây. Do đó tín hiệu nhận được có bề rộng xung rộng hơn xung gốc.

Cáp quang

Sự phân tán có thể được hạn chế bằng cách dùng vật liệu lõi có hệ số khúc xạ thay đổi hay đa mode khúc xạ tăng *dần-multimode graded index,* ánh sáng bị khúc xạ một lượng lớn khi di chuyền ra xa lõi. Điều này làm hẹp bề rộng xung của tín hiệu nhận, nhờ đó cho phép gia tăng tốc độ bit.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Cáp quang

Một cải tiến cao hơn có thể đạt được bằng cách giảm đường kính lõi đến chiều dài bước sóng đơn (3-10um) để tất cả các ánh sáng phát ra sẽ truyền theo một hướng dọc ống dẫn sóng (sợi quang cũng thường được gọi là ống dẫn sóng), và sợi quang dùng phương pháp này gọi là sợi đơn mode *-monomode fiber,* nhờ vậy bề rộng xung nhận được sẽ xấp xỉ bề rộng xung gốc , nhờ đó tăng được tốc độ truyền.

Truyền qua vệ tinh

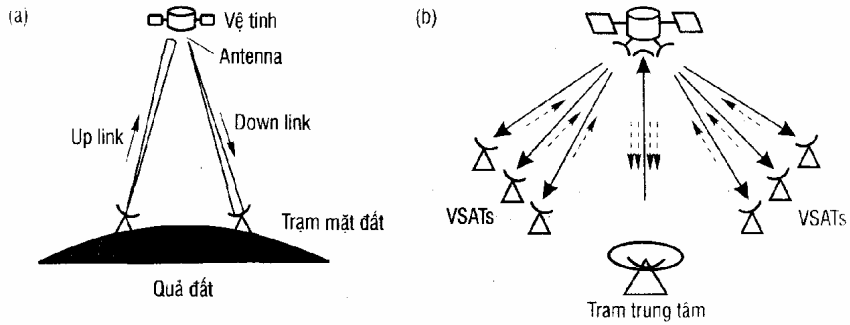
- Số liệu cũng có thể được truyền bằng cách dùng sóng điện từ qua không gian tự do như trong các hệ thống thông tin vệ tinh. -- Một chùm sóng vi ba trực xạ trên đó mang số liệu đã được điều chế, được truyền đến vệ tinh từ trạm mặt đất. Chùm sóng này được thu và được truyền lại đến các đích xác định trước nhờ một mạch tích hợp thường được gọi là transponder. Một vệ tinh có nhiều transponder, mỗi transponder đảm trách một băng tần đặc biệt. Mỗi kênh vệ tinh thông thường đều có một băng thông cực cao (500MHZ) và có thể cung cấp cho hàng trăm liên kết tốc độ cao thông qua kỹ thuật ghép kênh.

Truyền qua vệ tinh

- Các vệ tinh dùng cho mục đích liên lạc thường thuộc dạng địa tĩnh.Quĩ đạo của vệ tinh được chọn sao cho đường truyền thẳng với trạm thu phát ở mặt đất, mức độ chuẩn trực của chùm sóng truyền lại từ vệ tinh có thể không cao để tín hiệu có thể được tiếp nhận trên một vùng rộng lớn, hoặc có thể hội tụ tốt để chỉ thu được trên một vùng giới hạn. Trong trường hợp thứ hai tín hiệu có năng lượng lớn cho phép dùng các bộ thu có đường kính nhỏ hơn thường gọi là chảo parabol, là các đầu cuối có độ mở rất nhỏ hay VSAT (Very Small Aperture Terminal).

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Truyền qua vệ tinh



Truyền qua kênh viba

Các liên kết vi ba mặt đất được dùng rộng rãi để thực hiện các liên kết thông tin khi không thể hay quá đắt tiền để thực hiện một môi trường truyền vật lý. Ví dụ khi vượt sông. sa mạc, đồi núi hiểm trở .v.v. Khi chùm sóng vi ba trực xạ đi xuyên ngang môi trường khí quyển, nó có thể bị nhiễu bởi nhiều yếu tố như địa hình và các diều kiện thời tiết bất lợi. Tuy nhiên, liên lạc vi ba trực xạ xuyên môi trường khí quyển có thể dùng một cách tin cậy cho cự ly truyền dài hơn 50km.

Truyền vô tuyến tần số thấp.

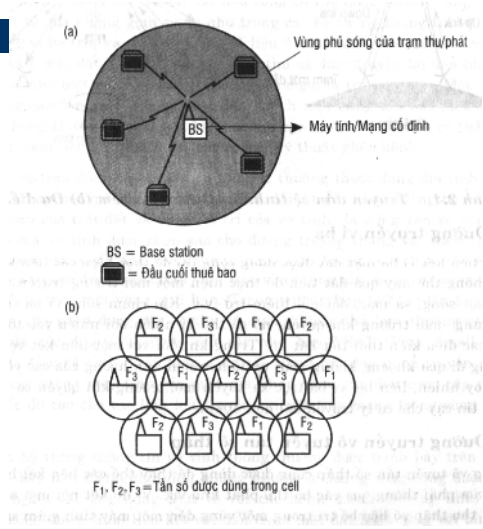
- Sóng vô tuyến tần số thấp cũng được dùng để thay thế các liên kết hữu tuyến có cự ly vừa phải thông qua các bộ thu phát khu vực. Ví dụ kết nối một số lớn các máy tính thu thập số liệu bố trí trong một vùng đến một máy tính giám sát số liệu từ xa, hay kết nối các máy tính trong một thành phố đến máy cục bộ hay ở xa.
- Sẽ rất tốn kém khi lắp đặt các cáp dẫn hữu tuyến cho các ứng dụng như vậy Sóng vô tuyến thường được dùng đế thực hiện các liên kết không dây giữa một điểm kết cuối hữu tuyến và các máy tính phán tán. Một trạm phát vô tuyến được gọi là trạm cơ bản (base station) được đặt tại điểm kết cuối hữu tuyến.

Truyền vô tuyến tần số thấp.

- Cần nhiều trạm cơ bản cho các ứng dụng yêu cầu phạm vi rộng và mật độ phân bố user cao. Phạm vi bao phủ của mỗi trạm cơ bản là giới hạn, do sự giới hạn nguồn phát cua nó, nó chỉ đủ kênh để hỗ trợ cho toàn bộ tải trong phạm vi đó.
- Phạm vi rộng hơn có thể được thực hiện bằng cách tổ chức đa trạm theo cấu trúc tế bào (cell). Trong thực tế, kích thước của mồi tế bào thay đổi và được xác định bởi các yếu tố như mật độ đầu cuối và địa hình cục bộ. Mỗi trạm cơ bán dùng một dải tần khác với trạm kế. Tuy nhiên, vì vùng phủ của mỗi trạm có giới hạn nên có thể dùng lại băng tần của nó cho các phần khác của mạng.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Truyền vô tuyến tần số thấp.



Truyền vô tuyến tần số thấp.

- Các trạm cơ bản được kết nối thành mạng hữu tuyến. Thông thường, tốc độ số liệu của mỗi máy tính trong một tế bào (cell) đạt được vài chục kbps. Dạng tổ chức tương tự có thể được dùng trong một tòa cao ốc để cung cấp các liên kết không dây cho thiết bị máy tính trong mỗi phòng.

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

* Suy hao:

- Khi một tín hiệu lan truyền theo dây dẫn thì biên độ của nó sẽ bị giảm xuống và người ta gọi là sự suy hao của tín hiệu.
- Thông thường mức độ suy giảm cho phép được qui định trên chiều dài cáp dẫn để đảm bảo rằng hệ thống nhận có thể phát hiện và dịch được tín hiệu ở máy thu.
- Nếu trường hợp cáp quá dài thì có một hay nhiều bộ khuếch đại (hay còn gọi là repeater) được thêm vào từng khoảng dọc theo cáp nhằm tiếp nhận và tái sinh tín hiệu.

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

- Sự suy giảm tín hiệu gia tăng theo một hàm của tần số, trong khi đo tín hiệu lại bao gồm một vài tần vì vậy tín hiệu sẽ bị biến dạng do các thành phần suy hao khác nhau. Để khắc phục vấn đề này, các bộ khuếch đại được thiết kế sao cho khuếch đại các tín hiệu có tần số khác nhau với hệ số khuếch đại khác nhau. Ngoài ra còn có thiết bị cân chỉnh gọi là *equalizer* được dùng để cân bằng sự suy hao trong một băng tần xác định.
- Sự suy hao và sự khuếch đại được đánh giá và đo lường bằng đơn vị decibels (dB).

dB = 10 log10 P1/P2 (dB).

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

* Băng thông của đường truyền:
- Bất kỳ một kênh hay đường truyền nào: cáp xoắn, cáp đông trục, radio đều có một băng thông xác định liên hệ với nó, băng thông chỉ ra các thành phần tần số nào của tín hiệu sẽ được truyền qua kênh mà không bị suy giảm quá nhiều.
- Băng thông B = fmax - fmin
- Công thức Nyquist xác định tốc độ tối đa của kênh trong trường hợp không nhiễu với băng thông của kênh như sau:

MTR = 2 B log2 M (bps)

B - Băng thông kênh tính bằng Hz.

M - Số mức trên một phần tử tín hiệu.

MTR (Max Transfer Rate) - Tín bằng bps

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

* Biến dạng xung do trễ:
- Tốc độ lan truyền của một tín hiệu thuần nhất dọc theo một đường truyền thay đổi tùy tần số.
- Khi truyền một tín hiệu số, nó có thể phân tích ra thành một loạt các thành phần có tần số khác nhau (phân tích Furier) các thành phần tần số khác nhau tạo nên nó sẽ đến máy thu với độ trễ pha khác nhau, dẫn đến biến dạng do trễ của tín hiệu tại máy thu.
- Sự biến dạng sẽ gia tăng khi tốc đô bit tăng. Khi các thành phần có tần số khác nhau của tín hiệu giao thoa với nhau người ta gọi đó là hiện tượng tự giao thoa.
- Méo do trễ gây khó nhăn cho việc lấy mẫu tín hiệu.

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

* Sự can nhiễu (tạp âm-noise):
- Một thông số quan trọng của đường truyền là tỉ số giữa tín hiệu và tạp âm - người ta gọi là SNR - được đo bằng dB.

SNR= 10log10(S/N) (dB)

S - Công suất tín hiệu tính bằng W.

N - Công suất tạp âm tính bằng W.

Các hiện tượng ảnh hưởng đến tín hiệu trên đường truyền

- Rõ ràng nếu tỉ số SNR cao thì chất lượng tín hiệu thu cao, SNR thấp thì chất lượng tín hiệu thu thấp.
- Tốc độ truyền tối đa của kênh có liên hệ chặt chẽ với tỉ số SNR và được xác định theo công thức Shannon-Harley:

MTR = B log2(1+S/N) (bps)

B - băng thông tính bằng Hz

S - công suất tín hiệu tính bằng W.

N - Công suất ồn tính bằng W.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các chuẩn vật lý

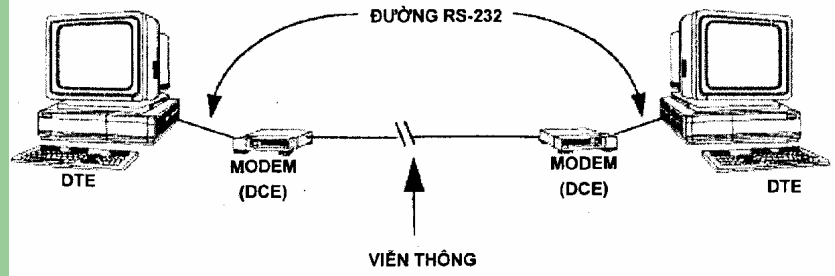
Truyền dữ liệu nối tiếp, không đồng bộ là phương pháp được sử dụng chủ yếu trong việc kết nối các DTE và DCE cũng như trong hệ thống mạng công nghiệp. Với phương pháp này, các bit được truyền từ bên gửi tới bên nhận một cách tuần tự trên cùng một đường truyền. Cũng chính vì không có một đường dây riêng biệt mang tín hiệu nhịp, nên việc đồng bộ hóa thuộc trách nhiệm do bên gửi và bên nhận thỏa thuận trên cơ sở một giao thức truyền thông. Vậy ta cần phải có chuẩn vật lý cho phần thu và phát.

RS-232

- RS-232 (tương ứng với chuẩn châu âu là CCITT V.24) lúc đầu được xây dựng phục vụ chủ yếu trong việc ghép nối điểm-điểm giữa hai thiết bị đầu cuối , giữa máy tính và máy in, hoặc giữa một thiết bị đầu cuối và và một thiết bị truyền dữ liệu.

- Mặc dù tính năng hạn chế, RS-232 là một trong các chuẩn tín hiệu có từ lâu nhất, vì thế được sử dụng rất rộng rãi. Ngày nay, mỗi máy tính cá nhân đều có một vài cổng RS-232 (cổng COM), có thể sử dụng tự do để nối với các thiết bị ngoại vi hoặc với các máy tính khác. Nhiều thiết bị công nghiệp cũng tích hợp cổng RS-232 phục vụ lập trình hoặc tham số hóa.

RS-232



RS-232

+ Đặc tính điện học:

- RS-232 sử dụng phương thức truyền không đối xứng, tức là sử dụng tín hiệu điện áp chênh lệch giữa một dây dẫn và đất.
- Mức điện áp logic được định nghĩa -3V÷ -25V mức logic "1" và +3V÷ +25V mức logic "0".
- Tốc độ truyền dẫn tối đa phụ thuộc vào chiều dài dây dẫn. Đa số các hệ thống hiện nay chỉ hỗ trợ tới tốc độ 19.2 kbps
- Chiều dài cho phép 15m (50 feet).
- Truyền số liệu Full-duplex sử dụng 3 dây: TxD, RxD, GND.
- Các tín hiệu điều khiển dùng để bắt tay (Handshaking) phần cứng là: RTS, CTS, DSR, DTR. Mức logic:+3V÷ +25V -> "1" và <0V -> "0".

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

RS-232

- Truyền không đồng bộ, cấu trúc một khung truyền bao gồm: 1 start bit, 7-8 data bit, 1- 0 parity bit, 1-1,5-2 stop bit.

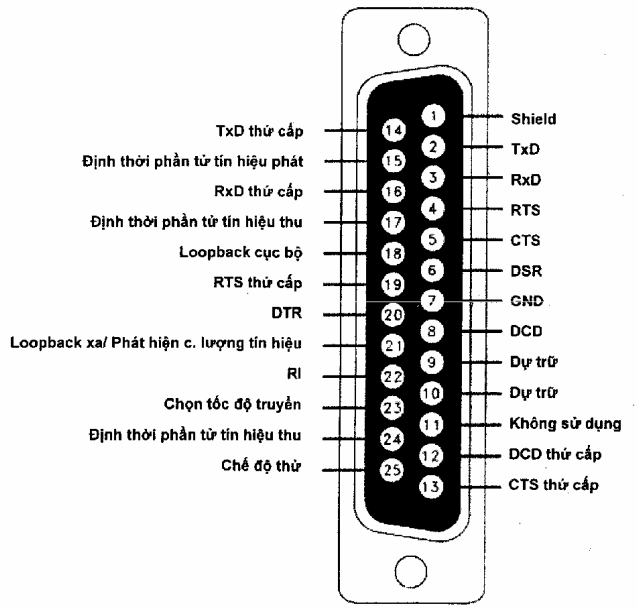
(Gần đây, sự tiến bộ trong vi mạch đã góp phần nâng cao tốc độ của cổng RS-232 lên nhiều lần so với tốc độ 19,2kbps. Hiện nay đã có những mạch thu phát đạt tốc độ 460kbd và hơn nữa, tuy nhiên tốc độ truyền dẫn thực tế lớn hơn 1 15.2 kbd theo chuẩn RS-232 trong một hệ thống làm việc dựa vào ngắt là một điều khó có thể thực hiện).

RS-232

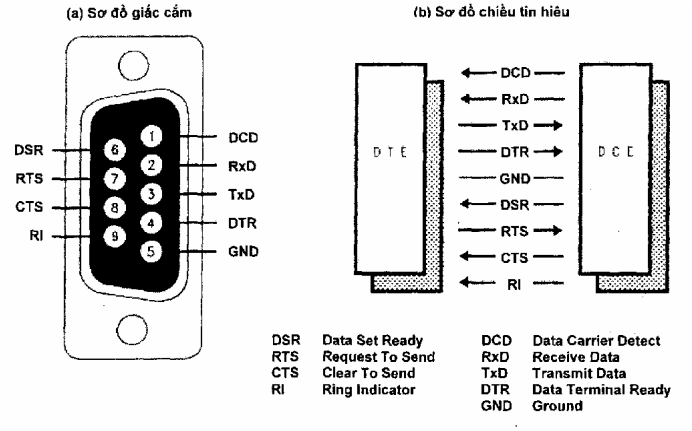
+ Giao diện cơ học:

Chuẩn RS-232 qui định ba loại giắc cắm RS-232 là DB-9, DB-25 và ALT-A, trong đó hai loại đầu được sử dụng rộng rãi hơn.

RS-232



RS-232



RS-232

- RXD (*receive Data):* Đường nhận dữ liệu.
- TXD *(Transmit Data):* Đường gửi dữ liệu.
- DTR *(Data Terminal Ready):* Báo DTE sẵn sàng .Chân DTR thường ở trạng thái ON khi thiết bị đầu cuối sẵn sàng thiết lập kênh truyền thông (tự động quay số hay tự động trả lời). DTR ở trạng thái OFF chỉ khi thiết bị đầu cuối không muốn DCE của nó chấp nhận lời gọi từ xa.
- DSR *(Data Set Ready):* Báo DCE sẵn sàng, ở chế độ trả lời, 1 tone trả lời và DSR ON sau 2 giây khi Modem nhấc máy.
- DCD (*Data Carrier Detect):* Tín hiệu này tích cực khi Modem nhận được tín hiệu từ trạm từ xa và nó duy trì trong suốt quá trình liên kết.

RS-232

- RTS *(Request To Send*): Đường RTS kiểm soát chiều truyền dữ liệu. Khi một trạm cần gửi dữ liệu, nó đóng mạch RTS sang ON để báo hiệu với modem của nó.
- CTS *(Clear To Send):* Khi CTS chuyển sang ON, Modem xác nhận là DTE có thể truyền số liệu. Quá trình ngược lại nếu đổi chiều truyền số liệu
- RI (*Ring Indicator):* Khi modem nhận được tín hiệu chuông, RI chuyển ON/OFF một cách tuần tự với chuông điện thoại để báo hiệu cho trạm đầu cuối. Tín hiệu này chỉ thị rằng một modem xa yêu cầu thiết lập liên kết dial-up.

RS-232

+ Các hạn chế khi ứng dụng RS-232 trong CN:

- Giao diện thông tin P-P hạn chế khi kết nối 1 vài thiết bị thông minh với nhau.
- Khoảng cách 15 mét là quá ngắn cho phần lớn các ứng dụng trong CN
- Tốc độ 19,2 kbps là thấp cho nhiều ứng dụng.
- Mức điện áp không thực sự tương thích với chuẩn nguồn thiết bị công nghiệp.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

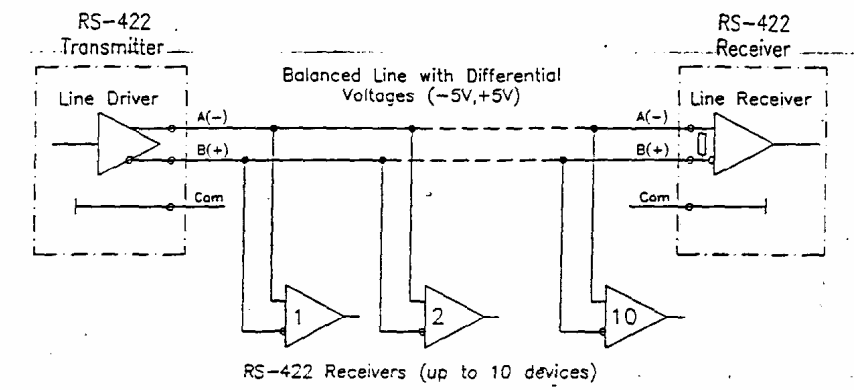
RS-422

Đây là chuẩn thông tin đối xứng sử dụng một cặp dây (A-B) cho mỗi tín hiệu, nhờ vậy mà nó có thể giảm được nhiễu, hạn chế tối đa các vấn đề do sự thay đổi điện thế đất gây ra, nó phù hợp với các ứng dụng truyền ở tốc độ cao, khoảng cách truyền xa.

- Khoảng cách truyền tối đa 1200m.
- Tốc độ truyền tối đa 10 Mbps.
- Có 1 bộ phát và 10 bộ thu trên một đường truyền.
- Mức Logic: -2V ÷ -6V -> "1",
  +2V ÷ +6V ->"0"

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

RS-422



RS-422

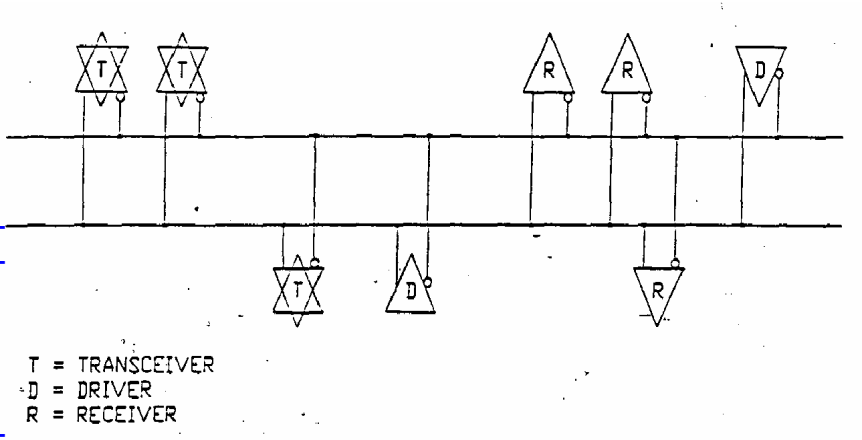
- Chuẩn RS-422 là sự sai khác ± 5V giữa hai dây do vậy có thể dùng nguồn cung cấp đơn 5V cho bộ phát.
- Truyền ở chế độ Full-duplex sử dụng 5 dây.
- Điểm cuối của đường truyền RS-422 có một trở kháng để cân bằng với trở kháng của đường dây. Giá trị thường dùng là Z0=120Ω.

RS-485

Chuẩn RS-485 tương tự như RS-422 nhưng nó có số bộ thu phát trên một đường truyền nhiều hơn. Các đặ điểm chính của RS-485 như sau:

- Khoảng cách truyền tối đa 1200m.
- Tốc độ truyền tối đa 10 Mbps.
- Có 32 bộ phát và 32 bộ thu trên một đường truyền.
- Mức Logic: -1.5V ÷ -6V -> "1",
  +1.5V ÷ +6V ->"0"

RS-485



Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

RS-485

- Chuẩn RS-485 là sự sai khác ± 5V giữa hai dây do vậy có thể dùng nguồn cung cấp đơn 5V cho bộ phát.
- Truyền ở chế độ Full-duplex sử dụng 5 dây.
- Truyền ở chế độ Half-duplex sử dụng 3 dây.
- Bộ phát của RS-485 có thể hoạt động ở 3 trạng thái: mức logic "1", mức logic "0" và trạng thái cao trở (có thể hiểu như trạng thái cấm và được điều khiển bằng một chân tín hiệu).
- Có 32 bộ phát trên một đường truyền nhưng tại một thời điểm chỉ có một cái hoạt động.

RS-485

- Các nút mạng phải được đánh địa chỉ và phải có một giao thức điều khiển truy nhập đường truyền. Các đầu ra phải có mạch hạn dòng để tránh hỏng hóc khi có xung đột xảy ra.
- Điểm cuối của đường truyền RS-485 có một trở kháng để cân bằng với trở kháng của đường dây. Giá trị thường dùng là Z0=100-120Ω.
- Để tăng khoảng cách truyền và số trạm ta cần phải dùng các bộ Repeater.

MBP (IEC 1158-2):(Manchester Coded, Bus-powered)

Đây là một kỹ thuật truyền dẫn được đưa ra trong chuẩn IEC 1158-2 cũ nhằm vào các ứng dụng điều khiển quá trình trong công nghiệp chế biến như lọc dầu, hóa chất, nơi có yêu cầu nghiêm ngặt về an toàn cháy nổ và nguồn cung cấp cho các thiết bị trường.

- MBP sử dụng mã Manchester, cho phép đồng tải nguồn trên đường bus, chế độ truyền đồng bộ và tốc độ truyền 31,25 kbit/s.
- Về mặt tín hiệu, thực chất MBP cũng sử dụng phương thức truyền đối xứng, với cáp đôi dây xoắn và trở đầu cuối là 100Ω.
- Mức điện áp tối đa được qui định nằm trong khoảng 0,75-1V.

MBP (IEC 1158-2):(Manchester Coded, Bus-powered)

Các nguyên tắc đảm bảo an toàn cho việc truyền dẫn trong môi trường dễ cháy nổ được đưa ra:

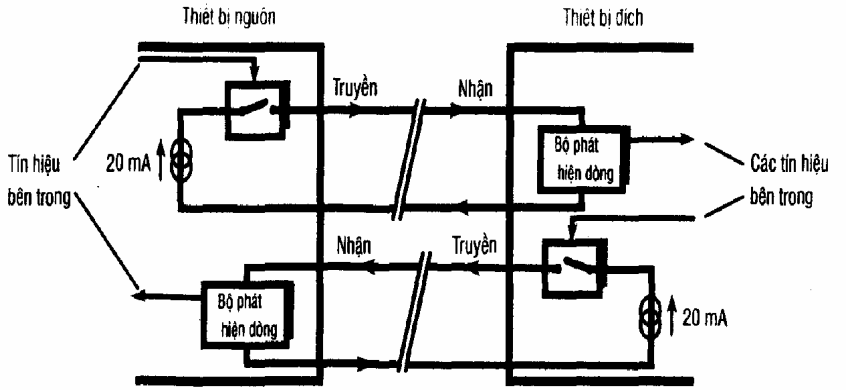
- Một đoạn mạng chỉ được phép có một bộ nguồn cung cấp điện.
- Trong trạng thái bình thường, mỗi thiết bị trường tiêu hao một dòng cơ sở cố định (> l0 mA) .
- Mỗi thiết bị trường hoạt động như một bộ tiêu hao dòng bị động.
- Mỗi đầu dây được kết thúc bằng một trở đẩu cuối bị động.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

20mA Curent-loop

- Một dạng tín hiệu khác có thể chọn bên cạnh RS-232 là giao tiếp vòng 20mA. Tên của giao tiếp này ngụ ý rằng dùng tín hiệu là dòng điện thay cho điện áp. Mặc dù không mở rộng tốc độ nhưng nó tăng khoảng cách vật lý giữa hai thiết bị thông tin.
- Hoạt động chính là trạng thái của chuyển mạch được điều khiên bởi luồng bit dữ liệu truyền: chuyển mạch đóng tương ứng với bit 1, do đó cho dòng 20ma qua, và ngược lại chuyển mạch mở cho bit 0, do đó không cho dòng 20ma qua. Tại đầu thu dòng diện được phát hiện bởi mạch cảm biến dòng và các tín hiệu nhị phân được tái tạo lại. Giao tiếp này loại bỏ nhiễu tốt hơn so với giao tiếp điều khiển bằng điện áp, phù hợp với đường dây dài (đến lkm), nhưng tốc độ vừa phải.

20mA Curent-loop



Truyền không đối xứng

- Một tín hiệu chỉ có một dây truyền tín hiệu điện áp so với dây đất.
- Dây đất chung cho nhiều tín hiệu khác nhau vì vậy đòi hỏi có trở kháng nhỏ.
- Khi truyền ở khoảng cách xa không có chung điện thế đất.

Truyền đối xứng

- Sử dụng một cặp dây cho một tín hiệu.
- Triệt tiêu được sự ảnh hưởng của nhiễu.
- Tránh được sự sai khác điện thề đất.
- Thích hợp cho trường hợp truyền ở tốc độ cao, khảng cách xa và trong môi trường có nhiễu lớn.

Các tiêu chuẩn mã hoá đường truyền

- Tần số của tín hiệu.
- Thông tin đồng bộ hoá.
- Triệt tiêu dòng một chiều.
- Bền vững với nhiễu và có khả năng nhận biết lỗi.

NRZ,RZ

- NRZ *(Non-return To Zero), RZ (Return to Zero)* là một trong những phương pháp được sử dụng phổ biến nhất trong các hệ thống.
- NRZ và RZ đều là các phương pháp điều chế biên độ xung.
- NRZ mức bit "0" và "1" được mã hóa với hai mức biên độ tín hiệu khác nhau, mức tín hiệu này không thay đổi trong suốt chu kỳ bit T (một nhịp bus). Cái tên NRZ được sử dụng, bởi mức tín hiệu không quay trở về không sau mỗi nhịp. Các khả năng thể hiện hai mức có thể là:
- Đất và điện áp dương
- Điện áp âm và đất
- Điện áp âm và điện áp dương cùng giá trị (tín hiệu lưỡng cực)

NRZ,RZ

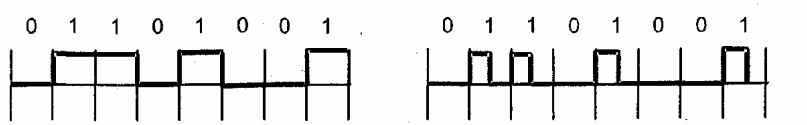
- Một trong những ưu điểm của phương pháp NRZ là tín hiệu có tần số thường thấp hơn nhiều so với tần số nhịp bus.
- Phương pháp này không thích hợp cho việc đồng bộ hóa, bởi một dãy bit "0" hoặc "1" liên tục không làm thay đổi mức tín hiệu.
- Tín hiệu không được triệt tiêu dòng một chiều ngay cả khi sử dụng tín hiệu lưỡng cực, nên không có khả năng đồng tải nguồn.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

NRZ,RZ

- Phương pháp RZ *(Return to Zero)* cũng mã hóa bít "0" và "1" với hai mức tín hiệu khác nhau giống như ở NRZ. Tuy nhiên, như cái tên của nó hàm ý mức tín hiệu cao chỉ tồn tại trong nửa đầu của chu kỳ bit T, sau đó quay trở lại "0".
- Tần số cao nhất của tín hiệu chính bằng tần số nhịp bus.
- Giống như NRZ, tín hiệu mã RZ không mang thông tin đồng bộ hóa, không có khả năng đồng tải nguồn.

NRZ,RZ



NRZ: 1 ứng với mức tín hiệu cao, 0 với mức thấp trong suốt chu kỳ bit

RZ: 1 ứng với mức tín hiệu cao trong nửa chu kỳ bit T, 0 với mức thấp trong suốt chu kỳ bit

Manchester

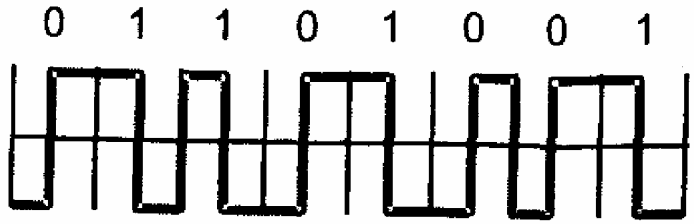
- Mã Manchester và các dạng dẫn xuất của nó không những được sử dụng rất rộng rãi trong truyền thông công nghiệp, mà còn phổ biến trong các hệ thống truyền dữ liệu khác.
- Thực chất, đây là một trong các phương pháp điều chế pha xung, tham số thông tin được thể hiện qua các sườn xung. Bít "1" được mã hóa bằng sườn lên, bit "0" bằng sườn xuống của xung ở giữa chu kỳ bit T, hoặc ngược lại (Manchester-II).

Manchester

- Đặc điểm của tín hiệu là có tần số tương đương với tần số nhịp bus, các xung của nó có thể sử dụng trong việc đồng bộ hóa giữa bên gửi và bên nhận.
- Sử dụng tín hiệu lưỡng cực, dòng một chiều sẽ bị triệt tiêu. Do đó phương pháp này thích hợp với các ứng dụng đòi hỏi khả năng đồng tải nguồn.
- Một điểm đáng chú ý nữa là do sử dụng sườn xung, mã Manchester rất bền vững đối với nhiễu bên ngoài. Nhưng ngược lại, nhiễu xạ của tín hiệu cũng tương đối lớn bởi tần số cao.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Manchester



Manchester-II: 1 ứng với sườn xuống, 0 ứng với sườn lên của xung ở giữa chu kỳ bit

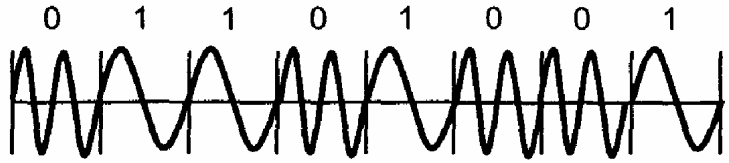
FSK

- Trong phương pháp điều chế dịch tần số FSK *(Frequency Shift Keying)*, hai tần số khác nhau được dùng để mã hóa các trạng thái logic "0" và "1".
- Đây chính là phương pháp điều chế tần số tín hiệu mang, hay truyền tải dải mang.
- Tín hiệu có dạng hình sin, các tần số có thể bằng hoặc là bội số tần số nhịp bus nên có thể dùng để đồng bộ nhịp.
- ưu điểm tiếp theo của phương pháp này là độ bền vững đối với tác động của nhiễu.

FSK

- Nhờ tính chất điều hòa của tín hiệu mà dòng một chiều được triệt tiêu, nên có thể sử dụng chính đường truyền để đồng tải nguồn nuôi các thiết bị kết nối mạng.
- Nhược điểm của FSK là tần số tín hiệu tương đối cao. Điều này một mặt dẫn đến khả năng gây nhiễu mạnh đối với bên ngoài và mặt khác hạn chế việc tăng tốc độ truyền. Thực tế, phương pháp này chỉ được sử dụng cho các hệ thống có tốc độ truyền tương đối thấp.

FSK



FSK: 0 và 1 ứng với các tần số khác nhau

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các nguyên nhân gây ra lỗi

- Các hiện tượng tĩnh.

- ồn nhiệt.

- Các hiện tượng ngẫu nhiên

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các định nghĩa

+ Tỉ lệ bit lỗi: *Tỉ lệ bit lỗi p* là thước đo đặc trưng cho độ nhiễu của kênh truyền dẫn, được tính bằng tỉ lệ giữa số bit bị lỗi trên tổng số bit được truyền đi. Nói một cách khác, tỉ lệ bit lỗi chính là xác suất một bit truyền đi bị lỗi. Lưu ý rằng, tỉ lệ bit lỗi xấu nhất không phải là 1, mà là 0,5. Trong trường hợp $p = 1$ tức là bất cứ bit nào truyền đi cũng bị sai lệch, ta chỉ việc đảo các bit để khôi phục lại dữ liệu. Khi $p = 0,5$ tức xác suất cứ hai bit truyền đi lại có một bit bị lỗi thì đường truyền này hoàn toàn không sử dụng được, bởi theo lý thuyết thông tin thì không thể có một phương pháp bảo toàn dữ liệu nào có thể áp dụng tin cậy, có hiệu quả. Trong kỹ thuật, $p = 10^{-4}$ là một giá trị thường chấp nhận được. Một đường truyền có tỉ lệ bit lỗi như vậy có thể thực hiện được tương đối dễ dàng.

Các định nghĩa

+ Tỉ lệ lỗi còn lại: Tỉ lệ lỗi còn lại *R* là thông số đặc trưng cho độ tin cậy dữ liệu của một hệ thống truyền thông, sau khi đã thực hiện các biện pháp bảo toàn kể cả truyền lại trong trường hợp phát hiện ra lỗi. Tỉ lệ lỗi còn lại được tính bằng tỉ lệ giữa số bức điện còn bị lỗi không phát hiện được trên tổng số bức điện đã được truyền. Đương nhiên, giá trị này không những phụ thuộc vào tỉ lệ bit lỗi và phương pháp bảo toàn dữ liệu mà còn phụ thuộc vào chiều dài trung bình của các bức điện. Một bức điện càng dài thì xác suất lỗi càng lớn.

Các định nghĩa

+ Thời gian trung bình giữa hai lần lỗi: Tỉ lệ lỗi còn lại là một thông số tương đối khó hình dung, vì vậy trong thực tế người ta hay xét tới thời gian trung bình giữa hai lần lỗi *TMTBF (MTBF = Mean Time Between Failures)*. Thông số này có liên quan chặt chẽ tới giá trị tỉ lệ lỗi còn lại:

$$TMTBF = n/(v * R)$$

Với *n* là chiều dài bức điện tính bằng bit và *v* là tốc độ truyền tính bằng bit/s. Giả sử một bức điện có chiều dài n = 100 bit được truyền liên tục với tốc độ 1200 bit/s, quan hệ giữa tỉ lệ bit lỗi và thời gian trung bình giữa hai lần lỗi sẽ được thể hiện như sau:

R	TMTBF:
10^{-6}	1 ngày
10^{-10}	26 năm
10^{-14}	260 000 năm

Các định nghĩa

+ Khoảng cách Hamming (Hamming Distance, HD): Khoảng cách Hamming (gọi theo nhà khoa học Mỹ R.W. Hamming) là thông số đặc trưng cho độ bền vững của một mã dữ liệu, hay nói cách khác chính là khả năng phát hiện lỗi của một phương pháp bảo toàn dữ liệu HD có giá trị bằng số lượng bít lỗi tối thiểu mà không đảm bảo chắc chắn phát hiện được trong một bức điện. Nếu trong một bức điện chỉ có thể phát hiện một cách chắc chắn *k* bit bị lỗi, thì HD = *k*+ 1 . Ví dụ, nếu một lỗi duy nhất có thể phát hiện được một cách chắc chắn (như trong phương pháp dùng parity bit 1 chiều), thì khoảng cách Hamming là 2. Đây là giá trị tối thiểu mà một phương pháp truyền đòi hỏi. Các hệ thống bus trường thông dụng thường có khoảng cách Hamming là 4, các hệ thông đạt độ tin cậy rất cao với HD = 6.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Các định nghĩa

+ Hiệu suất truyền dữ liệu: Hiệu suất truyền dữ liệu *E* là một thông số đặc trưng cho việc sử dụng hiệu quả các bức điện phục vụ chức năng bảo toàn dữ liệu, được tính bằng tỉ lệ số bit mang thông tin nguồn (bit dữ liệu không bị lỗi trên toàn bộ số bit được truyền. Ta có:

E = m (l-p)n/n

m - Số lượng bit dữ liệu trong mỗi bức điện

n - Chiều dài bức điện

p - Tỉ lệ bit lỗi

Phát hiện và sửa lỗi

Phần lớn các phương pháp phát hiện lỗi là công thêm vào bản tin các bit vừa giúp để biểu diễn bản tin vừa để phát hiện lỗi.

Phát hiện và sửa lỗi

Có hai cách sửa lỗi cho bản tin:

+ Sửa lỗi có phản hồi: Bộ thu sẽ phân tích và phát hiện ra các lỗi có trong bản tin được gửi đi từ bộ truyền. Đã được định nghĩa ở trong giao thức, bộ thu sẽ yêu cầu bộ phát gửi lại bản tin. Phần lớn các giao thức mạng máy tính và công nghiệp sử dụng cách này.

+ Sửa lỗi không có phản hồi: Trong phương pháp này bộ thu không chỉ phát hiện ra lỗi có ở trong bản tin mà nó còn phục hồi lại bản tin đúng từ các thông tin sửa lỗi đi kèm theo. Cách này thường được sử dụng khi truyền ở khoảng cách lớn trong không gian, ở đây thời gian đòi hỏi cho việc truyền lại bản tin là quá lớn, hay trong hệ truyền thông tin theo một chiều (phát thanh, truyền hình).

Kiểm tra chẵn lẻ ký tự

- Trước khi truyền đi một ký tự, bên phát sẽ căn cứ vào mức độ là chẵn (EVEN) hay lẻ (ODD) để tính toán một bit công thêm vào ký tự.

Lẻ (ODD): số bit "1" trong ký tự là lẻ.

Chẵn (EVEN): số bit "1" trong ký tự là chẵn.

- Phương pháp này cung cấp hiệu quả phát hiện lỗi thấp (HD=2), khi có 2 bit cùng thay đổi giá trị thì không phát hiện được.

- Phương pháp này phù hợp trong trường hợp đơn giản, giá thành thực hiện thấp, cho phép kiểm tra nhanh độc hính xác của dữ liệu, dễ dàng tính nhẩm để kiểm tra.

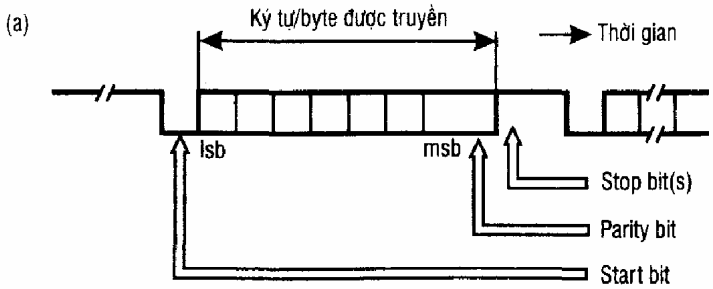
Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Kiểm tra chẵn lẻ ký tự

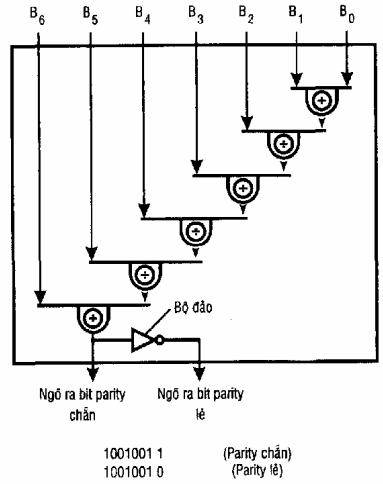
- Mặc dù có nhiều hạn chế, nhưng nó vẫn được dùng trong các ứng dụng không đòi hỏi cao như truyền giữa máy tính và máy in, hay trong các ứng dụng mà các thiết bị đặt gần nhau và trong môi trường có độ ồn thấp.

- Kiểm tra chẵn lẻ phát hiện được 60% lỗi.

Kiểm tra chẵn lẻ ký tự



Kiểm tra chẵn lẻ ký tự



Kiểm tra khối

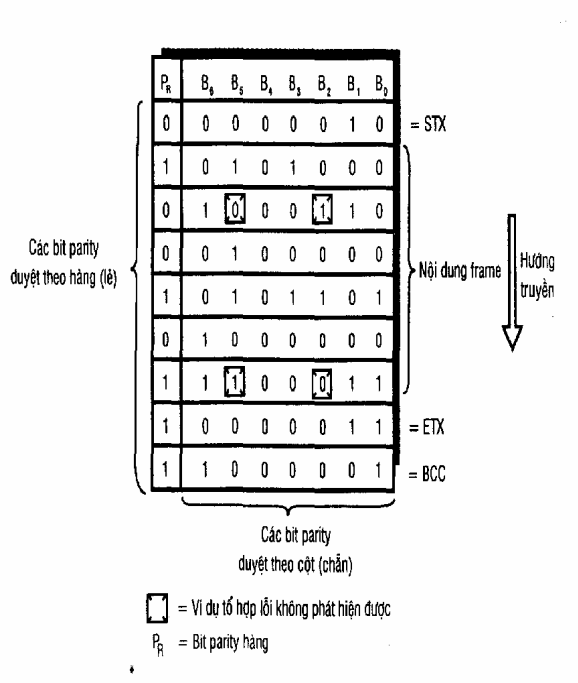
- Kiểm tra chẵn lẻ khối:

(BCC-Block Check Character ; LRC-Longitudinal Redundancy Check)

Trong cách kiểm tra khối bản tin các ký tự được xem như là mảng bit hai chiều. Một bit chẵn lẻ được gắn thêm cho mỗi ký tự. Sau một số lượng đã định trước các ký tự, 1ký tự mà nó thực hiện việc kiểm tra chẵn lẻ của cột sẽ được truyền. Mặc dù tốt hơn nhưng phương pháp này cũng không phát hiện hết lỗi.(PP này có HD=4).

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Kiểm tra khối



Kiểm tra khối

- Lấy Check-sum toán học: Nó đơn giản là lấy tổng của tất cả các ký tự trong khối. Nó cung cấp khả năng kiểm tra tốt hơn nhưng đòi hỏi thêm hai byte khi truyền.

Kiểm tra CRC

- Phương pháp có tên như vậy do các bit trong một bản tin được dịch chuyển quay vòng qua một thanh ghi. Nó cũng còn được gọi là phương pháp mã đa thức (polynomial code) vì có sử dụng khái niệm đa thức đại số quen thuộc.

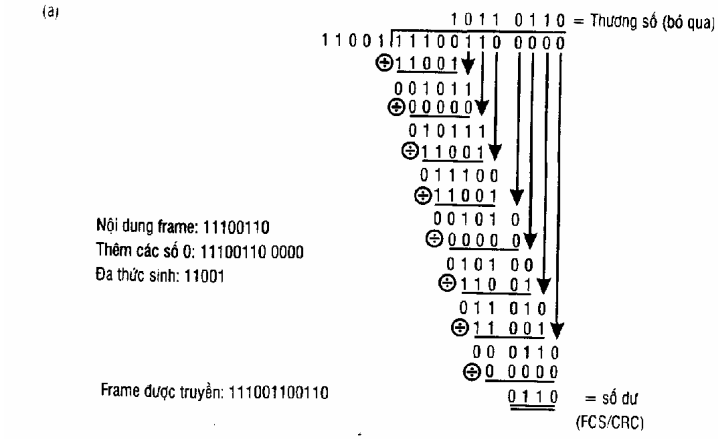
- Một xâu bit bất kỳ được xem được xem như là một tập hợp các hệ số (0 và 1) của một đa thức đại số. Nếu xâu bit gồm k bits thì đa thức tương ứng sẽ có bậc k-1, gồm k số hạng từ x0 đến xk-1.

Kiểm tra CRC

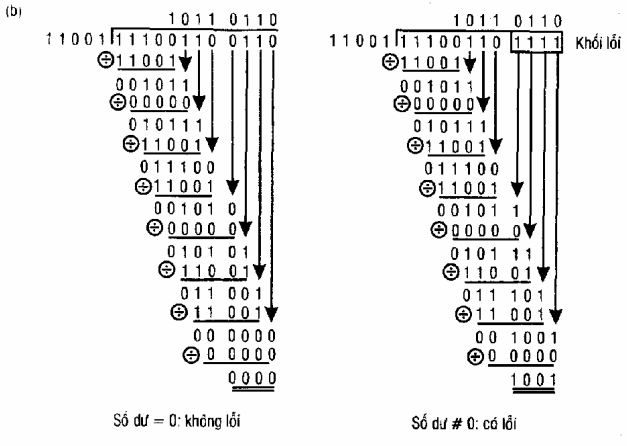
- Để tìm tập bits kiểm tra (được gọi là checksum) thích hợp để ghép vào sâu bit cần truyền đi sao cho bên nhận có thể kiểm soát đượclỗi, tư tưởng của phương pháp CRC là:
- Chọn trước một đa thức (gọi là đa thức sinh - Generator polynomial) G(x) với hệ số cao nhất và thấp nhất bằng 1.
- Checksum được tìm thoả mãn điều kiện: đa thức tương ứng với sâu ghép (Gốc và checksum) phải chia hết (Modulo 2) cho G(x).
- Khi nhận tin để kiểm soát lỗi, lấy đa thức tương ứng với sâu bit nhận được chia cho G(x). Nếu chia không hết thì khẳng định là đã có lỗi. Nếu chia hết thì chưa thể khẳng định là đúng.

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Kiểm tra CRC



Kiểm tra CRC



Kiểm tra CRC

Hiện nay có một số đa thức sinh chuẩn:

CRC - 12 = x12 + x11 + x3 + x2 + x + 1

CRC - 16 = x16 + x15 + x2 + 1

CRC - CCITT = x16 + x12 + x5 + 1

CRC-32=
x32+x26+x23+x16+x12+x11+x10+x8+x7+x5+x4+x2+x+1

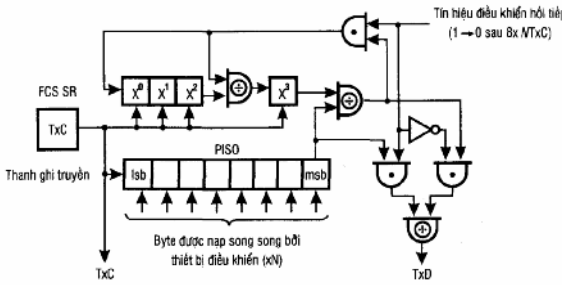
Kiểm tra CRC

Phương pháp này có hiệu quả phát hiệ lỗi tốt. Với CRC-16 và CRC-CCITT như sau:

- Lỗi 1 bit :100%.
- Lỗi 2 bit: 100%.
- Lỗi lẻ bit: 100%.
- Khối lỗi <16 bit: 100%.
- Khối lỗi >16 bit: 99,9969%
- Các lỗi khác: 99,9984%

Simpo PDF Merge and Split Unregistered Version - http://www.simpopdf.com

Kiểm tra CRC

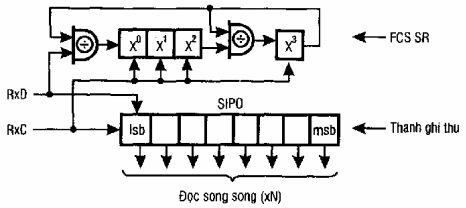


Xung TxC	Thanh ghi truyền								FCS SR			
	lsb							msb	X^0	X^1	X^2	X^3
0	0	1	1	0	0	1	1	1	0	0	0	0
1	0	0	1	1	0	0	1	1	1	0	0	1
2	0	0	0	1	1	0	0	1	0	1	0	0
3	0	0	0	0	1	1	0	0	1	0	1	1
4	0	0	0	0	0	1	1	0	1	1	0	0
5		0	0	0	0	0	1	1	0	1	1	0
6			0	0	0	0	0	1	1	0	1	0
7				0	0	0	0	0	1	1	0	0
8					0	0	0	0	0	1	1	0
9						0	0	0		0	1	1
10							0	0			0	1
11								0				0

Thời gian

TxD

Kiểm tra CRC



RxC	RxD	Thanh ghi thu								FCS SR			
		lsb							msb	X^0	X^1	X^2	X^3
0	1	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1								1	0	0	0
2	1	1	1							1	1	0	0
3	0	1	1	1						1	1	1	0
4	0	0	1	1	1					0	1	1	1
5	1	0	0	1	1	1				1	0	1	0
6	1	1	0	0	1	1	1			1	1	0	1
7	0	1	1	0	0	1	1	1		0	1	1	1
8	0	0	1	1	0	0	1	1	1	1	0	1	0
9	1	Byte được đọc bởi thiết bị điều khiển								0	1	0	1
10	1									0	0	1	1
11	0									0	0	0	0
12										0	0	0	0

Thời gian

Số dư = 0

SÁCH

Nhập môn Mạng máy tính

PHẦN I. NHẬP MÔN LÝ THUYẾT MẠNG

Chương 1. Tổng quan về công nghệ mạng máy tính và mạng cục bộ

I. Lịch sử mạng máy tính

II. Giới thiệu mạng máy tính

2.1. Định nghĩa mạng máy tính và mục đích của việc kết nối mạng

2.1.1. Nhu cầu của việc kết nối mạng máy tính.

2.1.2. Định nghĩa mạng máy tính

2.2. Đặc trưng kỹ thuật của mạng máy tính

2.2.1. Đường truyền

2.2.2. Kỹ thuật chuyển mạch

2.2.3. Kiến trúc mạng

2.2.4. Hệ điều hành mạng

2.3. Phân loại mạng máy tính

2.3.1. Phân loại mạng theo khoảng cách địa lý :

2.3.2. Phân loại mạng theo kỹ thuật chuyển mạch

2.3.3. Phân loại theo kiến trúc mạng sử dụng

2.3.4. Phân loại theo hệ điều hàng mạng

2.4. Giới thiệu các mạng máy tính thông dụng nhất

2.4.1. Mạng cục bộ

2.4.2. Mạng diện rộng với kết nối LAN TO LAN.

2.4.3. Liên mạng INTERNET.

2.4.4. Mạng INTRANET

III. Mạng cục bộ, kiến trúc mạng cục bộ

3.1. Mạng cục bộ

3.2. Kiến trúc mạng cục bộ.

3.3. Các phương pháp truy cập đường truyền vật lý

3.3.1 Phương pháp đa truy nhập sử dụng sóng mang có phát hiện xung đột CSMA/CD

3.3.2. Phương pháp Token Bus.

3.3.2. Phương pháp Token Ring.

IV. Chuẩn hoá mạng máy tính

4.1. Vấn đề chuẩn hoá mạng và các tổ chức chuẩn hoá mạng

4.2. Mô hình tham chiếu OSI 7 lớp

4.3. Các chuẩn kết nối thông dụng nhất IEEE 802.X và ISO 8802.X

Chương 2. Các thiết bị mạng thông dụng và các chuẩn kết nối vật lý

I. Các thiết bị mạng thông dụng

1.1. Các loại cáp truyền

1.1.1. Cáp đôi dây xoắn (Twisted pair cable)

1.1.2. Cáp đồng trục (Coaxial cable) băng tần cơ sở

1.1.3. Cáp đồng trục băng rộng (Broadband Coaxial Cable)

1.1.4. Cáp quang

1.2. Các thiết bị ghép nối.

1.2.1. Card giao tiếp mạng (Network Interface Card viết tắt là NIC).

1.2.2. Bộ chuyển tiếp (REPEATER)

1.2.3. Các bộ tập trung (HUB).

1.2.4. Switching Hub

1.2.5. Modem

1.2.6. Router

II. Một số kiểu nối mạng thông dụng và các chuẩn

2.1.Các thành phần thông thường trên một mạng cục bộ gồm có

2.2. Kiểu 10BASE5

2.3. Kiểu 10BASE2

2.4. Kiểu 10BASE-T

2.5. Kiểu 10BASE-F

Chương 3. Giới thiệu giao thức TCP/IP

I. Giao thức IP.

1.1. Họ giao thức TCP/IP

1.2. Chức năng chính của - Giao thức liên mạng IP(v4)

1.2.1. Địa chỉ IP

1.2.2. Cấu trúc gói dữ liệu IP

1.2.3. Phân mảnh và hợp nhất các gói IP

1.2.4. Định tuyến IP

II. Giao thức lớp chuyển tải (Transport Layer)

2.1. Giao thức TCP

2.2 Cấu trúc gói dữ liệu TCP

2.3. Thiết lập và kết thúc kết nối TCP

Chương 4. Giao thức (4)

I. Chức năng của giao thức

II. Giao thức trong kiến trúc phân tầng. Chồng giao thức

III. Các giao thức chuẩn

IV. Cài đặt và gỡ bỏ giao thức

Chương 5. Quản trị mạng (6)

I. Khái quát.

II. Quản lý tài khoản mạng

2.1. Khái niệm, các loại Account. Kích hoạt, huỷ bỏ, vô hiệu hoá tạm thời tài khoản.

2.2. Chiến lược quản trị tài khoản. Khái niệm Group, Profile.

2.3. Theo dõi hiệu suất mạng. Hiện tượng tắc nghẽn. Các công cụ quản trị. Giao thức SNMP.

2.4. Duy trì nhật ký mạng.

III. Phòng ngừa mất dữ liệu

3.1. Các loại nguy cơ đe doạ dữ liệu.

3.2. Hệ thống sao lưu trên băng từ. Lịch biểu sao lưu.

3.3. Các hệ thống dung lỗi.

PHẦN II. QUẢN TRỊ MẠNG

Chương 1. Tổng quan về công nghệ mạng máy tính và mạng cục bộ

Chương này cung cấp các khái niệm, các kiến thức cơ bản nhất về mạng máy tính và phân loại mạng máy tính. Các nội dung giới thiệu mang tính tổng quan về mạng cục bộ, kiến trúc mạng cục bộ, phương pháp truy cập trong mạng cục bộ và các chuẩn vật lý về các thiết bị mạng. Đây là những kiến thức cơ bản rất hữu ích do phạm vi sử dụng của mạng cục bộ là đang phổ biến hiện nay. Hầu hết các cơ quan, tổ chức, công ty có sử dụng công nghệ thông tin đều thiết lập mạng cục bộ riêng.

Các khái niệm, nội dung cơ bản trong chương 1 cần phải nắm vững đối với tất cả các học viên vì chúng sẽ được sử dụng nhiều trong các chương tiếp theo.

I. Lịch sử mạng máy tính

Internet bắt nguồn từ đề án ARPANET (Advanced Research Project Agency Network) khởi sự trong năm 1969 bởi Bộ Quốc phòng Mỹ (American Department of Defense). Đề án ARPANET với sự tham gia của một số trung tâm nghiên cứu, đại học tại Mỹ (UCLA, Stanford, . . .) nhằm mục đích thiết kế một mạng WAN (Wide Area Network) có khả năng tự bảo tồn chống lại sự phá hoại một phân mạng bằng chiến tranh nguyên tử. Đề án này dẫn tới sự ra đời của nghi thức truyền IP (Internet Protocol). Theo nghi thức này, thông tin truyền sẽ được đóng thành các gói dữ liệu và truyền trên mạng theo nhiều đường khác nhau từ người gửi tới nơi người nhận. Một hệ thống máy tính nối trên mạng gọi là **Router** làm nhiệm vụ tìm đường đi tối ưu cho các gói dữ liệu, tất cả các máy tính trên mạng đều tham dự vào việc truyền dữ liệu, nhờ vậy nếu một phân mạng bị phá huỷ các **Router** có thể tìm đường khác để truyền thông tin tới người nhận. Mạng ARPANET được phát triển và sử dụng trước hết trong các trường đại học, các cơ quan nhà nước Mỹ, tiếp theo đó, các trung tâm tính toán lớn, các trung tâm truyền vô tuyến điện và vệ tinh được nối vào mạng, . . . trên cơ sở này, ARPANET được nối với khắp các vùng trên thế giới.

Tới năm 1983, trước sự thành công của việc triển khai mạng ARPANET, Bộ quốc phòng Mỹ tách một phân mạng giành riêng cho quân đội Mỹ (MILNET). Phần còn lại, gọi là NSFnet, được quản lý bởi NSF (National Science Foundation) NSF dùng 5 siêu máy tính để làm **Router** cho mạng, và lập một tổ chức không chính phủ để quản lý mạng, chủ yếu dùng cho đại học và nghiên cứu cơ bản trên toàn thế giới. Tới năm 1987, NSFnet mở cửa cho cá nhân và cho các công ty tư nhân (BITnet), tới năm 1988 siêu mạng được mang tên INTERNET.

Tuy nhiên cho tới năm 1988, việc sử dụng INTERNET còn hạn chế trong các dịch vụ truyền mạng (FTP), thư điện tử(E-mail), truy nhập từ xa (TELNET) không thích ứng với nhu cầu kinh tế và đời sống hàng ngày. INTERNET chủ yếu được dùng trong môi trường nghiên cứu khoa học và giảng dạy đại học. Trong năm 1988, tại trung tâm nghiên cứu nguyên tử của Pháp CERN (Centre Européen de Recherche Nuclaire) ra đời đề án Mạng nhện thế giới WWW

(World Wide Web). Đề án này, nhằm xây dựng một phương thức mới sử dụng INTERNET, gọi là phương thức Siêu văn bản (HyperText). Các tài liệu và hình ảnh được trình bày bằng ngôn ngữ HTML (HyperText Markup Language) và được phát hành trên INTERNET qua các hệ chủ làm việc với nghi thức HTTP (HyperText Transport Protocol). Từ năm 1992, phương thức làm việc này được đưa ra thử nghiệm trên INTERNET, rất nhanh chóng, các công ty tư nhân tìm thấy qua phương thức này cách sử dụng INTERNET trong kinh tế và đời sống. Vốn đầu tư vào INTERNET được nhân lên hàng chục lần. Từ năm 1994 INTERNET trở thành siêu mạng kinh doanh. Số các công ty sử dụng INTERNET vào việc kinh doanh và quảng cáo lên gấp hàng nghìn lần kể từ năm 1995. Doanh số giao dịch thương mại qua mạng INTERNET lên hàng chục tỉ USD trong năm 1996 . . .

Với phương thức siêu văn bản, người sử dụng, qua một phần mềm truy đọc (Navigator, Browser), có thể tìm đọc tất cả các tài liệu siêu văn bản công bố tại mọi nơi trên thế giới (kể cả hình ảnh và tiếng nói). Với công nghệ WWW, chúng ta bước vào giai đoạn mà mọi thông tin có thể có ngay trên bàn làm việc của mình. Mỗi công ty hoặc người sử dụng, được phân phối một trang cội nguồn (Home Page) trên hệ chủ HTTP. Trang cội nguồn, là siêu văn bản gốc, để tự do có thể tìm tới tất cả các siêu văn bản khác mà người sử dụng muốn phát hành. Địa chỉ của trang cội nguồn được tìm thấy từ khắp mọi nơi trên thế giới. Vì vậy, đối với một xí nghiệp, trang cội nguồn trở thành một văn phòng đại diện điện tử trên INTERNET. Từ khắp mọi nơi, khách hàng có thể xem các quảng cáo và liên hệ trực tiếp với xí nghiệp qua các dòng siêu liên (HyperLink) trong siêu văn bản.

Tới năm 1994, một điểm yếu của INTERNET là không có khả năng lập trình cục bộ, vì các máy nối vào mạng không đồng bộ và không tương thích. Thiếu khả năng này, INTERNET chỉ được dùng trong việc phát hành và truyền thông tin chứ không dùng để xử lý thông tin được. Trong năm 1994, hãng máy tính SUN Corporation công bố một ngôn ngữ mới, gọi là JAVA (cafe), cho phép lập trình cục bộ trên INTERNET, các chương trình JAVA được gọi thẳng từ các siêu văn bản qua các siêu liên (Applet). Vào mùa thu năm 1995, ngôn ngữ JAVA chính thức ra đời, đánh dấu một bước tiến quan trọng trong việc sử dụng INTERNET. Trước hết, một chương trình JAVA, sẽ được chạy trên máy khách (Workstation) chứ không phải trên máy chủ (Server). Điều này cho phép sử dụng công suất của tất cả các máy khách vào việc xử lý số liệu. Hàng triệu máy tính (hoặc vi tính) có thể thực hiện cùng một lúc một chương trình ghi trên một siêu văn bản trong máy chủ. Việc lập trình trên INTERNET cho phép truy nhập từ một trang siêu văn bản vào các chương trình xử lý thông tin, đặc biệt là các chương trình điều hành và quản lý thông tin của một xí nghiệp. phương thức làm việc này, được gọi là INTRANET. Chỉ trong năm 1995-1996, hàng trăm nghìn dịch vụ phần mềm INTRANET được phát triển. Nhiều hãng máy tính và phần mềm như Microsoft, SUN, IBM, Oracle, Netscape,... đã phát triển và kinh doanh

hàng loạt phần mềm hệ thống và phần mềm cơ bản để phát triển các ứng dụng INTERNET / INTRANET.

II. Giới thiệu mạng máy tính

2.1. Định nghĩa mạng máy tính và mục đích của việc kết nối mạng

2.1.1. Nhu cầu của việc kết nối mạng máy tính

Việc nối máy tính thành mạng từ lâu đã trở thành một nhu cầu khách quan vì :

- Có rất nhiều công việc về bản chất là phân tán hoặc về thông tin, hoặc về xử lý hoặc cả hai đòi hỏi có sự kết hợp truyền thông với xử lý hoặc sử dụng phương tiện từ xa.

- Chia sẻ các tài nguyên trên mạng cho nhiều người sử dụng tại một thời điểm (ổ cứng, máy in, ổ CD ROM . . .)

- Nhu cầu liên lạc, trao đổi thông tin nhờ phương tiện máy tính.

- Các ứng dụng phần mềm đòi hòi tại một thời điểm cần có nhiều người sử dụng, truy cập vào cùng một cơ sở dữ liệu.

2.1.2. Định nghĩa mạng máy tính

Nói một cách ngắn gọn thì mạng máy tính là tập hợp các máy tính độc lập (autonomous) được kết nối với nhau thông qua các đường truyền vật lý và tuân theo các quy ước truyền thông nào đó.

Khái niệm máy tính độc lập được hiểu là các máy tính không có máy nào có khả năng khởi động hoặc đình chỉ một máy khác.

Các đường truyền vật lý được hiểu là các môi trường truyền tín hiệu vật lý (có thể là hữu tuyến hoặc vô tuyến).

Các quy ước truyền thông chính là cơ sở để các máy tính có thể "nói chuyện" được với nhau và là một yếu tố quan trọng hàng đầu khi nói về công nghệ mạng máy tính.

2.2. Đặc trưng kỹ thuật của mạng máy tính

Một mạng máy tính có các đặc trưng kỹ thuật cơ bản như sau:

2.2.1. Đường truyền

Là thành tố quan trọng của một mạng máy tính, là phương tiện dùng để truyền các tín hiệu điện tử giữa các máy tính. Các tín hiệu điệu tử đó chính là các thông tin, dữ liệu được biểu thị dưới dạng các xung nhị phân (ON_OFF), mọi tín hiệu truyền giữa các máy tính với nhau đều thuộc sóng điện từ, tuỳ theo tần số mà ta có thể dùng các đường truyền vật lý khác nhau

Đặc trưng cơ bản của đường truyền là giải thông nó biểu thị khả năng truyền tải tín hiệu của đường truyền.

Thông thuờng người ta hay phân loại đường truyền theo hai loại:

- Đường truyền hữu tuyến (các máy tính được nối với nhau bằng các dây cáp mạng).

- Đường truyền vô tuyến: các máy tính truyền tín hiệu với nhau thông qua các sóng vô tuyền với các thiết bị điều chế/giải điều chế ở các đầu mút.

2.2.2. Kỹ thuật chuyển mạch:

Là đặc trưng kỹ thuật chuyển tín hiệu giữa các nút trong mạng, các nút mạng có chức năng hướng thông tin tới đích nào đó trong mạng, hiện tại có các kỹ thuật chuyển mạch như sau:

- Kỹ thuật chuyển mạch kênh: Khi có hai thực thể cần truyền thông với nhau thì giữa chúng sẽ thiết lập một kênh cố định và duy trì kết nối đó cho tới khi hai bên ngắt liên lạc. Các dữ liệu chỉ truyền đi theo con đường cố định đó.

- Kỹ thuật chuyển mạch thông báo: thông báo là một đơn vị dữ liệu của người sử dụng có khuôn dạng được quy định trước. Mỗi thông báo có chứa các thông tin điều khiển trong đó chỉ rõ đích cần truyền tới của thông báo. Căn cứ vào thông tin điều khiển này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp trên con đường dẫn tới đích của thông báo

- Kỹ thuật chuyển mạch gói: ở đây mỗi thông báo được chia ra thành nhiều gói nhỏ hơn được gọi là các gói tin (packet) có khuôn dạng qui định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của cùng một thông báo có thể được gởi đi qua mạng tới đích theo nhiều con đường khác nhau.

2.2.3. Kiến trúc mạng

Kiến trúc mạng máy tính (network architecture) thể hiện cách nối các máy tính với nhau và tập hợp các quy tắc, quy ước mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để đảm bảo cho mạng hoạt động tốt.

Khi nói đến kiến trúc của mạng người ta muốn nói tới hai vấn đề là hình trạng mạng (Network topology) và giao thức mạng (Network protocol)

- Network Topology: Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng. Các hình trạng mạng cơ bản đó là: hình sao, hình bus, hình vòng

- Network Protocol: Tập hợp các quy ước truyền thông giữa các thực thể truyền thông mà ta gọi là giao thức (hay nghi thức) của mạng. Các giao thức thường gặp nhất là : TCP/IP, NETBIOS, IPX/SPX, . . .

2.2.4. Hệ điều hành mạng

Hệ điều hành mạng là một phần mềm hệ thống có các chức năng sau:

- Quản lý tài nguyên của hệ thống, các tài nguyên này gồm:

+ Tài nguyên thông tin (về phương diện lưu trữ) hay nói một cách đơn giản là quản lý tệp. Các công việc về lưu trữ tệp, tìm kiếm, xoá, copy, nhóm, đặt các thuộc tính đều thuộc nhóm công việc này

+ Tài nguyên thiết bị. Điều phối việc sử dụng CPU, các ngoại vi... để tối ưu hoá việc sử dụng

- Quản lý người dùng và các công việc trên hệ thống.

Hệ điều hành đảm bảo giao tiếp giữa người sử dụng, chương trình ứng dụng với thiết bị của hệ thống.

- Cung cấp các tiện ích cho việc khai thác hệ thống thuận lợi (ví dụ FORMAT đĩa, sao chép tệp và thư mục, in ấn chung ...)

Các hệ điều hành mạng thông dụng nhất hiện nay là: WindowsNT, Windows9X, Windows 2000, Unix, Novell.

2.3. Phân loại mạng máy tính

Có nhiều cách phân loại mạng khác nhau tuỳ thuộc vào yếu tố chính được chọn dùng để làm chỉ tiêu phân loại, thông thường người ta phân loại mạng theo các tiêu chí như sau

- Khoảng cách địa lý của mạng

- Kỹ thuật chuyển mạch mà mạng áp dụng

- Kiến trúc mạng

- Hệ điều hành mạng sử dụng ...

Tuy nhiên trong thực tế người ta thường chỉ phân loại theo hai tiêu chí đầu tiên

2.3.1. Phân loại mạng theo khoảng cách địa lý :

Nếu lấy khoảng cách địa lý làm yếu tố phân loại mạng thì ta có mạng cục bộ, mạng đô thị, mạng diện rộng, mạng toàn cầu.

Mạng cục bộ (LAN - Local Area Network) : là mạng được cài đặt trong phạm vi tương đối nhỏ hẹp như trong một toà nhà, một xí nghiệp...với khoảng cách lớn nhất giữa các máy tính trên mạng trong vòng vài km trở lại.

Mạng đô thị (MAN - Metropolitan Area Network) : là mạng được cài đặt trong phạm vi một đô thị, một trung tâm văn hoá xã hội, có bán kính tối đa khoảng 100 km trở lại.

Mạng diện rộng (WAN - Wide Area Network) : là mạng có diện tích bao phủ rộng lớn, phạm vi của mạng có thể vượt biên giới quốc gia thậm chí cả lục địa.

Mạng toàn cầu (GAN - Global Area Network) : là mạng có phạm vi trải rộng toàn cầu.

2.3.2. Phân loại theo kỹ thuật chuyển mạch:

Nếu lấy kỹ thuật chuyển mạch làm yếu tố chính để phân loại sẽ có: mạng chuyển mạch kênh, mạng chuyển mạch thông báo và mạng chuyển mạch gói.

Mạch chuyển mạch kênh (circuit switched network) : Khi có hai thực thể cần truyền thông với nhau thì giữa chúng sẽ thiết lập một kênh cố định và duy trì kết nối đó cho tới khi hai bên ngắt liên lạc. Các dữ liệu chỉ truyền đi theo con đường cố định đó. Nhược điểm của chuyển mạch kênh là tiêu tốn thời gian để thiết lập kênh truyền cố định và hiệu suất sử dụng mạng không cao.

Mạng chuyển mạch thông báo (message switched network) : Thông báo là một đơn vị dữ liệu của người sử dụng có khuôn dạng được quy định trước. Mỗi thông báo có chứa các thông tin điều khiển trong đó chỉ rõ đích cần truyền tới của thông báo. Căn cứ vào thông tin điều khiển này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp trên con đường dẫn tới đích của thông báo. Như vậy mỗi nút cần phải lưu giữ tạm thời để đọc thông tin điều khiển trên thông báo, nếu thấy thông báo không gửi cho mình thì tiếp tục chuyển tiếp thông báo đi. Tuỳ vào điều kiện của mạng mà thông báo có thể được chuyển đi theo nhiều con đường khác nhau.

Ưu điểm của phương pháp này là :

- Hiệu suất sử dụng đường truyền cao vì không bị chiếm dụng độc quyền mà được phân chia giữa nhiều thực thể truyền thông.

- Mỗi nút mạng có thể lưu trữ thông tin tạm thời sau đó mới chuyển thông báo đi, do đó có thể điều chỉnh để làm giảm tình trạng tắc nghẽn trên mạng.

- Có thể điều khiển việc truyền tin bằng cách sắp xếp độ ưu tiên cho các thông báo.

- Có thể tăng hiệu suất sử dụng giải thông của mạng bằng cách gắn địa chỉ quảng bá (broadcast addressing) để gửi thông báo đồng thời tới nhiều đích.

Nhược điểm của phương pháp này là:

- Không hạn chế được kích thước của thông báo dẫn đến phí tổn lưu giữ tạm thời cao và ảnh hưởng đến thời gian trả lời yêu cầu của các trạm .

Mạng chuyển mạch gói (packet switched network) : ở đây mỗi thông báo được chia ra thành nhiều gói nhỏ hơn được gọi là các gói tin (packet) có khuôn dạng qui định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của cùng một thông báo có thể được gởi đi qua mạng tới đích theo nhiều con đường khác nhau.

Phương pháp chuyển mạch thông báo và chuyển mạch gói là gần giống nhau. Điểm khác biệt là các gói tin được giới hạn kích thước tối đa sao cho các nút mạng (các nút chuyển mạch) có thể xử lý toàn bộ gói tin trong bộ nhớ mà không phải lưu giữ tạm thời trên đĩa. Bởi vậy nên mạng chuyển mạch gói truyền dữ liệu hiệu quả hơn so với mạng chuyển mạch thông báo.

Tích hợp hai kỹ thuật chuyển mạch kênh và chuyển mạch gói vào trong một mạng thống nhất được mạng tích hợp số ISDN (Integated Services Digital Network).

2.3.3. Phân loại theo kiến trúc mạng sử dụng

Kiến trúc của mạng bao gồm hai vấn đề: hình trạng mạng (Network topology) và giao thức mạng (Network protocol)

Hình trạng mạng: Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng

Giao thức mạng: Tập hợp các quy ước truyền thông giữa các thực thể truyền thông mà ta gọi là giao thức (hay nghi thức) của mạng

Khi phân loại theo topo mạng người ta thường có phân loại thành: mạng hình sao, tròn, tuyến tính

Phân loại theo giao thức mà mạng sử dụng người ta phân loại thành mạng : TCP/IP, mạng NETBIOS . ..

Tuy nhiên cách phân loại trên không phổ biến và chỉ áp dụng cho các mạng cục bộ.

2.3.4. Phân loại theo hệ điều hàng mạng

Nếu phân loại theo hệ điều hành mạng người ta chia ra theo mô hình mạng ngang hàng, mạng khách/chủ hoặc phân loại theo tên hệ điều hành mà mạng sử dụng: Windows NT, Unix, Novell . . .

2.4. Giới thiệu các mạng máy tính thông dụng nhất

2.4.1. Mạng cục bộ

Một mạng cục bộ là sự kết nối một nhóm máy tính và các thiết bị kết nối mạng được lắp đặt trên một phạm vị địa lý giới hạn, thường trong một toà nhà hoặc một khu công sở nào đó.

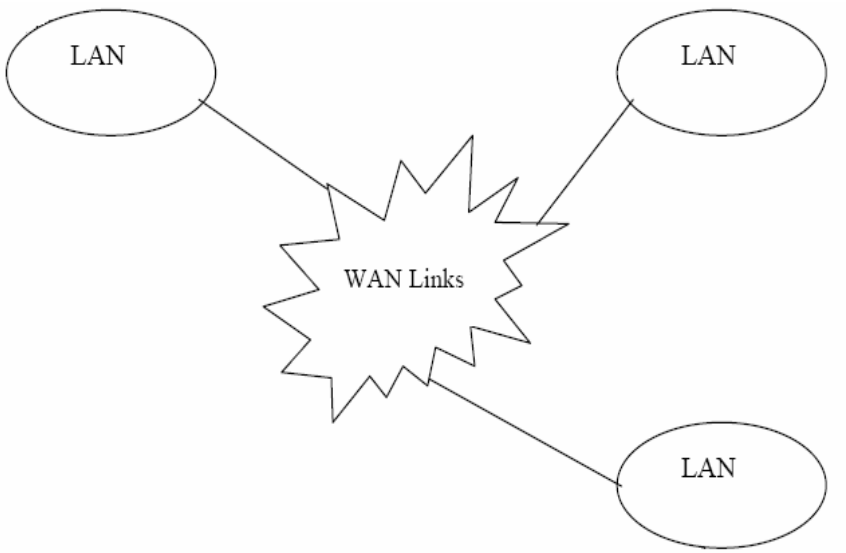
Mạng cục bộ có các đặc tính sau:

- Tốc độ truyền dữ liệu cao

- Phạm vi địa lý giới hạn

-Sở hữu của một cơ quan/tổ chức

2.4.2. Mạng diện rộng với kết nối LAN TO LAN



Mạng diện rộng bao giờ cũng là sự kết nối của các mạng LAN, mạng diện rộng có thể trải trên phạm vi một vùng, quốc gia hoặc cả một lục địa thậm chí trên phạm vi toàn cầu.

- Tốc độ truyền dữ liệu không cao

- Phạm vi địa lý không giới hạn

- Thường triển khai dựa vào các công ty truyền thông, bưu điện và dùng các hệ thống truyền thông này để tạo dựng đường truyền

- Một mạng WAN có thể là sở hữu của một tập đoàn/tổ chức hoặc là mạng kết nối của nhiều tập đoàn/tỗ chức 16

2.4.3. Liên mạng INTERNET

Với sự phát triển nhanh chóng của công nghệ là sự ra đời của liên mạng INTERNET,

- Là một mạng toàn cầu

- Là sự kết hợp của vô số các hệ thống truyền thông, máy chủ cung cấp thông tin và dịch vụ, các máy trạm khai thác thông tin

- Dựa trên nhiều nền tảng truyền thông khác nhau, nhưng đều trên nền giao thức TCP/IP

- Là sở hữu chung của toàn nhân loại

- Càng ngày càng phát triển mãnh liệt

2.4.4. Mạng INTRANET

Thực sự là một mạng INTERNET thu nhỏ vào trong một cơ quan/công ty/tổ chức hay một bộ/nghành . . ., giới hạn phạm vi người sử dụng, có sử dụng các công nghệ kiểm soát truy cập và bảo mật thông tin .

Được phát triển từ các mạng LAN, WAN dùng công nghệ INTERNET

III. Mạng cục bộ, kiến trúc mạng cục bộ

3.1. Mạng cục bộ

Tên gọi "mạng cục bộ" được xem xét từ quy mô của mạng. Tuy nhiên, đó không phải là đặc tính duy nhất của mạng cục bộ nhưng trên thực tế, quy mô của mạng quyết định nhiều đặc tính và công nghệ của mạng. Sau đây là một số đặc điểm của mạng cục bộ:

Đặc điểm của mạng cục bộ

- Mạng cục bộ có quy mô nhỏ, thường là bán kính dưới vài km. Đặc điểm này cho phép không cần dùng các thiết bị dẫn đường với các mối liên hệ phức tạp

- Mạng cục bộ thường là sở hữu của một tổ chức. Điều này dường như có vẻ ít quan trọng nhưng trên thực tế đó là điều khá quan trọng để việc quản lý mạng có hiệu quả.

- Mạng cục bộ có tốc độ cao và ít lỗi. Trên mạng rộng tốc độ nói chung chỉ đạt vài Kbit/s. Còn tốc độ thông thường trên mạng cục bộ là 10, 100 Kb/s và tới nay với Gigabit Ethernet, tốc độ trên mạng cục bộ có thể đạt 1Gb/s. Xác suất lỗi rất thấp.

3.2. Kiến trúc mạng cục bộ

*** Định nghĩa Topo mạng:**

Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng

Có hai kiểu nối mạng chủ yếu đó là :

- Nối kiểu điểm - điểm (point - to - point).

- Nối kiểu điểm - nhiều điểm (point - to - multipoint hay broadcast).

Theo kiểu điểm - điểm, các đường truyền nối từng cặp nút với nhau và mỗi nút đều có trách nhiệm lưu giữ tạm thời sau đó chuyển tiếp dữ liệu đi cho tới đích. Do cách làm việc như vậy nên mạng kiểu này còn được gọi là mạng "lưu và chuyển tiếp" (store and forward).

Theo kiểu điểm - nhiều điểm, tất cả các nút phân chia nhau một đường truyền vật lý chung. Dữ liệu gửi đi từ một nút nào đó sẽ được tiếp nhận bởi tất cả các nút còn lại trên mạng, bởi vậy cần chỉ ra địa chỉ đích của dữ liệu để căn cứ vào đó các nút kiểm tra xem dữ liệu đó có phải gửi cho mình không.

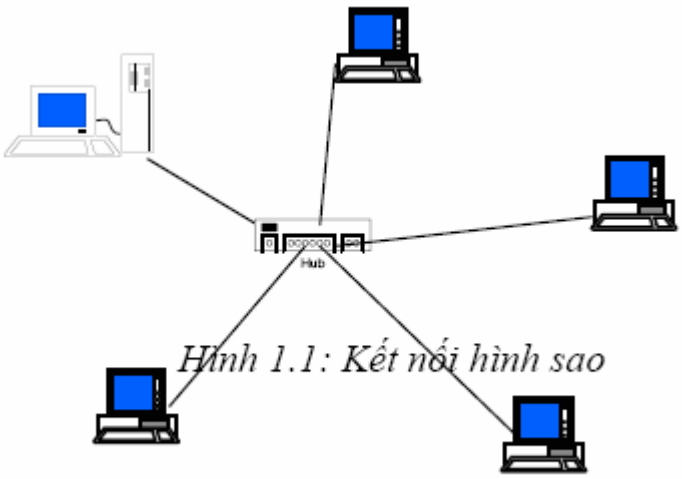
*** Phân biệt kiểu tô pô của mạng cục bộ và kiểu tô pô của mạng rộng.**

Tô pô của mạng rộng thông thường là nói đến sự liên kết giữa các mạng cục bộ thông qua các bộ dẫn đường (router). Đối với mạng rộng topo của mạng là hình trạng hình học của các bộ dẫn đường và các kênh viễn thông còn khi nói tới tô pô của mạng cục bộ người ta nói đến sự liên kết của chính các máy tính.

a) Mạng hình sao

Mạng hình sao có tất cả các trạm được kết nối với một thiết bị trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển đến trạm đích. Tuỳ theo yêu cầu truyền thông trên mạng mà thiết bị trung tâm có thể là bộ chuyển mạch (switch), bộ chọn đường (router) hoặc là bộ phân kênh (hub). Vai trò của thiết bị trung tâm này là thực hiện việc thiết lập các liên kết điểm-điểm (point-to-point) giữa các trạm.

Ưu điểm: Thiết lập mạng đơn giản, dễ dàng cấu hình lại mạng (thêm, bớt các trạm), dễ dàng kiểm soát và khắc phục sự cố, tận dụng được tối đa tốc độ



Hình 1.1: Kết nối hình sao

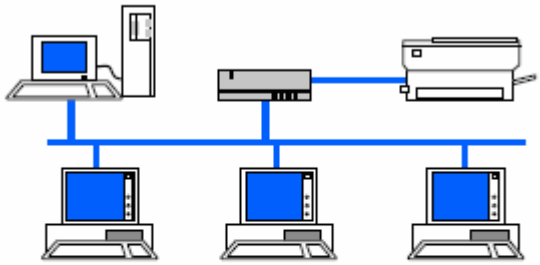
truyền của đường truyền vật lý.

Nhược điểm: Độ dài đường truyền nối một trạm với thiết bị trung tâm bị hạn chế (trong vòng 100m, với công nghệ hiện nay). Hub

b) Mạng trục tuyến tính (Bus):

Trong mạng trục tất cả các trạm phân chia một đường truyền chung (bus). Đường truyền chính được giới hạn hai đầu bằng hai đầu nối đặc biệt gọi là terminator. Mỗi trạm được nối với trục chính qua một đầu nối chữ T (T-connector) hoặc một thiết bị thu phát (transceiver).

Khi một trạm truyền dữ liệu tín hiệu được quảng bá trên cả hai chiều của bus, tức là mọi trạm còn lại đều có thể thu được tín hiệu đó trực tiếp. Đối với các bus một chiều thì tín hiệu chỉ đi về một phía, lúc đó các terminator phải được thiết kế sao cho các tín hiệu đó phải được dội lại trên bus để cho các trạm trên mạng đều có thể thu nhận được tín hiệu đó. Như vậy với topo mạng trục dữ liệu được truyền theo các liên kết điểm-đa điểm (point-to-multipoint) hay quảng bá (broadcast).



Hình 1.2. Kết nối kiểu bus

Ưu điểm : Dễ thiết kế, chi phí thấp

Nhược điểm: Tính ổn định kém, chỉ một nút mạng hỏng là toàn bộ mạng bị ngừng hoạt động

c) Mạng hình vòng

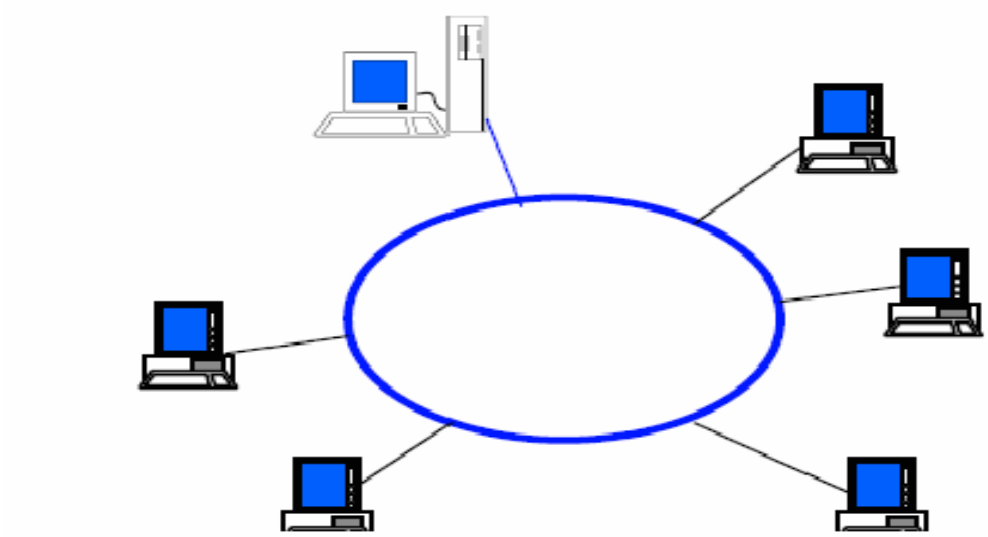
Trên mạng hình vòng tín hiệu được truyền đi trên vòng theo một chiều duy nhất. Mỗi trạm của mạng được nối với vòng qua một bộ chuyển tiếp (repeater) có nhiệm vụ nhận tín hiệu rồi chuyển tiếp đến trạm kế tiếp trên vòng. Như vậy tín hiệu được lưu chuyển trên vòng theo một chuỗi liên tiếp các liên kết điểm-điểm giữa các repeater do đó cần có giao thức điều khiển việc cấp phát quyền được truyền dữ liệu trên vòng mạng cho trạm có nhu cầu.

Để tăng độ tin cậy của mạng ta có thể lắp đặt thêm các vòng dự phòng, nếu vòng chính có sự cố thì vòng phụ sẽ được sử dụng.

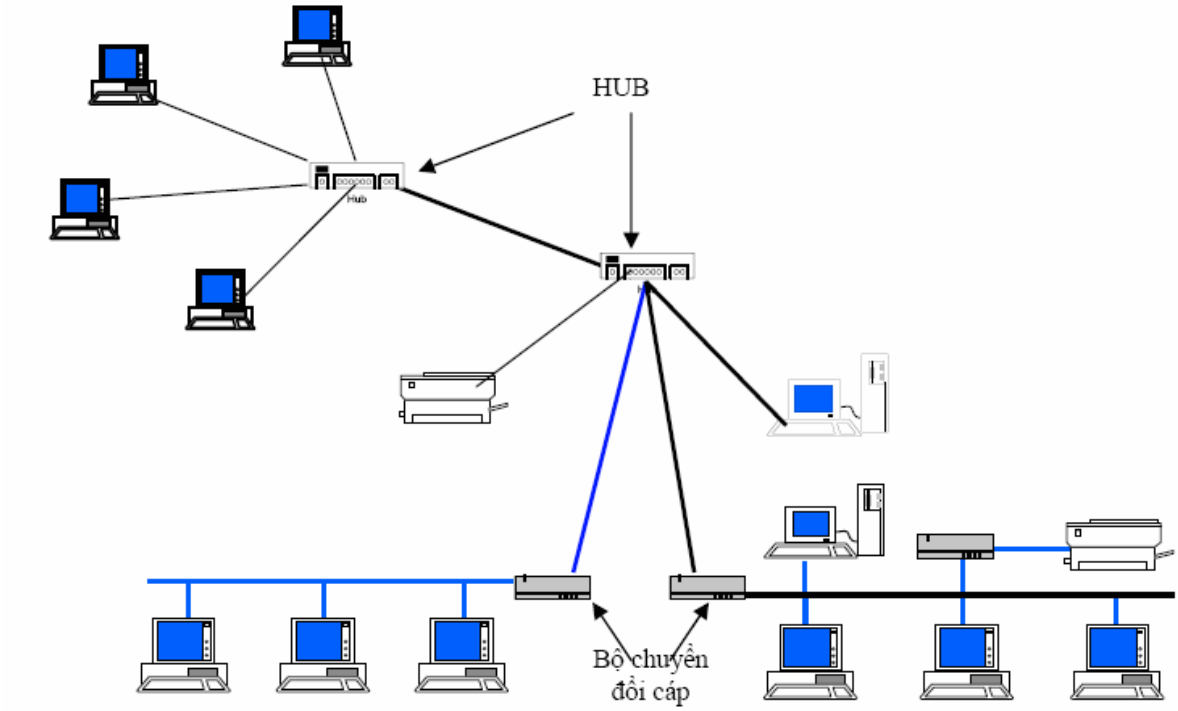
Mạng hình vòng có ưu nhược điểm tương tự mạng hình sao, tuy nhiên mạng hình vòng đòi hỏi giao thức truy nhập mạng phức tạp hơn mạng hình sao.

d) Kết nối hỗn hợp

Là sự phối hợp các kiểu kết nối khác nhau, ví du hình cây là cấu trúc phân tầng của kiểu hình sao hay các HUB có thể được nối với nhau theo kiểu bus còn từ các HUB nối với các máy theo hình sao.



Hình 1.3 Kết nối kiểu vòng



Hình 1.4. Một kết nối hỗn hợp

3.3. Các phương pháp truy cập đường truyền vật lý

Trong mạng cục bộ, tất cả các trạm kết nối trực tiếp vào đường truyền chung. Vì vậy tín hiệu từ một trạm đưa lên đường truyền sẽ được các trạm khác "nghe thấy". Một vấn đề khác là, nếu nhiều trạm cùng gửi tín hiệu lên đường truyền đồng thời thì tín hiệu sẽ chồng lên nhau và bị hỏng. Vì vậy cần phải có

một phương pháp tổ chức chia sẻ đường truyền để việc truyền thông đực đúng đắn.

Có hai phương pháp chia sẻ đường truyền chung thường được dùng trong các mạng cục bộ:

- Truy nhập đường truyền một cách ngẫu nhiên, theo yêu cầu. Đương nhiên phải có tính đến việc sử dụng luân phiên và nếu trong trường hợp do có nhiều trạm cùng truyền tin dẫn đến tín hiệu bị trùm lên nhau thì phải truyền lại.

- Có cơ chế trọng tài để cấp quyền truy nhập đường truyền sao cho không xảy ra xung đột

3.3.1 Phương pháp đa truy nhập sử dụng sóng mang có phát hiện xung đột CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

Giao thức CSMA (Carrier Sense Multiple Access) - đa truy nhập có cảm nhận sóng mang được sử dụng rất phổ biến trong các mạng cục bộ. Giao thức này sử dụng phương pháp thời gian chia ngăn theo đó thời gian được chia thành các khoảng thời gian đều đặn và các trạm chỉ phát lên đường truyền tại thời điểm đầu ngăn.

Mỗi trạm có thiết bị nghe tín hiệu trên đường truyền (tức là cảm nhận sóng mang). Trước khi truyền cần phải biết đường truyền có rỗi không. Nếu rỗi thì mới được truyền. Phương pháp này gọi là LBT (Listening before talking). Khi phát hiện xung đột, các trạm sẽ phải phát lại. Có một số chiến lược phát lại như sau:

- Giao thức CSMA không kiên trì.Trạm nghe đường, nếu kênh rỗi thì truyền, nếu không thì ngừng nghe một khoảng thời gian ngẫu nhiên rồi mới thực hiện lại thủ tục. Cách này có hiệu suất dùng kênh cao hơn. (1)

- Giao thức CSMA 1-kiên trì. Khi trạm phát hiện kênh rỗi trạm truyền ngay. Nhưng nếu có xung đột, trạm đợi khoảng thời gian ngẫu nhiên rồi truyền lại. Do vậy xác suất truyền khi kênh rỗi là 1. Chính vì thế mà giao thức có tên là CSMA 1-kiên trì. (2)

- Giao thức CSMA p-kiên trì. Khi đã sẵn sàng truyền, trạm cảm nhận đường, nếu đường rỗi thì thực hiện việc truyền với xác suất là $p < 1$ (tức là ngay cả khi đường rỗi cũng không hẳn đã truyền mà đợi khoảng thời gian tiếp theo lại tiếp tục thực hiện việc truyền với xác suất còn lại $q=1-p$. (3)

• Ta thấy giải thuật (1) có hiệu quả trong việc tránh xung đột vì hai trạm cần truyền thấy đường truyền bận sẽ cùng rút lui chờ trong những khoảng thời gian ngẫu nhiên khác nhau sẽ quay lại tiếp tục nghe đường truyền. Nhược điểm của nó là có thể có thời gian không sử dụng đường truyền sau mỗi cuộc gọi.

• Giải thuật (2) cố gắng làm giảm thời gian "chết" bằng cách cho phép một trạm có thể được truyền dữ liệu ngay sau khi một cuộc truyền kết thúc. Tuy

nhiên nếu lúc đó lại có nhiều trạm đang đợi để truyền dữ liệu thì khả năng xẩy ra xung đột sẽ rất lớn.

• Giải thuật (3) với giá trị p được họn hợp lý có thể tối thiểu hoá được cả khả năng xung đột lẫn thời gian "chết" của đường truyền.

• Xẩy ra xung đột thường là do độ trễ truyền dẫn, mấu chốt của vấn đề là: các trạm chỉ "nghe" trước khi truyền dữ liệu mà không "nghe" trong khi truyền, cho nên thực tế có xung đột thế nhưng các trạm không biết do đó vẫn truyền dữ liệu.

• Để có thể phát hiện xung đột, CSMA/CD đã bổ xung thêm các quy tắc sau đây:

- Khi một trạm truyền dữ liệu, nó vẫn tiếp tục "nghe" đường truyền . Nếu phát hiện xung đột thì nó ngừng ngay việc truyền, nhờ đó mà tiết kiệm được thời gian và giải thông, nhưng nó vẫn tiếp tục gửi tín hiệu thêm một thời gian nữa để đảm bảo rằng tất cả các trạm trên mạng đều "nghe" được sự kiện này.(như vậy phải tiếp tục nghe đường truyền trong khi truyền để phát hiện đụng độ (Listening While Talking))

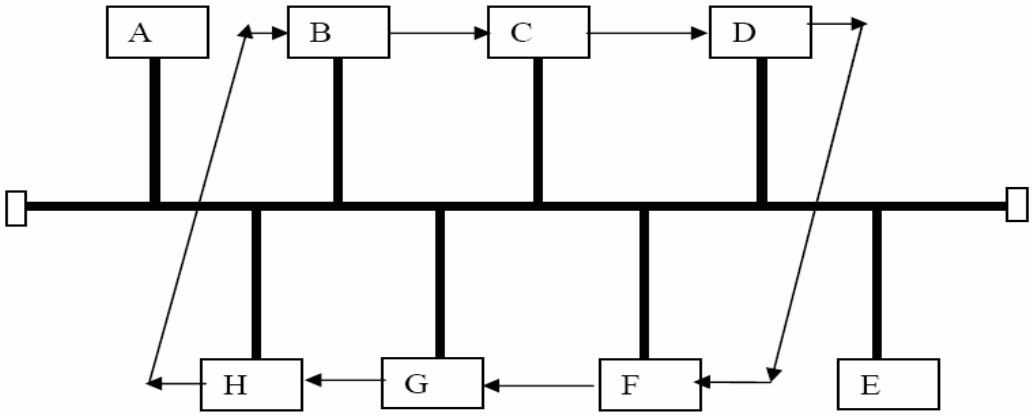
- Sau đó trạm sẽ chờ trong một khoảng thời gian ngẫu nhiên nào đó rồi thử truyền lại theo quy tắc CSMA.

Giao thức này gọi là **CSMA có phát hiện xung đột** (Carrier Sense Multiple Access with Collision Detection viết tắt là CSMA/CD), dùng rộng rãi trong LAN và MAN.

3.3.2. Phương pháp Token Bus

Nguyên lý chung của phương pháp này là để cấp phát quyền truy nhập đường truyền cho các trạm đang có nhu cầu truyền dữ liệu, một thẻ bài được lưu chuyển trên một vòng logic được thiết lập bởi các trạm đó. Khi một trạm nhận được thẻ bài thì sẽ được phép sử dụng đường truyền trong một thời gian nhất định. Trong khoảng thời gian đó nó có thể truyền một hay nhiều đơn vị dữ liệu. Khi đã truyền xong dữ liệu hoặc thời gian đã hết thì trạm đó phải chuyển thẻ bài cho trạm tiếp theo. Như vậy, công việc đầu tiên là thiết lập vòng logic (hay còn gọi là vòng ảo) bao gồm các trạm đang có nhu cầu truyền dữ liệu được xác định vị trí theo một chuỗi thứ tự mà trạm cuối cùng của chuỗi sẽ tiếp liền sau bởi trạm đầu tiên. Mỗi trạm sẽ biết địa chỉ của trạm liền trước và kề sau nó. Thứ tự của các trạm trên vòng logic có thể độc lập với thứ tự vật lý. Các trạm không hoặc chưa có nhu cầu truyền dữ liệu không được vào trong vòng logic.

Trong ví dụ trên, các trạm A, E nằm ngoài vòng logic do đó chỉ có thể tiếp nhận được dữ liệu dành cho chúng.



Hình 1.5. Ví dụ về vòng logic

Việc thiết lập vòng logic không khó nhưng việc duy trì nó theo trạng thái thực tế của mạng mới là khó. Cụ thể phải thực hiện các chức năng sau:

a) Bổ xung một trạm vào vòng logic : các trạm nằm ngoài vòng logic cần được xem xét một cách định kỳ để nếu có nhu cầu truyền dữ liệu thì được bổ xung vào vòng logic.

b) Loại bỏ một vòng khỏi vòng logic : khi một trạm không có nhu cầu truyền dữ liệu thì cần loại bỏ nó ra khỏi vòng logic để tối ưu hoá việc truyền dữ liệu bằng thẻ bài

c) Quản lý lỗi : một số lỗi có thể xẩy ra như trùng hợp địa chỉ, hoặc đứt vòng logic.

d) Khởi taọ vòng logic : khi khởi tạo mạng hoặc khi đứt vòng logic cần phải khởi tạo lại vòng logic.

3.3.3. Phương pháp Token Ring

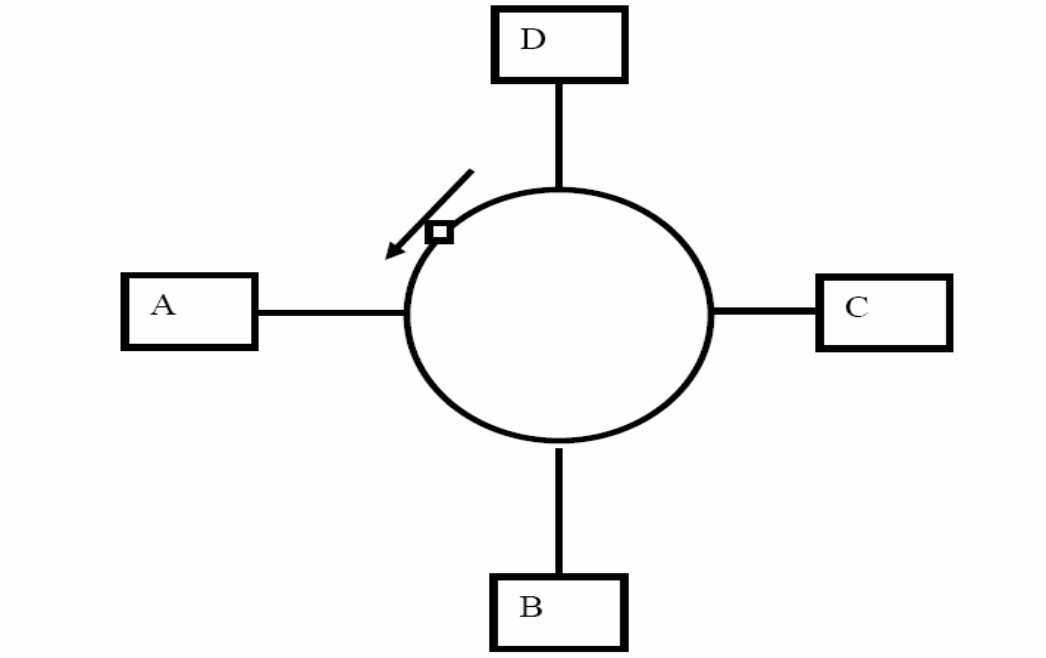
Phương pháp này cũng dựa trên nguyên tắc dùng thẻ bài để cấp phát quyền truy nhập đường truyền. Nhưng ở đây thẻ bài lưu chuyển theo theo vòng vật lý chứ không theo vòng logic như dối với phương pháp token bus.

Thẻ bài là một đơn vị truyền dữ liệu đặc biệt trong đó có một bit biểu diễn trạng thái của thẻ (bận hay rỗi). Một trạm muốn truyền dữ liệu phải chờ cho tới khi nhận được thẻ bài "rỗi". Khi đó trạm sẽ đổi bit trạng thái thành "bận" và truyền một đơn vị dữ liệu đi cùng với thẻ bài đi theo chiều của vòng. Lúc này không còn thẻ bài "rỗi " nữa do đó các trạm muốn truyền dữ liệu phải đợi. Dữ liệu tới trạm đích được sao chép lại, sau đó cùng với thẻ bài trở về trạm nguồn. Trạm nguồn sẽ xoá bỏ dữ liệu đổi bit trạng thái thành "rỗi" và cho lưu chuyển thẻ trên vòng để các trạm khác có nhu cầu truyền dữ liệu được phép truyền

Sự quay trở lại trạm nguồn của dữ liệu và thẻ bài nhằm tạo khả năng báo nhận tự nhiên: trạm đích có thể gửi vào đơn vị dữ liệu (phần header) các thông tin về kết quả tiếp nhận dữ liệu của mình. Chẳng hạn các thông tin đó có thể là: trạm đích không tồn tại hoặc không hoạt động, trạm đích tồn tại nhưng dữ liệu không được sao chép, dữ liệu đã được tiếp nhận, có lỗi...

Trong phương pháp này cần giải quyết hai vấn đề có thể dẫn đến phá vỡ hệ thống đó là mất thẻ bài và thẻ bài "bận" lưu chuyển không dừng trên vòng .Có nhiều phương pháp giải quyết các vấn đề trên, dưới đây là một phương pháp được khuyến nghị:

Đối với vấn đề mất thẻ bài có thể quy định trước một trạm điều khiển chủ động. Trạm này sẽ theo dõi, phát hiện tình trạng mất thẻ bài bằng cách dùng cơ



Hình 1.6. Thẻ bài trong mạng Ring

chế ngưỡng thời gian (time - out) và phục hồi bằng cách phát đi một thẻ bài "rỗi" mới.

Đối với vấn đề thẻ bài bận lưu chuyển không dừng, trạm điều khiển sử dụng một bit trên thẻ bài để đánh dấu khi gặp một thẻ bài "bận" đi qua nó. Nếu nó gặp lại thẻ bài bận với bit đã đánh dấu đó có nghĩa là trạm nguồn đã không nhận lại được đơn vị dữ liệu của mình do đó thẻ bài "bận" cứ quay vòng mãi. Lúc đó trạm điều khiển sẽ chủ động đổi bit trạng thái "bận" thành "rỗi" và cho thẻ bài chuyển tiếp trên vòng. Trong phương pháp này các trạm còn lại trên mạng sẽ đóng vai trò bị động, chúng theo dõi phát hiện tình trạng sự cố trên trạm chủ động và thay thế trạm chủ động nếu cần.

IV. Chuẩn hoá mạng máy tính

4.1. Vấn đề chuẩn hoá mạng và các tổ chức chuẩn hoá mạng

Khi thiết kế, các nhà thiết kế tự do lựa chọn kiến trúc mạng cho riêng mình. Từ đó dẫn tới tình trạng không tương thích giữa các mạng máy tính với nhau. Nhu cầu trao đổi thông tin càng lớn thúc đẩy việc xây dựng khung chuẩn về kiến trúc mạng để làm căn cứ cho các nhà thiết kế và chế tạo thiết bị mạng.

Chính vì lý do đó, tổ chức tiêu chuẩn hoá quốc tế ISO (Internatinal Organnization for Standarzation) đã xây dựng mô hình tham chiếu cho việc kết nối các hệ thống mở OSI (reference model for Open Systems Interconnection). Mô hình này là cơ sở cho việc kết nối các hệ thống mở phục vụ cho các ứng dụng phân tán.

Có hai loại chuẩn cho mạng đó là :

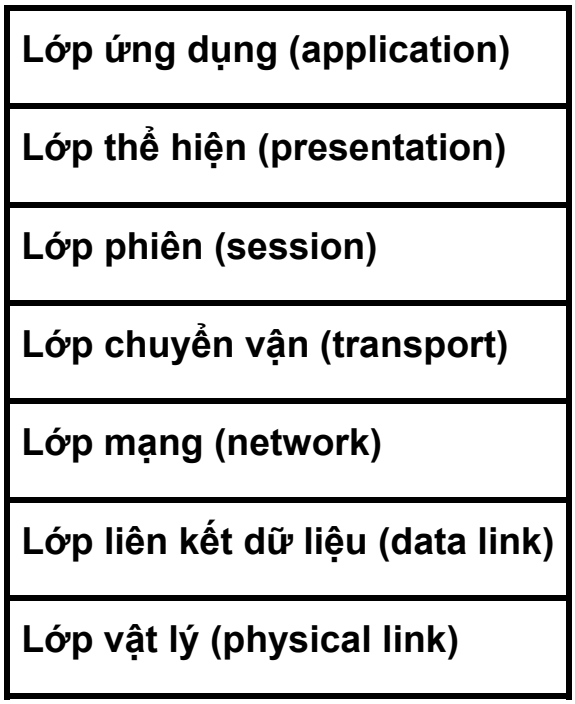
- Các chuẩn chính thức (de jure) do các tổ chức chuẩn quốc gia và quốc tế ban hành.

- Các chuẩn tực tiễn (de facto) do các hãng sản xuất, các tổ chức người sử dụng xây dựng và được dùng rộng rãi trong thực tế

4.2. Mô hình tham chiếu OSI 7 lớp

Khi thiết kế, các nhà thiết kế tự do lựa chọn kiến trúc mạng cho riêng mình. Từ đó dẫn tới tình trạng không tương thích giữa các mạng máy tính với nhau. Vấn đề không tương thích đó làm trở ngại cho sự tương tác giữa những người sử dụng mạng khác nhau. Nhu cầu trao đổi thông tin càng lớn thúc đẩy việc xây dựng khung chuẩn về kiến trúc mạng để làm căn cứ cho các nhà thiết kế và chế tạo thiết bị mạng .

Chính vì lý do đó, tổ chức tiêu chuẩn hoá quốc tế ISO (Internatinal Organnization for Standarzation) đã xây dựng mô hình tham chiếu cho việc kết nối các hệ thống mở OSI (reference model for Open Systems Interconnection). Mô hình này là cơ sở cho việc kết nối các hệ thống mở phục vụ cho các ứng dụng phân tán. Mô hình OSI được biểu diễn theo hình dưới đây:



Hình 1.7. Mô hình OSI 7 lớp

a) Lớp vật lý

Lớp này bảo đảm các công việc sau:

- Lập, cắt cuộc nối.
- Truyền tin dạng bit qua kênh vật lý.
- Có thể có nhiều kênh.

b) Lớp liên kết dữ liệu

Lớp này đảm bảo việc biến đổi các tin dạng bit nhận được từ lớp dưới (vật lý) sang khung số liệu, thông báo cho hệ phát, kết quả thu được sao cho các thông tin truyền lên cho mức 3 không có lỗi. Các thông tin truyền ở mức 1 có thể làm hỏng các thông tin khung số liệu (frame error). Phần mềm mức hai sẽ thông báo cho mức một truyền lại các thông tin bị mất / lỗi. Đồng bộ các hệ có tốc độ xử lý tính khác nhau, một trong những phương pháp hay sử dụng là dùng bộ đệm trung gian để lưu giữ số liệu nhận được. Độ lớn của bộ đệm này phụ thuộc vào tương quan xử lý của các hệ thu và phát. Trong trường hợp đường truyền song công toàn phần, lớp datalink phải đảm bảo việc quản lý các thông tin số liệu và các thông tin trạng thái.

c) Lớp mạng

Nhiệm vụ của lớp mạng là đảm bảo chuyển chính xác số liệu giữa các thiết bị cuối trong mạng. Để làm được việc đó, phải có chiến lược đánh địa chỉ thống nhất trong toàn mạng. Mỗi thiết bị cuối và thiết bị mạng có một địa chỉ

mạng xác định. Số liệu cần trao đổi giữa các thiết bị cuối được tổ chức thành các gói (packet) có độ dài thay đổi và được gán đầy đủ địa chỉ nguồn (source address) và địa chỉ đích (destination address).

Lớp mạng đảm bảo việc tìm đường tối ưu cho các gói dữ liệu bằng các giao thức chọn đường dựa trên các thiết bị chọn đường (router). Ngoài ra, lớp mạng có chức năng điều khiển lưu lượng số liệu trong mạng để tránh xảy ra tắc ngẽn bằng cách chọn các chiến lược tìm đường khác nhau để quyết định việc chuyển tiếp các gói số liệu.

d) Lớp chuyển vận

Lớp này thực hiện các chức năng nhận thông tin từ lớp phiên (**session**) chia thành các gói nhỏ hơn và truyền xuống lớp dưới, hoặc nhận thông tin từ lớp dưới chuyển lên phục hồi theo cách chia của hệ phát (Fragmentation and Reassembly). Nhiệm vụ quan trọng nhất của lớp vận chuyển là đảm bảo chuyển số liệu chính xác giữa hai thực thể thuộc lớp phiên (end-to-end control). Để làm được việc đó, ngoài chức năng kiểm tra số tuần tự phát, thu, kiểm tra và phát hiện, xử lý lỗi.Lớp vận chuyển còn có chức năng điều khiển lưu lượng số liệu để đồng bộ giữa thể thu và phát, tránh tắc nghẽn số liệu khi chuyển qua lớp mạng. Ngoài ra, nhiều thực thể lớp phiên có thể trao đổi số liệu trên cùng một kết nối lớp mạng (multiplexing).

e) Lớp phiên

Liên kết giữa hai thực thể có nhu cầu trao đổi số liệu, ví dụ người dùng và một máy tính ở xa, được gọi là một phiên làm việc. Nhiệm vụ của lớp phiên là quản lý việc trao đổi số liệu, ví dụ: thiết lập giao diện giữa người dùng và máy, xác định thông số điều khiển trao đổi số liệu (tốc độ truyền, số bit trong một byte, có kiểm tra lỗi parity hay không, v.v.), xác định loại giao thức mô phỏng thiết bị cuối (terminal emulation), v.v. Chức năng quan trọng nhất của lớp phiên là đảm bảo đồng bộ số liệu bằng cách thực hiện các điểm kiểm tra. Tại các điểm kiểm tra này, toàn bộ trạng thái và số liệu của phiên làm việc được lưu trữ trong bộ nhớ đệm. Khi có sự cố, có thể khởi tạo lại phiên làm việc từ điểm kiểm tra cuối cùng (không phải khởi tạo lại từ đầu).

f) Lớp thể hiện

Nhiệm vụ của lớp thể hiện là thích ứng các cấu trúc dữ liệu khác nhau của người dùng với cấu trúc dữ liệu thống nhất sử dụng trong mạng. Số liệu của người dùng có thể được nén và mã hoá ở lớp thể hiện, trước khi chuyển xuống lớp phiên. Ngoài ra, lớp thể hiện còn chứa các thư viện các yêu cầu của người dùng, thư viện tiện ích, ví dụ thay đổi dạng thể hiện của các tệp, nén tệp...

g) Lớp ứng dụng

Lớp ứng dụng cung cấp các phương tiện để người sử dụng có thể truy nhập được vào môi trường OSI, đồng thời cung cấp các dịch vụ thông tin phân

tán. Lớp mạng cho phép người dùng khai thác các tài nguyên trong mạng tương tự như tài nguyên tại chỗ.

4.3. Các chuẩn kết nối thông dụng nhất IEEE 802.X và ISO 8802.X

Bên cạnh việc chuẩn hoá cho mạng nối chung dẫn đến kết quả cơ bản nhất là mô hình tham chiếu OSI như đã giới thiệu. Việc chuẩn hoá mạng cục bộ nói riêng đã được thực hiện từ nhiều năm nay để đáp ứng sự phát triển của mạng cục bộ.

Cũng như đối với mạng nói chung, có hai loại chuẩn cho mạng cục bộ, đó là :

- Các chuẩn chính thức (de jure) do các tổ chức chuẩn quốc gia và quốc tế ban hành.
- Các chuẩn tực tiễn (de facto) do các hãng soản xuất, các tổ chức người sử dụng xây dựng và được dùng rộng rãi trong thực tế
- Các chuẩn IEEE 802.x và ISO 8802.x

IEEE là tổ chức đi tiên phong trong lĩnh vực chuẩn hoá mạng cục bộ với đề án IEEE 802 với kết quả là một loạt các chuẩn thuộc họ IEEE 802.x ra đời. Cuối những năm 80, tổ chức ISO đã tiếp nhận họ chuẩn này và ban hành thành chuẩn quốc tế dưới mã hiệu tương ứng là ISO 8802.x.

IEEE 802.: là chuẩn đặc tả kiến trúc mạng, kết nối giữa các mạng và việc quản trị mạng đối với mạng cục bộ.

IEEE 802.2: là chuẩn đặc tả tầng dịch vụ giao thức của mạng cục bộ.

IEEE 802.3: là chuẩn đặc tả một mạng cục bộ dựa trên mạng Ethernet nổi tiếng của Digital, Intel và Xerox hợp tác xây dựng từ năm 1980.

Tầng vật lý của IEEE 802.3 có thể dùng các phương án sau để xây dựng:

- 10BASE5 : tốc độ 10Mb/s, dùng cáp xoắn đôi không bọc kim UTP (Unshield Twisted Pair), với phạm vi tín hiệu lên tới 500m, topo mạng hình sao.

- 10BASE2 : tốc độ 10Mb/s, dùng cáp đồng trục thin-cable với trở kháng 50 Ohm, phạm vi tín hiệu 200m,topo mạng dạng bus.

- 10BASE5 : tốc độ 10Mb/s, dùng cáp đồng trục thick-cable (đường kính 10mm) với trở kháng 50 Ohm, phạm vi tín hiệu 500m, topo mạng dạng bus.

- 10BASE-F: dùng cáp quang, tốc độ 10Mb/s phạm vi cáp 2000m.

IEEE 802.4: là chuẩn đặc tả mạng cục bộ với topo mạng dạng bus dùng thẻ bài để điều việc truy nhập đường truyền.

IEEE 802.5: là chuẩn đặc tả mạng cục bộ với topo mạng dạng vòng (ring) dùng thẻ bài để điều việc truy nhập đường truyền.

IEEE 802.6: là chuẩn đặc tả mạng tốc độ cao kết nối với nhiều mạng cục bộ thuộc các khu vực khác nhau của một đô thị (còn được gọi là mạng MAN - Metropolitan Area Network)

IEEE 802.9: là chuẩn đặc tả mạng tích hợp dữ liệu và tiếng nói bao gồm 1 kênh dị bộ 10 Mb/s cùng với 96 kênh 64Kb/s. Chuẩn này được thiết kế cho môi trường có lượng lưu thông lớn và cấp bách.

IEEE 802.10: là chuẩn đặc tả về an toàn thông tin trong các mạng cục bộ có khả năng liên tác .

IEEE 802.11: là chuẩn đặc tả mạng cục bộ không dây (Wireless LAN) hiện đang được tiếp tục phát triển.

IEEE 802.12: là chuẩn đặc tả mạng cục bộ dựa trên công nghệ được đề xuất bởi AT&T, IBM và HP gọi là 100 VG - AnyLAN. Mạng này có topo mạng hình sao và một phương pháp truy nhập đường truyền có điều khiển tranh chấp. Khi có nhu cầu truyền dữ liệu, một trạm sẽ gửi yêu cầu đến hub và trạm chỉ có truyền dữ liệu khi hub cho phép.

Chương 2. Các thiết bị mạng thông dụng và các chuẩn kết nối vật lý

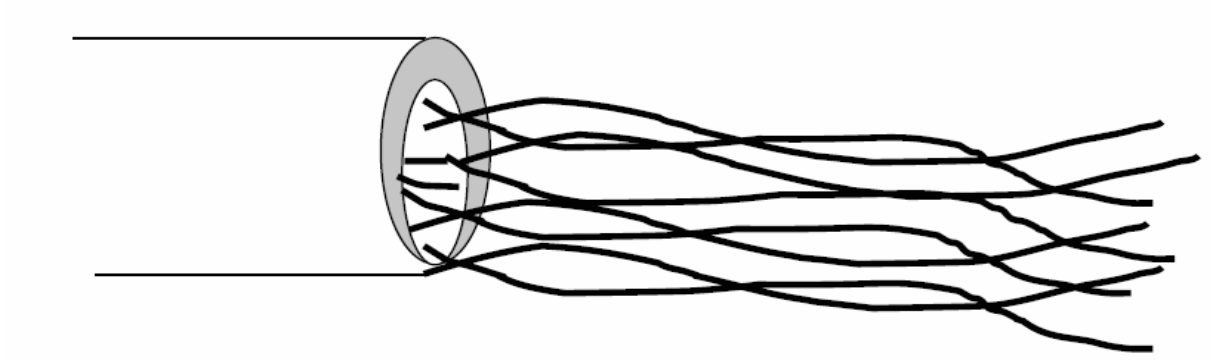
I. Các thiết bị mạng thông dụng

1.1. Các loại cáp truyền

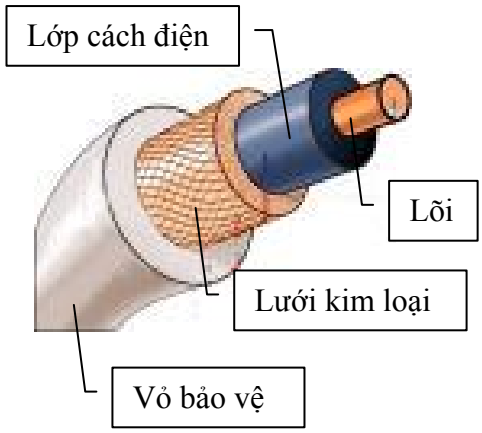
1.1.1. Cáp đôi dây xoắn (Twisted pair cable)

Cáp đôi dây xoắn là cáp gồm hai dây đồng xoắn để tránh gây nhiễu cho các đôi dây khác, có thể kéo dài tới vài km mà không cần khuyếch đại. Giải tần trên cáp dây xoắn đạt khoảng 300–4000Hz, tốc độ truyền đạt vài kbps đến vài trăm Mbps. Cáp xoắn có hai loại:

- Loại có bọc kim loại để tăng cường chống nhiễu gọi là cap STP (Shield Twisted Pair). Loại này trong vỏ bọc kim có thể có nhiều đôi dây. Về lý thuyết thì tốc độ truyền có thể đạt 500 Mb/s nhưng thực tế thấp hơn rất nhiều (chỉ đạt 155 Mbps với cáp dài 100 m)



Hình 2.1 Cáp UTP CAT 5



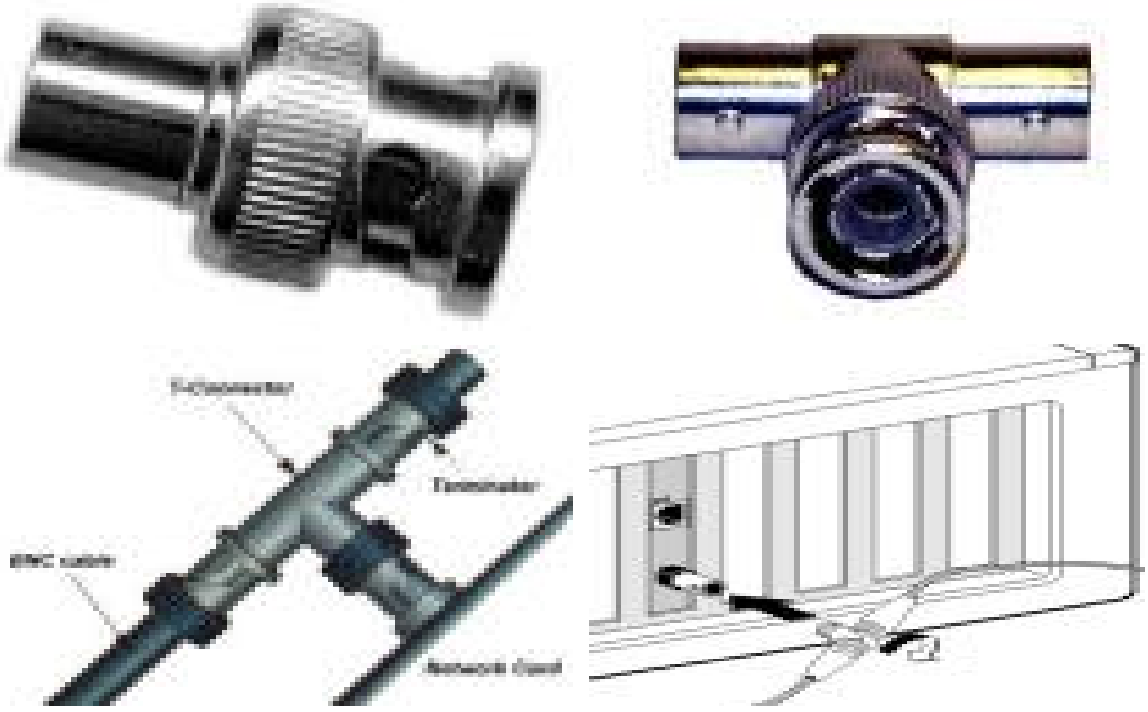
Hình 2.2 Cáp đồng trục

- Loại không bọc kim gọi là UTP (UnShield Twisted Pair), chất lượng kém hơn STP nhưng rất rẻ. Cap UTP được chia làm 5 hạng tuỳ theo tốc độ truyền. Cáp loại 3 dùng cho điện thoại. Cáp loại 5 có thể truyền với tốc độ 100Mb/s rất hay dùng trong các mạng cục bộ vì vừa rẻ vừa tiện sử dụng. Cáp này có 4 đôi dây xoắn nằm trong cùng một vỏ bọc

1.1.2. Cáp đồng trục (Coaxial cable) băng tần cơ sở

Là cáp mà hai dây của nó có lõi lồng nhau, lõi ngoài là lưới kim loại. , Khả năng chống nhiễu rất tốt nên có thể sử dụng với chiều dài từ vài trăm met đến vài km. Có hai loại được dùng nhiều là loại có trở kháng 50 ohm và loại có trở kháng 75 ohm

Dải thông của cáp này còn phụ thuộc vào chiều dài của cáp. Với khoảng cách1 km có thể đạt tốc độ truyền từ 1– 2 Gbps. Cáp đồng trục băng tần cơ sở thường dùng cho các mạng cục bộ. Có thể nối cáp bằng các đầu nối theo chuẩn BNC có hình chữ T. ở VN người ta hay gọi cáp này là cáp gầy do dịch từ tên trong tiếng Anh là ‘Thin Ethernet”.

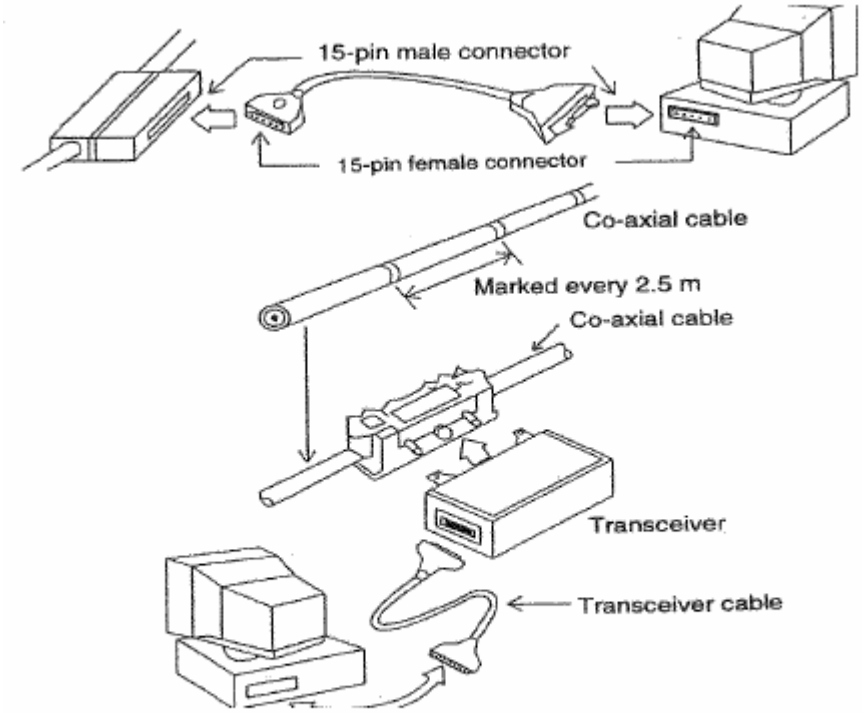


Một loại cáp khác có tên là “Thick Ethernet” mà ta gọi là cáp béo. Loại này thường có màu vàng. Người ta không nối cáp bằng các đầu nối chữ T như cáp gầy mà nối qua các kẹp bấm vào dây. Cứ 2m5 lại có đánh dấu để nối dây

(nếu cần). Từ kẹp đó người ta gắn các tranceiver rồi nối vào máy tính. (Xem hình 2.3)

1.1.3. Cáp đồng trục băng rộng (Broadband Coaxial Cable)

Đây là loại cáp theo tiêu chuẩn truyền hình (thường dùng trong truyền hình cap) có giải thông từ 4 – 300 Khz trên chiều dài 100 km. Thuật ngữ "băng rộng" vốn là thuật ngữ của ngành truyền hình còn trong ngành truyền số liệu điều này chỉ có nghĩa là cáp loại này cho phép truyền thông tin tương tự (analog) mà thôi. Các hệ thống dựa trên cáp đồng trục băng rộng có thể truyền song song nhiều kênh. Việc khuyếch đại tín hiệu chống suy hao có thể làm theo kiểu khuyếch đại tín hiệu tương tự (analog). Để truyền thông cho máy tính cần chuyển tín hiệu số thành tín hiệu tương tự.



Hình 2.3 Kết nối bằng Traceiver

1.1.4. Cáp quang

Dùng để truyền các xung ánh sáng trong lòng một sợi thuỷ tinh phản xạ toàn phần. Môi trường cáp quang rất lý tưởng vì

- Xung ánh sáng có thể đi hàng trăm km mà không giảm cường độ sáng.

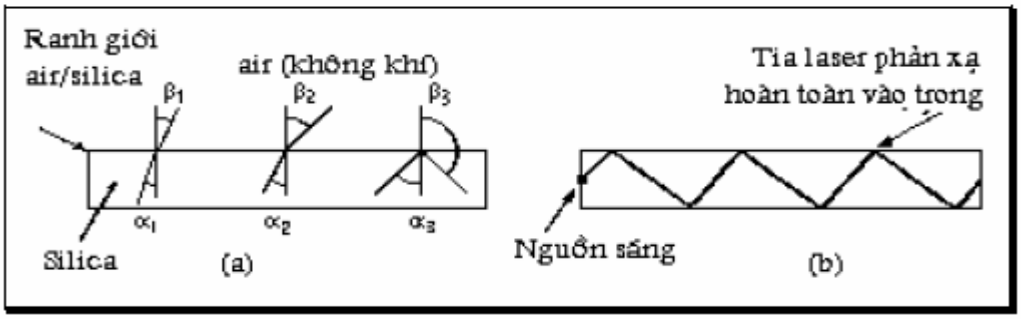
- Giải thông rất cao vì tần số ánh sáng dùng đối với cáp quang cỡ khoảng 1014 –1016

- An toàn và bí mật

- Không bị nhiễu điện từ

Chỉ có hai nhược điểm là khó nối dây và giá thành cao.

Để phát xung ánh sáng người ta dùng các đèn LED hoặc các diod laser. Để nhận người ta dùng các photo diode , chúng sẽ tạo ra xung điện khi bắt được xung ánh sáng



Hình 2.4. Truyền tín hiệu bằng cáp quang

Cáp quang cũng có hai loại

- Loại đa mode (multi mode fiber): khi góc tới thành dây dẫn lớn đến một mức nào đó thì có hiện tượng phản xạ toàn phần. Nhiều tia sáng có thể cùng truyền miễn là góc tới của chúng đủ lớn. Các cap đa mode có đường kính khoảng 50 µ

- Loại đơn mode (single mode fiber): khi đường kính dây dẫn bằng bước sóng thì cáp quang giống như một ống dẫn sóng, không có hiện tượng phản xạ nhưng chỉ cho một tia đi. Loại này có đường kính khoản 8 µ và phải dùng diode laser. Cáp quang đơn mode có thể cho phép truyền xa tới hàng trăm km mà không cần phải khuyếch đại.

1.2. Các thiết bị ghép nối

1.2.1. Card giao tiếp mạng (Network Interface Card viết tắt là NIC)

Đó là một card được cắm trực tiếp vào máy tính. Trên đó có các mạch điện giúp cho việc tiếp nhận (receiver) hoặc/và phát (transmitter) tín hiệu lên mạng. Người ta thường dùng từ tranceiver để chỉ thiết bị (mạch) có cả hai chức năng thu và phát. Transceiver có nhiều loại vì phải thích hợp đối với cả môi trường truyền và do đó cả đầu nối. Ví dụ với cáp gầy card mạng cần có đường giao tiếp theo kiểu BNC, với cáp UTP cần có đầu nối theo kiểu giắc điện thoại RJ45, cáp béo dùng đường nối kiểu AUI , với cáp quang phải có những transceiver cho phép chuyển tín hiệu điện thành các xung ánh sáng và ngược lại.

Để dễ ghép nối, nhiều card có thể có nhiều đầu nối ví dụ BNC cho cáp gầy, RJ45 cho UTP hay AUI cho cáp béo

Trong máy tính thường để sẵn các khe cắm để bổ sung các thiết bị ngoại vi hay cắm các thiết bị ghép nối.

1.2.2. Bộ chuyển tiếp (REPEATER)

Tín hiệu truyền trên các khoảng cách lớn có thể bị suy giảm. Nhiệm vụ của các repeater là hồi phục tín hiệu để có thể truyền tiếp cho các trạm khác. Một số repeater đơn giản chỉ là khuyếch đại tín hiệu. Trong trường hợp đó cả tín hiệu bị méo cũng sẽ bị khuyếch đại. Một số repeater có thể chỉnh cả tín hiệu.

1.2.3. Các bộ tập trung (Concentrator hay HUB)

HUB là một loại thiết bị có nhiều đầu để cắm các đầu cáp mạng. HUB có thể có nhiều loại ổ cắm khác nhau phù hợp với kiểu giắc mạng RJ45, AUI hay BCN. Như vậy người ta sử dụng HUB để nối dây theo kiểu hình sao. Ưu điểm của kiểu nối này là tăng độ độc lập của các máy . Nếu dây nối tới một máy nào đó tiếp xúc không tốt cũng không ảnh hưởng đến máy khác.

Đặc tính chủ yếu của HUB là hệ thống chuyển mạch trung tâm trong mạng có kiến trúc hình sao với việc chuyển mạch được thực hiện theo hai cách: store-and-forward hoặc on-the-fly. Tuy nhiên hệ thống chuyển mạch trung tâm làm nảy sinh vấn đề khi lỗi xảy ra ở chính trung tâm, vì vậy hướng phát triển trong suốt nhiều năm qua là khử lỗi để làm tăng độ tin cậy của HUB.

Có loại HUB thụ động (passive HUB) là HUB chỉ đảm bảo chức năng kết nối hoàn toàn không xử lý lại tín hiệu. Khi đó không thể dùng HUB để tăng khoảng cách giữa hai máy trên mạng.

HUB chủ động (active HUB) là HUB có chức năng khuyếch đại tín hiệu để chống suy hao. Với HUB này có thể tăng khoảng cách truyền giữa các máy.

HUB thông minh (intelligent HUB) là HUB chủ động nhưng có khả năng tạo ra các gói tin mang tin tức về hoạt động của mình và gửi lên mạng để người quản trị mạng có thể thực hiện quản trị tự động

1.2.4. Switching Hub (hay còn gọi tắt là switch)

Là các bộ chuyển mạch thực sự. Khác với HUB thông thường, thay vì chuyển một tín hiệu đến từ một cổng cho tất cả các cổng, nó chỉ chuyển tín hiệu đến cổng có trạm đích. Do vậy Switch là một thiết bị quan trọng trong các mạng cục bộ lớn dùng để phân đoạn mạng. Nhờ có switch mà đụng độ trên mạng giảm hẳn. Ngày nay switch là các thiết bị mạng quan trọng cho phép tuỳ biến trên mạng chẳng hạn lập mạng ảo.

Switch thực chất là một loại bridge, về tính năng kỹ thuật, nó là loại bridge có độ trễ nhỏ nhất. Khác với bridge là phải đợi đến hết frame rồi mới truyền, switch sẽ chờ cho đến khi nhận được địa chỉ đích của frame gửi tới và lập tức được truyền đi ngay. Điều này có nghĩa là frame sẽ được gửi tới LAN cần gửi trước khi nó được switch nhận xong hoàn toàn.

1.2.5. Modem

Là tên viết tắt từ hai từ điều chế (MOdulation) và giải điều chế (DEModulation) là thiết bị cho phép điều chế để biến đổi tín hiệu số sang tín hiệu tương tự để có thể gửi theo đường thoại và khi nhận tín hiệu từ đường thoại có thể biến đổi ngược lại thành tín hiệu số. Tuy nhiên có thể sử dụng nó theo kiểu kết nối từ xa theo đường điện thoại

1.2.6. Router

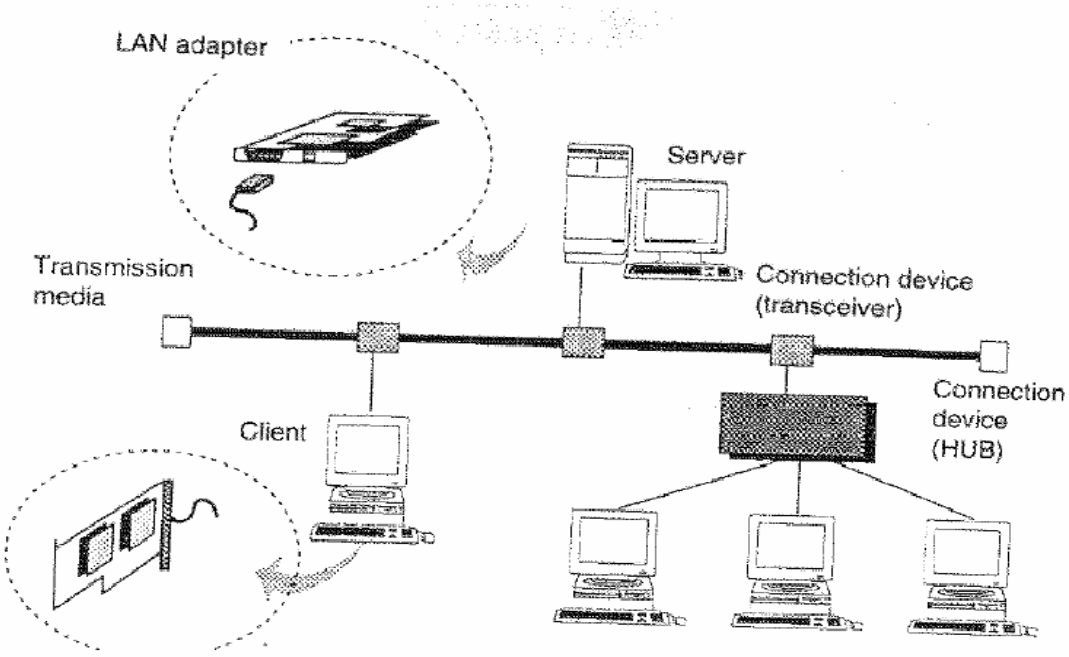
Router là một thiết bị không phải để ghép nối giữa các thiết bị trong một mạng cục bộ mà dùng để ghép nối các mạng cục bộ với nhau thành mạng rộng. Router thực sự là một máy tính làm nhiệm vụ chọn đường cho các gói tin hướng ra ngoài.

Khác với repeaters và bridges, router là thiết bị kết nối mạng độc lập phần cứng, nó được dùng để kết nối các mạng có cùng chung giao thức. Chức năng cơ bản nhất của router là cung cấp một môi trường chuyển mạch gói (packet switching) đáng tin cậy để lưu trữ và truyền số liệu. Để thực hiện điều đó, nó thiết lập các thông tin về các đường truyền hiện có trong mạng, và khi cần nó sẽ cung cấp hai hay nhiều đường truyền giữa hai mạng con bất kỳ tạo ra khả năng mềm dẻo trong việc tìm đường đi hợp lý nhất về một phương diện nào đó.

1.3. Một số kiểu nối mạng thông dụng và các chuẩn

1.3.1.Các thành phần thông thường trên một mạng cục bộ gồm có

- Các máy chủ cung cấp dịch vụ (server)
- Các máy trạm cho người làm việc (workstation)
- Đường truyền (cáp nối)
- Card giao tiếp giữa máy tính và đường truyền (network interface card)
- Các thiết bị nối (connection device)



Hình 2.6. Cấu hình của một mạng cục bộ

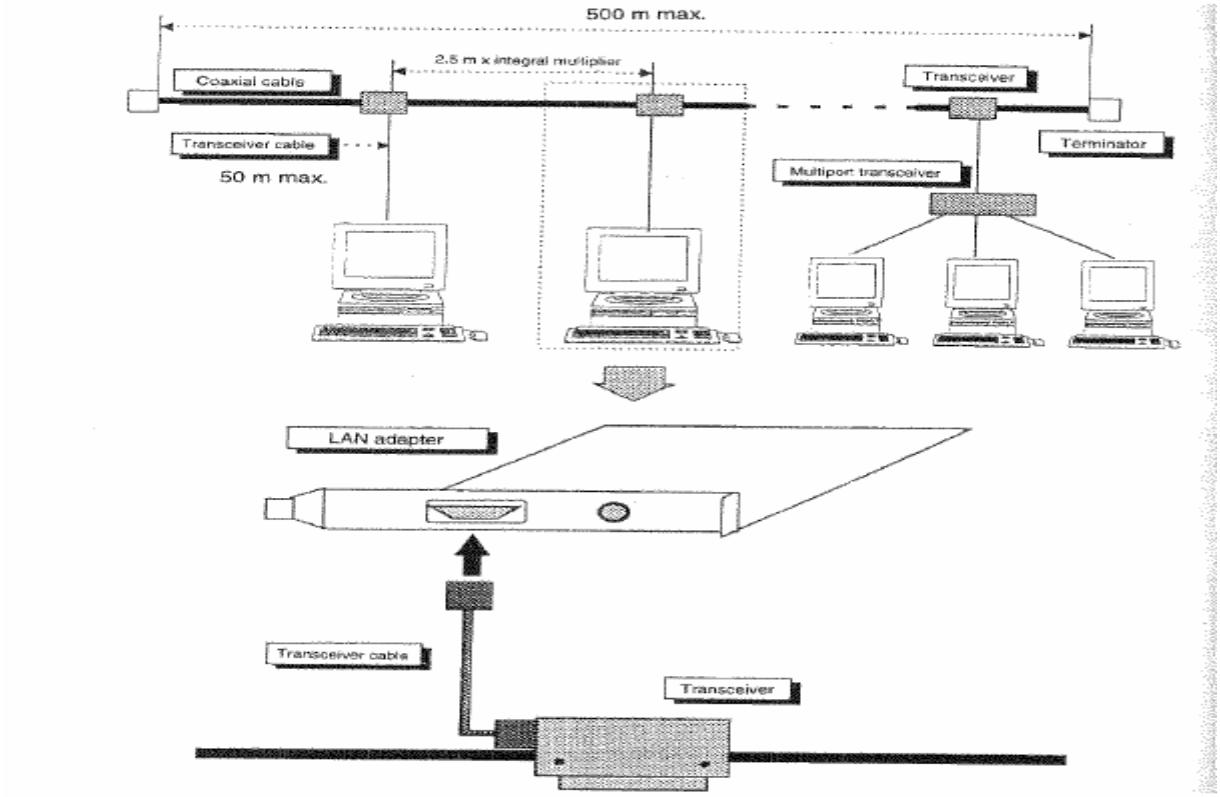
Hai yếu tố được quan tâm hàng đầu khi kết nối mạng cục bộ là tốc độ trong mạng và bán kính mạng. Tên các kiểu mạng dùng theo giao thức CSMA/CD cũng thể hiện điều này. Sau đây là một số kiểu kết nối đó với tốc độ 10 Mb/s khá thông dụng trong thời gian qua và một số thông số kỹ thuật:

Chuẩn	IEEE 802.3		
Kiểu	10BASE5	10BASE2	10BASE-T
Kiểu cáp	Cáp đồng trục	Cáp đồng trục	Cáp UTP
Tốc độ	10 Mb/s		
Độ dài cáp tối đa	500 m/segment	185 m/segment	100 m kể từ HUB
Số các thực thể truyền thông	100 host /segment	30 host / segment	Số cổng của HUB

1.3.2. Kiểu 10BASE5:

Là chuẩn CSMA/CD có tốc độ 10Mb và bán kính 500 m. Kiểu này dùng cáp đồng trục loại thick ethernet (cáp đồng trục béo) với tranceiver. Có thể kết nối vào mạng khoảng 100 máy

Tranceiver:Thiết bị nối giữa card mạng và đường truyền, đóng vai trò là bộ thu-phát



Hình 2.7 Kết nối theo chuẩn 10BASE5

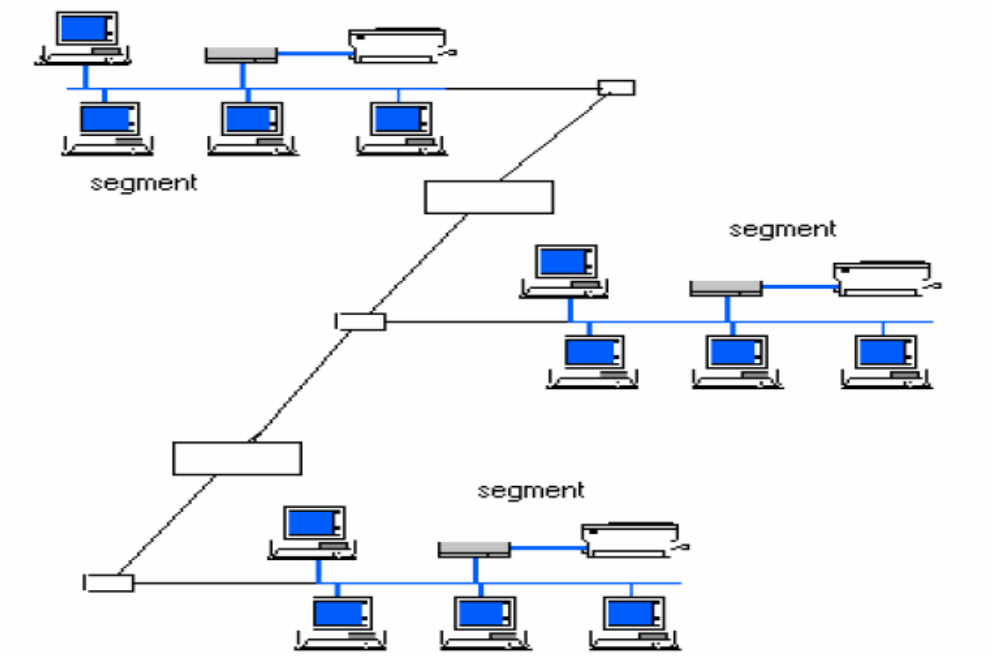
Đặc điểm của chuẩn 10BASE 5

Tốc độ tối đa	10 Mbps
Chiều dài tối đa của đoạn cáp của một phân đoạn (segment)	500 m
Số trạm tối đa trên mỗi đoạn	100
Khoảng cách giữa các trạm	>=2,5 m (bội số của 2,5 m (giảm thiểu hiện tượng giao thoa do sóng đứng trên các đoạn ?))
Khoảng cách tối đa giữa máy trạm và đường trục chung	50 m
Số đoạn kết nối tối đa	2 (=>tối đa có 3 phân đoạn)

Tổng chiều dài tối đa đoạn kết nối (có thể là một đoạn kết nối khi có hai phân đoạn, hoặc hai đoạn kết nối khi có ba phân đoạn)	1000 m
Tổng số trạm + các bộ lặp Repeater	Không quá 1024
Chiều dài tối đa	3*500+1000=2500 m

1.3.3. Kiểu 10BASE2:

Là chuẩn CSMA/CD có tốc độ 10Mb và bán kính 200 m. Kiểu này dùng cáp đồng trục loại thin ethernet với đầu nối BNC. Có thể kết nối vào mạng khoảng 30 máy

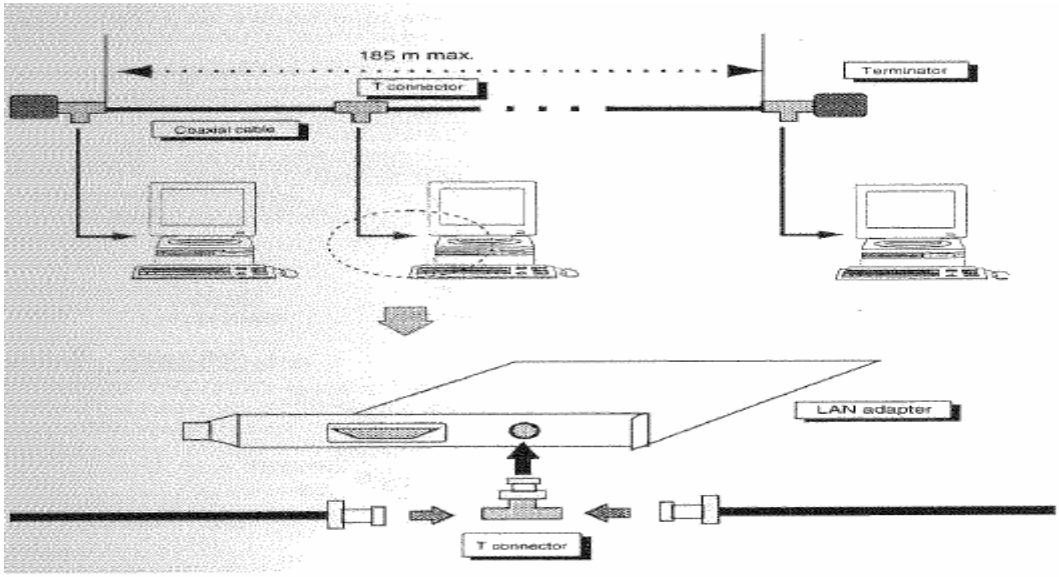


Hình 2.8 Kết nối tối đa 3 phân đoạn mạng

Đặc điểm của chuẩn 10BASE 2

Tốc độ tối đa	10 Mbps
Chiều dài tối đa của đoạn cáp của một phân đoạn (segment)	185 m
Số trạm tối đa trên mỗi đoạn	30
Khoảng cách giữa các trạm	>=0,5 m

Khoảng cách tối đa giữa máy trạm và đường trục chung	0 m
Số đoạn kết nối tối đa	2 (=>tối đa có 3 phân đoạn)
Tổng chiều dài tối đa đoạn kết nối (có thể là một đoạn kết nối khi có hai phân đoạn, hoặc hai đoạn kết nối khi có ba phân đoạn)	1000 m
Tổng số trạm + các bộ lặp Repeater	Không quá 1024



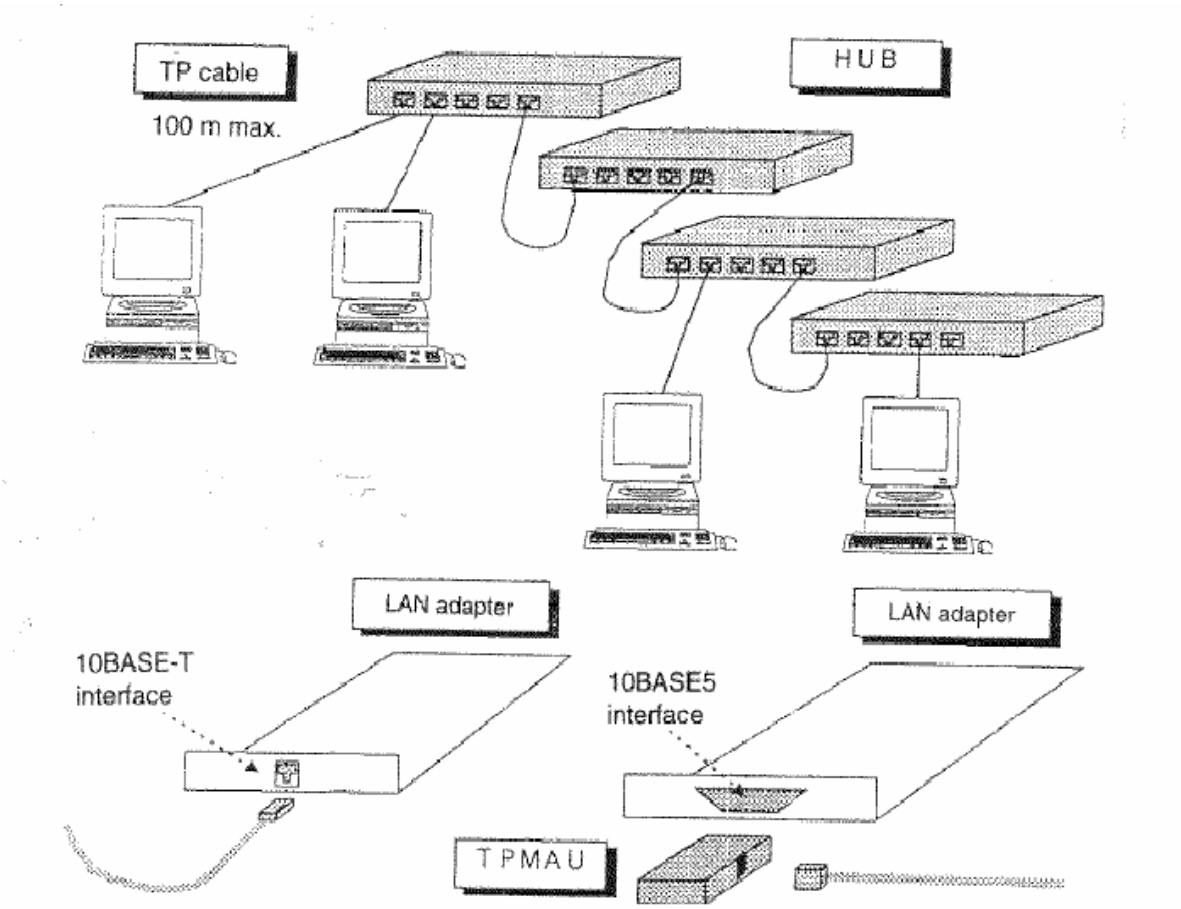
Hình 2.9 Nối theo chuẩn 10BASE2 với cáp đồng trục và đầu nối BNC

1.3.4. Kiểu 10BASE-T

Là kiểu nối dùng HUB có các ổ nối kiểu RJ45 cho các cáp UTP. Ta có thể mở rộng mạng bằng cách tăng số HUB, nhưng cũng không được tăng quá nhiều tầng vì hoạt động của mạng sẽ kém hiệu quả nếu độ trễ quá lớn .

Tốc độ tối đa	10 Mbps
Chiều dài tối đa của đoạn cáp nối giữa máy tính và bộ tập trung HUB	100 m

Hiện nay mô hình phiên bản 100BASE-T bắt đầu được sử dụng nhiều, tốc độ đạt tới 100 Mbps, với card mạng, cab mạng, hub đều phải tuân theo chuẩn 100BASE-T.



Hình 2.10 Nối mạng theo kiểu 10BASE-T với cáp UTP và HUB

1.3.5. Kiểu 10BASE-F

Dùng cab quang (Fiber cab), chủ yếu dùng nối các thiết bị xa nhau, tạo dựng đường trục xương sống (backborn) để nối các mạng LAN xa nhau (2-10 km)

Chương 3. Giới thiệu giao thức TCP/IP

Chương ba cung cấp các kiến thức liên quan đến TCP/IP và địa chỉ IP. Giao thức TCP/IP trở thành giao thức mạng phổ biến nhất nhờ sự phát triển không ngừng của mạng Internet. Các mạng máy tính của các cơ quan, tổ chức, công ty hầu hết đều sử dụng TCP/IP làm giao thức mạng nhờ tính dễ mở rộng và qui hoạch của nó. Đồng thời, do sự phát triển của mạng Internet nên nhu cầu kết nối ra Internet và sử dụng TCP/IP đã trở nên thiết yếu cho mọi đối tượng

Chương này đòi hỏi các học viên phải quen thuộc với các kiến thức cơ bản về hệ nhị phân, các khái niệm bit, byte, chuyển đổi nhị phân, thập phân. Các cách biểu diễn cấu trúc gói tin theo dạng trường bit, byte cũng yêu cầu học viên phải có được hiểu biết cơ sở về kỹ thuật thông tin truyền thông.

I. Giao thức IP

1.1. Họ giao thức TCP/IP

Sự ra đời của họ giao thức TCP/IP gắn liền với sự ra đời của Internet mà tiền thân là mạng **ARPA**net (**A**dvanced **R**esearch **P**rojects **A**gency) do Bộ Quốc phòng Mỹ tạo ra. Đây là bộ giao thức được dùng rộng rãi nhất vì tính mở của nó. Điều đó có nghĩa là bất cứ máy nào dùng bộ giao thức TCP/IP đều có thể nối được vào Internet. Hai giao thức được dùng chủ yếu ở đây là **TCP** (**T**ransmission **C**ontrol **P**rotocol) và **IP** (**I**nternet **P**rotocol). Chúng đã nhanh chóng được đón nhận và phát triển bởi nhiều nhà nghiên cứu và các hãng công nghiệp máy tính với mục đích xây dựng và phát triển một mạng truyền thông mở rộng khắp thế giới mà ngày nay chúng ta gọi là Internet. Phạm vi phục vụ của Internet không còn dành cho quân sự như **ARPA**net nữa mà nó đã mở rộng lĩnh vực cho mọi loại đối tượng sử dụng, trong đó tỷ lệ quan trọng nhất vẫn thuộc về giới nghiên cứu khoa học và giáo dục.

Khái niệm *giao thức (*protocol) là một khái niệm cơ bản của mạng thông tin máy tính. Có thể hiểu một cách khái quát rằng đó chính là tập hợp tất cả các qui tắc cần thiết (các thủ tục, các khuôn dạng dữ liệu, các cơ chế phụ trợ...) cho phép các thao tác trao đổi thông tin trên mạng được thực hiện một cách chính xác và an toàn. Có rất nhiều họ giao thức đang được thực hiện trên mạng thông tin máy tính hiện nay như IEEE 802.X dùng trong mạng cục bộ, CCITT X25 dùng cho mạng diện rộng và đặc biệt là họ giao thức chuẩn của ISO (tổ chức tiêu chuẩn hóa quốc tế) dựa trên mô hình tham chiếu bảy tầng cho việc nối kết các hệ thống mở. Gần đây, do sự xâm nhập của Internet vào Việt nam, chúng ta được làm quen với họ giao thức mới là TCP/IP mặc dù chúng đã xuất hiện từ hơn 20 năm trước đây.

TCP/IP (Transmission Control Protocol/ Internet Protocol) TCP/IP là một họ giao thức cùng làm việc với nhau để cung cấp phương tiện truyền thông liên mạng được hình thành từ những năm 70.

Đến năm 1981, TCP/IP phiên bản 4 mới hoàn tất và được phổ biến rộng rãi cho toàn bộ những máy tính sử dụng hệ điều hành UNIX. Sau này Microsoft cũng đã đưa TCP/IP trở thành một trong những giao thức căn bản của hệ điều hành Windows 9x mà hiện nay đang sử dụng.

Đến năm 1994, một bản thảo của phiên bản IPv6 được hình thành với sự cộng tác của nhiều nhà khoa học thuộc các tổ chức Internet trên thế giới để cải tiến những hạn chế của IPv4.

Khác với mô hình ISO/OSI tầng liên mạng sử dụng giao thức kết nối mạng "không liên kết" (connectionless) IP, tạo thành hạt nhân hoạt động của Internet. Cùng với các thuật toán định tuyến RIP, OSPF, BGP, tầng liên mạng IP cho phép kết nối một cách mềm dẻo và linh hoạt các loại mạng "vật lý" khác nhau như: Ethernet, Token Ring , X.25...

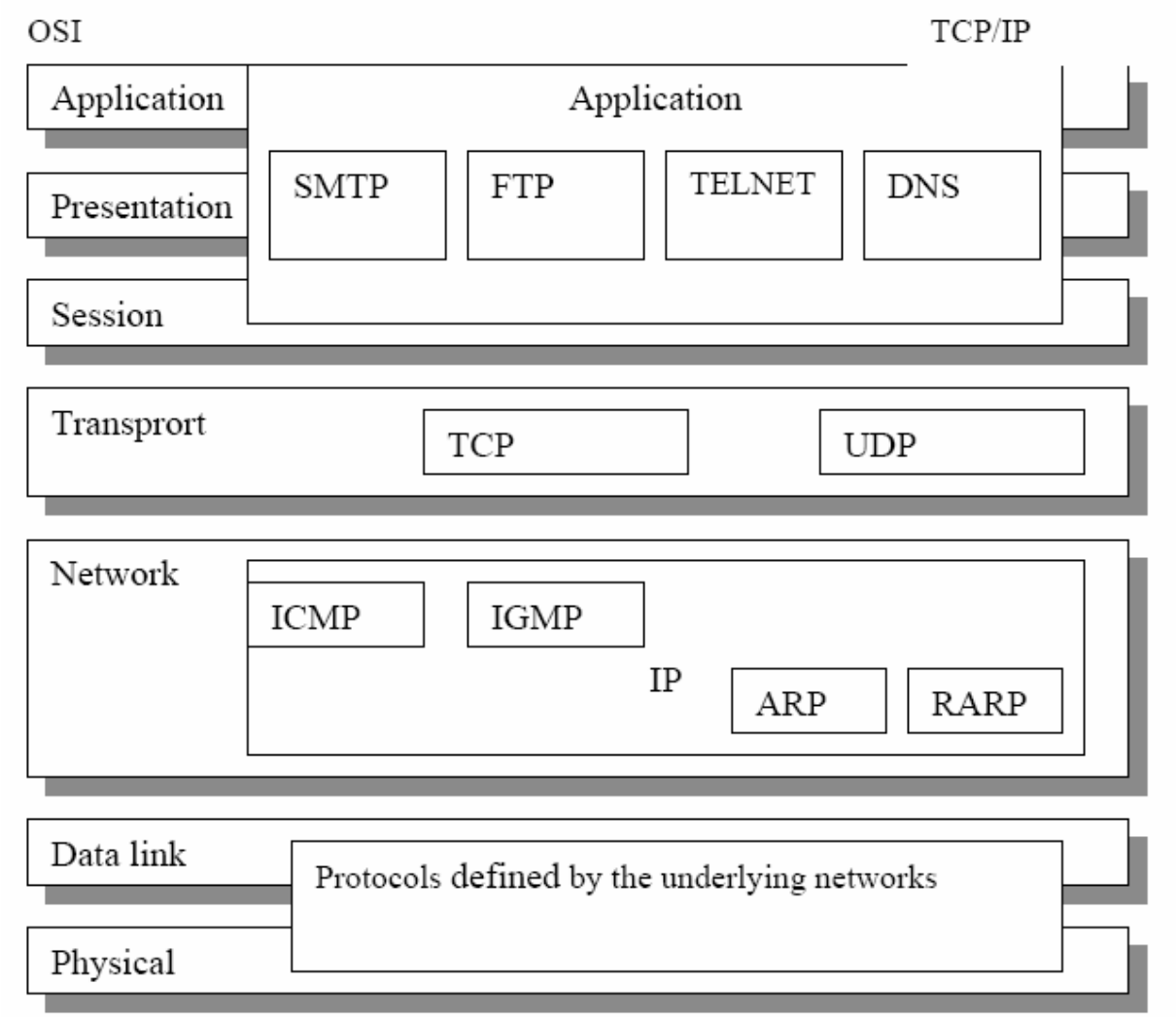
Giao thức trao đổi dữ liệu "có liên kết" (connection - oriented) TCP được sử dụng ở tầng vận chuyển để đảm bảo tính chính xác và tin cậy việc trao đổi dữ liệu dựa trên kiến trúc kết nối "không liên kết" ở tầng liên mạng IP.

Các giao thức hỗ trợ ứng dụng phổ biến như truy nhập từ xa (telnet), chuyển tệp (FTP), dịch vụ World Wide Web (HTTP), thư điện tử (SMTP), dịch vụ tên miền (DNS) ngày càng được cài đặt phổ biến như những bộ phận cấu thành của các hệ điều hành thông dụng như UNIX (và các hệ điều hành chuyên dụng cùng họ của các nhà cung cấp thiết bị tính toán như AIX của IBM, SINIX của Siemens, Digital UNIX của DEC), Windows9x/NT, Novell Netware,...

Như vậy, TCP tương ứng với lớp 4 cộng thêm một số chức năng của lớp 5 trong họ giao thức chuẩn ISO/OSI. Còn IP tương ứng với lớp 3 của mô hình OSI.

Trong cấu trúc bốn lớp của TCP/IP, khi dữ liệu truyền từ lớp ứng dụng cho đến lớp vật lý, mỗi lớp đều cộng thêm vào phần điều khiển của mình để đảm bảo cho việc truyền dữ liệu được chính xác. Mỗi thông tin điều khiển này được gọi là một header và được đặt ở trước phần dữ liệu được truyền. Mỗi lớp xem tất cả các thông tin mà nó nhận được từ lớp trên là dữ liệu, và đặt phần thông tin điều khiển header của nó vào trước phần thông tin này. Việc cộng thêm vào các header ở mỗi lớp trong quá trình truyền tin được gọi là encapsulation. Quá trình nhận dữ liệu diễn ra theo chiều ngược lại: mỗi lớp sẽ tách ra phần header trước khi truyền dữ liệu lên lớp trên.

Mỗi lớp có một cấu trúc dữ liệu riêng, độc lập với cấu trúc dữ liệu được dùng ở lớp trên hay lớp dưới của nó. Sau đây là giải thích một số khái niệm thường gặp.



Hình 3.1 Mô hình OSI và mô hình kiến trúc của TCP/IP

Stream là dòng số liệu được truyền trên cơ sở đơn vị số liệu là Byte.

Số liệu được trao đổi giữa các ứng dụng dùng TCP được gọi là stream, trong khi dùng UDP, chúng được gọi là message.

Mỗi gói số liệu TCP được gọi là segment còn UDP định nghĩa cấu trúc dữ liệu của nó là packet.

Lớp Internet xem tất cả các dữ liệu như là các khối và gọi là datagram. Bộ giao thức TCP/IP có thể dùng nhiều kiểu khác nhau của lớp mạng dưới cùng, mỗi loại có thể có một thuật ngữ khác nhau để truyền dữ liệu.

Phần lớn các mạng kết cấu phần dữ liệu truyền đi dưới dạng các packets hay là các frames.

Application	Stream
Transport	Segment/datagram
Internet	Datagram
Network Access	Frame

Hình 3.2 Cấu trúc dữ liệu tại các lớp của TCP/IP

Lớp truy nhập mạng

Network Access Layer là lớp thấp nhất trong cấu trúc phân bậc của TCP/IP. Những giao thức ở lớp này cung cấp cho hệ thống phương thức để truyền dữ liệu trên các tầng vật lý khác nhau của mạng. Nó định nghĩa cách thức truyền các khối dữ liệu (datagram) IP. Các giao thức ở lớp này phải biết chi tiết các phần cấu trúc vật lý mạng ở dưới nó (bao gồm cấu trúc gói số liệu, cấu trúc địa chỉ...) để định dạng được chính xác các gói dữ liệu sẽ được truyền trong từng loại mạng cụ thể.

So sánh với cấu trúc OSI/OSI, lớp này của TCP/IP tương đương với hai lớp Datalink, và Physical.

Chức năng định dạng dữ liệu sẽ được truyền ở lớp này bao gồm việc nhúng các gói dữ liệu IP vào các frame sẽ được truyền trên mạng và việc ánh xạ các địa chỉ IP vào địa chỉ vật lý được dùng cho mạng.

Lớp liên mạng

Internet Layer là lớp ở ngay trên lớp Network Access trong cấu trúc phân lớp của TCP/IP. Internet Protocol là giao thức trung tâm của TCP/IP và là phần quan trọng nhất của lớp Internet. IP cung cấp các gói lưu chuyển cơ bản mà thông qua đó các mạng dùng TCP/IP được xây dựng.

1.2. Chức năng chính của - Giao thức liên mạng IP(v4)

Trong phần này trình bày về giao thức IPv4 (để cho thuận tiện ta viết IP có nghĩa là đề cập đến IPv4).

Mục đích chính của IP là cung cấp khả năng kết nối các mạng con thành liên mạng để truyền dữ liệu. IP cung cấp các chức năng chính sau:

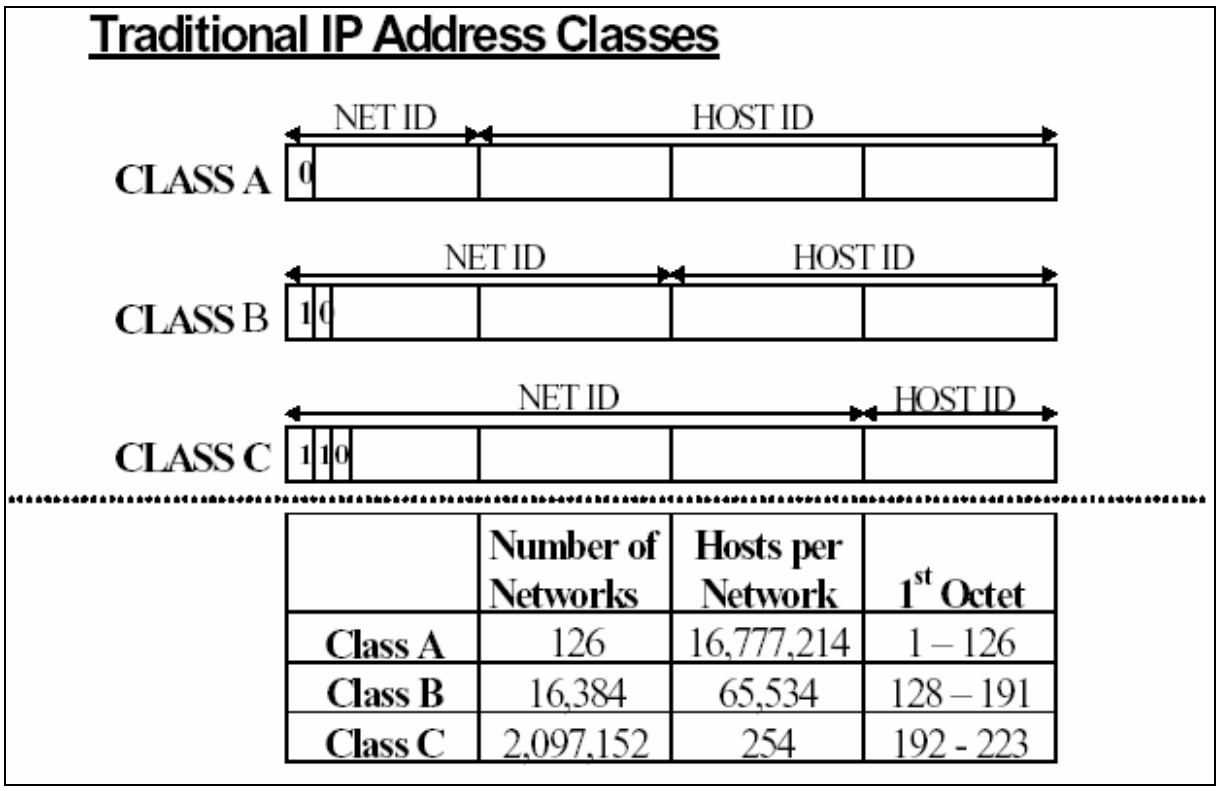
- Định nghĩa cấu trúc các gói dữ liệu là đơn vị cơ sở cho việc truyền dữ liệu trên Internet.

- Phần định danh địa chỉ mạng Network Number

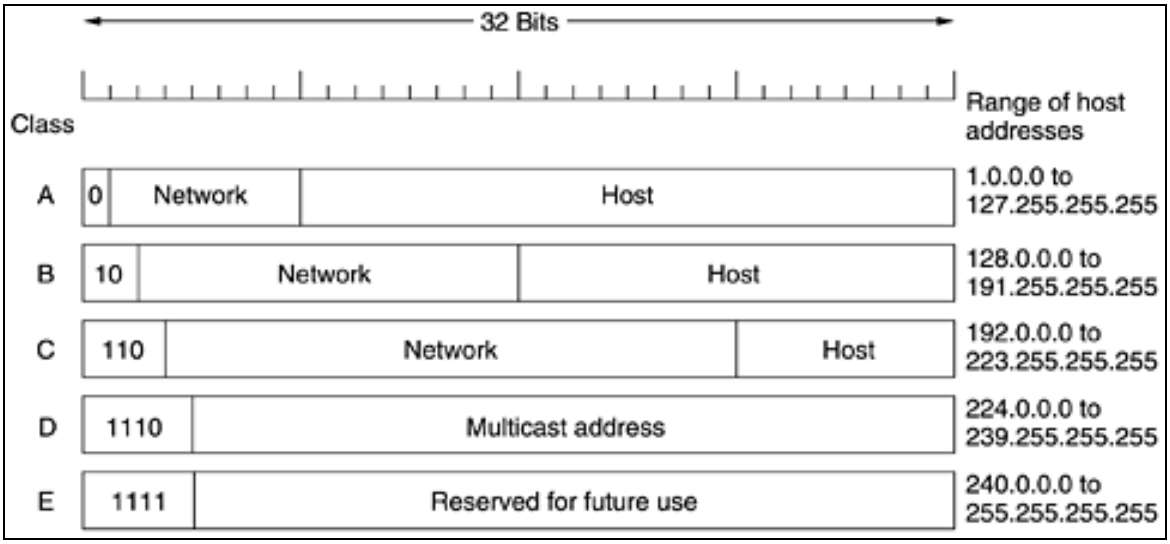
- Phần định danh địa chỉ các trạm làm việc trên mạng đó Host Number

Ví dụ 128.4.70.9 là một địa chỉ IP

Do tổ chức và độ lớn của các mạng con của liên mạng có thể khác nhau, người ta chia các địa chỉ IP thành 5 lớp ký hiệu A,B,C, D, E với cấu trúc được xác định trên hình 3.4.



	Number of Networks	Hosts per Network	1st Octet
Class A	126	16,777,214	1 – 126
Class B	16,384	65,534	128 – 191
Class C	2,097,152	254	192 - 223



Hình 3.4. Cấu trúc địa chỉ IP

Các bit đầu tiên của byte đầu tiên được dùng để định danh lớp địa chỉ (0- lớp A; 10 lớp B; 110 lớp C; 1110 lớp D; 11110 lớp E).

- Lớp A cho phép định danh tới 126 mạng (sử dụng byte đầu tiên), với tối đa 16 triệu host (3 byte còn lại, 24 bits) cho mỗi mạng. Lớp này được dùng cho các mạng có số trạm cực lớn. Tại sao lại có 126 mạng trong khi dùng 8 bits? Lí do đầu tiên, 127.x (01111111) dùng cho địa chỉ loopback, thứ 2 là bit đầu tiên của byte đầu tiên bao giờ cũng là 0, 1111111(127). Dạng địa chỉ lớp A (network number. host.host.host). Nếu dùng ký pháp thập phân cho phép 1 đến 126 cho vùng đầu, 1 đến 255 cho các vùng còn lại.

- Lớp B cho phép định danh tới 16384 mạng (10111111.11111111.host.host), với tối đa 65535 host trên mỗi mạng. Dạng của lớp B (network number. Network number.host.host). Nếu dùng ký pháp thập phân cho phép 128 đến 191 cho vùng đầu, 1 đến 255 cho các vùng còn lại

- Lớp C cho phép định danh tới 2.097.150 mạng và tối đa 254 host cho mỗi mạng. Lớp này được dùng cho các mạng có ít trạm. Lớp C sử dụng 3 bytes đầu định danh địa chỉ mạng (110xxxxx). Dạng của lớp C (network number. Network number.Network number.host). Nếu dùng dạng ký pháp thập phân cho phép 129 đến 233 cho vùng đầu và từ 1 đến 255 cho các vùng còn lại.

- Lớp D dùng để gửi IP datagram tới một nhóm các host trên một mạng. Tất cả các số lớn hơn 233 trong trường đầu là thuộc lớp D

- Lớp E dự phòng để dùng trong tương lai

Như vậy địa chỉ mạng cho lớp: A: từ 1 đến 126 cho vùng đầu tiên, 127 dùng cho địa chỉ loopback, B từ 128.1.0.0 đến 191.255.0.0, C từ 192.1.0.0 đến 233.255.255.0

Ví dụ:

192.1.1.1 địa chỉ lớp C có địa chỉ mạng 192.1.1.0, địa chỉ host là 1

200.6.5.4 địa chỉ lớp C có địa chỉ mạng 200.6.5, địa chỉ mạng là 4

150.150.5.6 địa chỉ lớp B có địa chỉ mạng 150.150.0.0, địa chỉ host là 5.6

9.6.7.8 địa chỉ lớp A có địa chỉ mạng 9.0.0.0, địa chỉ host là 6.7.8

128.1.0.1 địa chỉ lớp B có địa chỉ mạng 128.1.0.0, địa chỉ host là 0.1

Network ID không thể là 127 - dành cho chức năng loop-back là kiểm tra vòng lặp tại thiết bị, không thực hiện chuyển dữ liệu.

Network ID và host ID không thể là 255 (các bit đặt là 1) - 255 là địa chỉ quảng bá

Network ID và host ID không thể là 0 (các bit đặt là 0) - 0 có nghĩa là chính mạng đó.

IP Address	**160.30.20**.10	**10100000 00011110 00010100** 00001010
Subnet Mask	**255.255.255**.0	**11111111 11111111 11111111** 00000000
Result	**160.30.20**.0	**10100000 00011110 00010100** 00000000

IP Address	**160.30.20**.100	**10100000 00011110 11001000** 01100100
Subnet Mask	**255.255.255**.0	**11111111 11111111 11111111** 00000000
Result	**160.30.20**.0	**10100000 00011110 00010100** 00000000

Cách viết mask theo độ dài tiếp đầu ngữ (prefix length).

Để ngắn gọn có thể viết mask theo số bit 1 liên tiếp tính từ đầu.

Ví dụ 255.255.255.0 có 24 bit 1 do đó viết địa chỉ 160.30.20.10/24.

Theo quy tắc đó: lớp A có mask là 255.0.0.0 (/8),

lớp B - 255.255.0.0 (/16),

lớp C - 255.255.255.0 (/24).

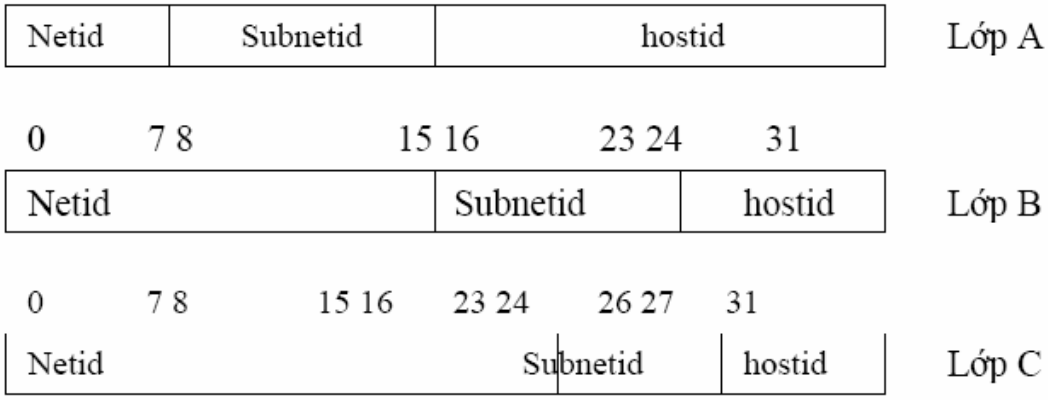
Subneting

Trong nhiều trường hợp, một mạng có thể được chia thành nhiều mạng con (subnet), lúc đó có thể đưa thêm các vùng subnetid để định danh các mạng con. Vùng subnetid được lấy từ vùng hostid, cụ thể đối với 3 lớp A, B, C như sau:

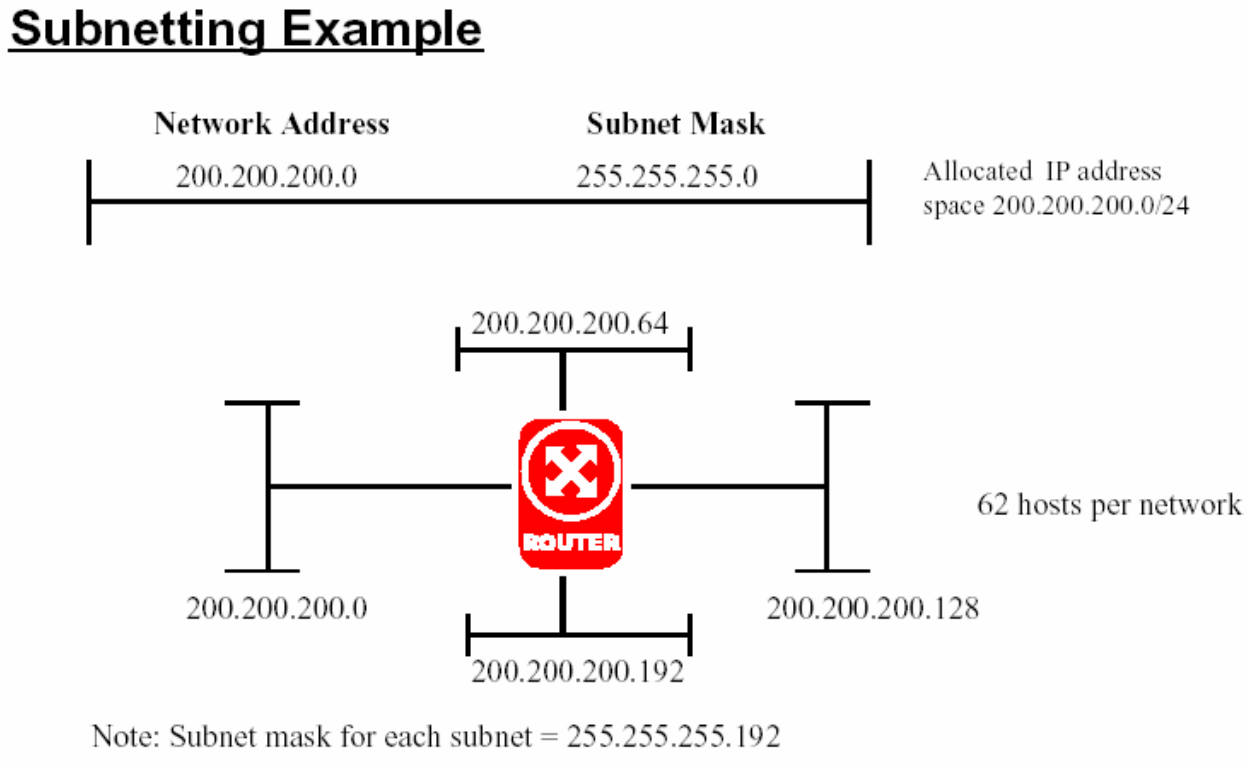
Ví dụ:

17.1.1.1 địa chỉ lớp A có địa chỉ mạng 17, địa chỉ subnet 1, địa chỉ host 1.1

129.1.1.1 địa chỉ lớp B có địa chỉ mạng 129.1, địa chỉ subnet 1, địa chỉ host 1.



Hình 3.6 Bổ sung vùng subnetid



Hình 3.7. Ví dụ SubNet

Subnet Mask là 255.255.255.192 hay là /26

04 mạng nhỏ hơn với địa chỉ mạng là

Mạng 1: 200.200.200.0/26 -> từ 200.200.200.1 đến 200.200.200.62

Mạng 2: 200.200.200.64/26 -> từ 200.200.200.65 đến 200.200.200.126

Mạng 3: 200.200.200.0/128 -> từ 200.200.200.129 đến 200.200.200.190

Mạng 4: 200.200.200.0/192 -> từ 200.200.200.193 đến 200.200.200.254.

Xác định tên máy tính

Mỗi máy tính được gán một địa chỉ IP. Để dễ nhớ thì gán thêm một tên dùng bảng chữ cái, gọi là domain name, ví dụ dhsp.edu.vn.

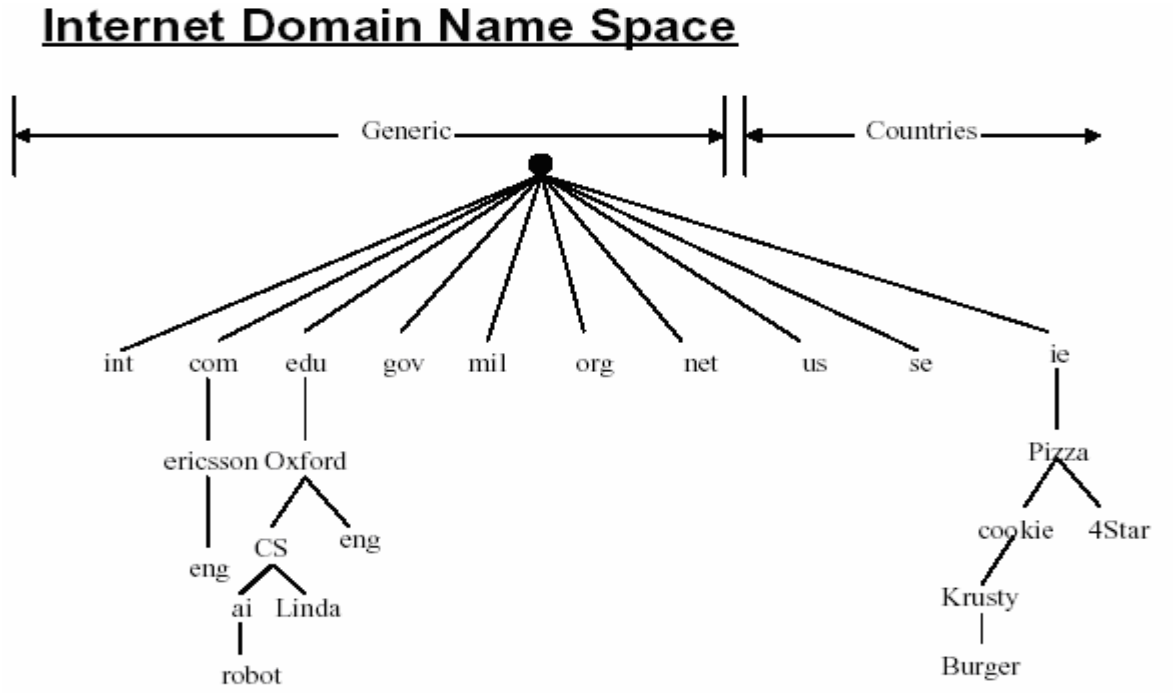
Để xác định tên của một máy tính, cần một phương pháp ánh xạ giữa địa chỉ số và tên gọi. Hệ thống xác định tên từ IP là CSDL DNS (Domain Name System).

DNS được tổ chức theo cấu trúc phân hệ, phần gần gốc hơn là tên ở phía bên phải, các hệ thống lớn chia ra các hệ thống nhỏ, và lại được chia tiếp theo. Các DNS có các loại chính như sau: loại top-level - bậc cao; loại thông thường; loại theo quốc gia.

Các loại thông thường:

- **com** (Commercial organisation)
- **edu** (Educational institution)
- **gov** (Government organisation)
- **mil** (Military group)
- **net** (Major network support centre)
- **org** (Organisation other than those above)
- **int** (International organisation)

Loại tên nước: hai chữ cái viết tắt (ISO 3166 quy định): vd Việt nam là vn; Anh - uk; Úc - au, vv.



Hình 3.9

Cách thức xác định tên và IP từ tên:

1. Client gửi yêu cầu xác định IP cho tên mr-a.khoacntt.dhsphn.edu.vn tới Local Name Server.

2. LNS không có quyền đối với tên này nên yêu cầu Root name server.

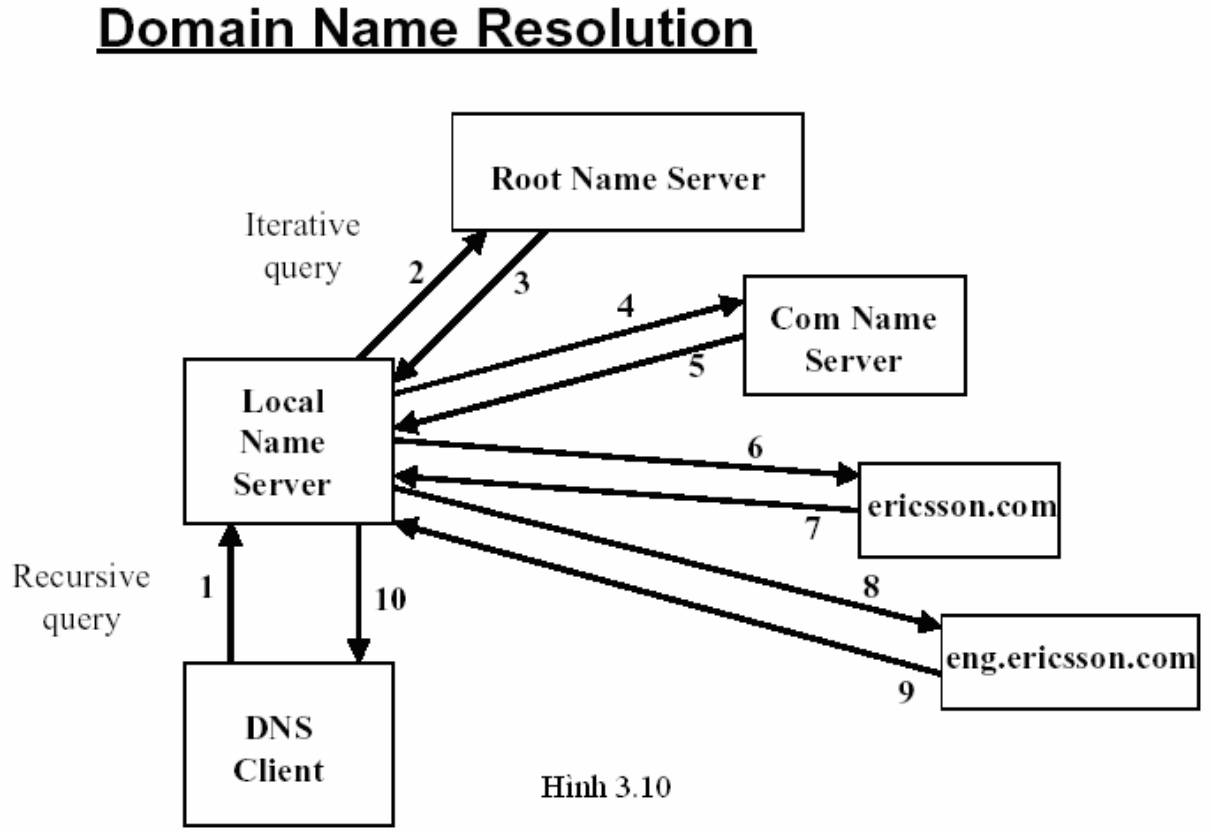
3. RNS gửi lại LNS địa chỉ IP của vn name server.

4. LNS yêu cầu tới vn server

5. vn server trả lời địa chỉ IP của DNS quản lý tên miền edu.vn.

6. LNS yêu cầu tới server trên và nhận trả lời cho server name tiếp theo,...

Quá trình tiếp diễn tới khi đạt được name server quản lý chính xác tên như trên.



Hình 3.10

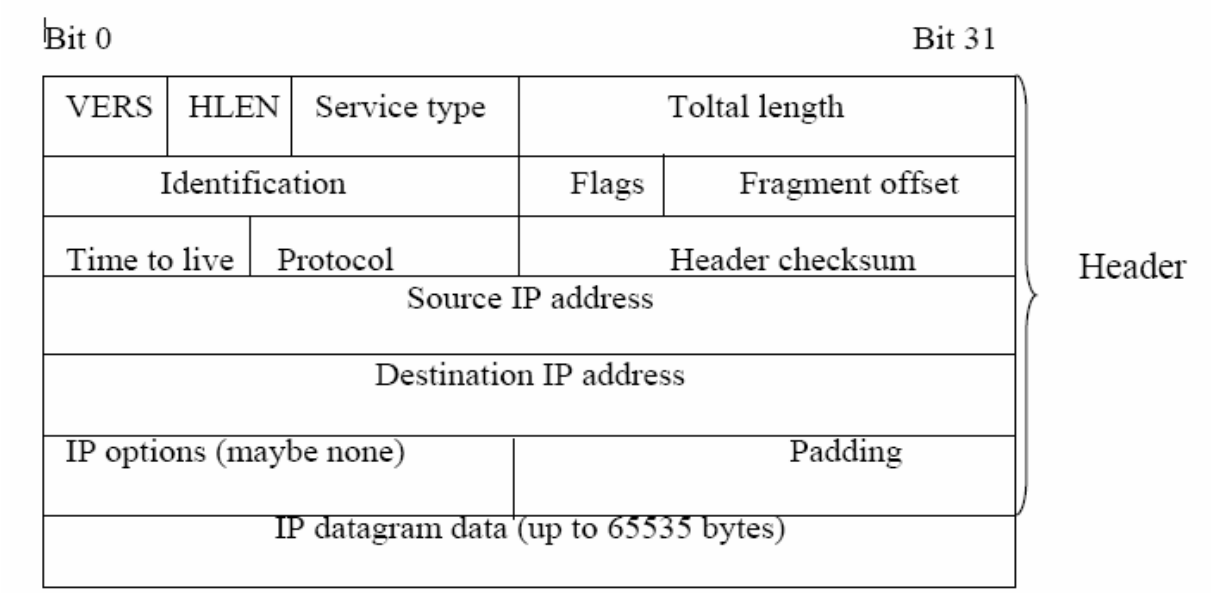
Các Name Server giữ các thông tin về xử lý tên miền trong bộ đệm, khi có thông tin sẽ gửi một thông báo gồm tên miền và địa chỉ IP tới Client và cách liên lạc với name server đó. Do đó việc xử lý tên sẽ nhanh hơn.

Các bộ đệm có cơ chế đặt thời gian sống (Time-To-Live) cho các thông tin lưu trữ.

1.2.2. Cấu trúc gói dữ liệu IP

IP là giao thức cung cấp dịch vụ truyền thông theo kiểu "không liên kết" (connectionless). Phương thức không liên kết cho phép cặp trạm truyền nhận không cần phải thiết lập liên kết trước khi truyền dữ liệu và do đó không cần phải giải phóng liên kết khi không còn nhu cầu truyền dữ liệu nữa. Phương thức kết nối "không liên kết" cho phép thiết kế và thực hiện giao thức trao đổi dữ liệu đơn giản (không có cơ chế phát hiện và khắc phục lỗi truyền). Cũng chính vì vậy độ tin cậy trao đổi dữ liệu của loại giao thức này không cao.

Các gói dữ liệu IP được định nghĩa là các datagram. Mỗi datagram có phần tiêu đề (header) chứa các thông tin cần thiết để chuyển dữ liệu (ví dụ địa chỉ IP của trạm đích). Nếu địa chỉ IP đích là địa chỉ của một trạm nằm trên cùng một mạng IP với trạm nguồn thì các gói dữ liệu sẽ được chuyển thẳng tới đích; nếu địa chỉ IP đích không nằm trên cùng một mạng IP với máy nguồn thì các gói dữ liệu sẽ được gửi đến một máy trung chuyển, IP gateway để chuyển tiếp. IP gateway là một thiết bị mạng IP đảm nhận việc lưu chuyển các gói dữ liệu IP giữa hai mạng IP khác nhau. Hình 3.11 mô tả cấu trúc gói số liệu IP.



Hình 3.11. Cấu trúc gói dữ liệu IP

- VER (4 bits) : chỉ Version hiện hành của IP được cài đặt.

- IHL (4 bits) : chỉ độ dài phần tiêu đề (Internet Header Length) của datagram, tính theo đơn vị word (32 bits). Nếu không có trường này thì độ dài mặc định của phần tiêu đề là 5 từ.

- Type of service (8 bits): cho biết các thông tin về loại dịch vụ và mức ưu tiên của gói IP, có dạng cụ thể như sau:

Precedence	D	T	R	Unused

Trong đó:

Precedence (3 bits): chỉ thị về quyền ưu tiên gửi datagram, cụ thể là:

111 Network Control (cao nhất) 011- flash

110 Internetwork Control 010 Immediate

101 CRITIC/ECP 001 Priority

100 Flas Override 000 Routine (thấp nhất)

D (delay) (1 bit) : chỉ độ trễ yêu cầu

D=0 độ trễ bình thường,

D=1 độ trễ thấp

T (Throughput) (1 bit) : chỉ số thông lượng yêu cầu

T=1 thông lượng bình thường

T=1 thông lượng cao

R (Reliability) (1 bit): chỉ độ tin cậy yêu cầu

R=0 độ tin cậy bình thường

R=1 độ tin cậy cao

- Total Length (16 bits): chỉ độ dài toàn bộ datagram, kể cả phần header (tính theo đơn vị bytes), vùng dữ liệu của datagram có thể dài tới 65535 bytes.

- Identification (16 bits) : cùng với các tham số khác như (Source Address và Destination Address) tham số này dùng để định danh duy nhất cho một datagram trong khoảng thời gian nó vẫn còn trên liên mạng

- Flags (3 bits) : liên quan đến sự phân đoạn (fragment) các datagram. Cụ thể

O	DF	MF

Bit 0 : reserved chưa sử dụng luôn lấy giá trị 0

Bit 1 : (DF)= 0 (may fragment)

1 (Don't Fragment)

Bit 2 : (MF)= 0 (Last Fragment)

1 (More Fragment)

- Fragment Offset (13 bits) : chỉ vị trí của đoạn (fragment) ở trong datagram, tính theo đơn vị 64 bits, có nghĩa là mỗi đoạn (trừ đoạn cuối cùng) phải chứa một vùng dữ liệu có độ dài là bội của 64 bits.

- Time To Live (TTL-8 bits) : quy định thời gian tồn tại của một gói dữ liệu trên liên mạng để tránh tình trạng một datagram bị quẩn trên mạng. Giá trị này được đặt lúc bắt đầu gửi đi và sẽ giảm dần mỗi khi gói dữ liệu được xử lý tại những điểm trên đường đi của gói dữ liệu (thực chất là tại các router). Nếu giá trị này bằng 0 trước khi đến được đích, gói dữ liệu sẽ bị huỷ bỏ.

- Protocol (8 bits): chỉ giao thức tầng kế tiếp sẽ nhận vùng dữ liệu ở trạm đích (hiện tại thường là TCP hoặc UDP được cài đặt trên IP).

- Header checksum (16 bits): mã kiểm soát lỗi sử dụng phương pháp CRC (Cyclic Redundancy Check) dùng để đảm bảo thông tin về gói dữ liệu được truyền đi một cách chính xác (mặc dù dữ liệu có thể bị lỗi). Nếu như việc kiểm tra này thất bại, gói dữ liệu sẽ bị huỷ bỏ tại nơi xác định được lỗi. Cần chú ý là IP không cung cấp một phương tiện truyền tin cậy bởi nó không cung cấp cho ta một cơ chế để xác nhận dữ liệu truyền tại điểm nhận hoặc tại những điểm trung gian. Giao thức IP không có cơ chế Error Control cho dữ liệu truyền đi, không có cơ chế kiểm soát luồng dữ liệu (flow control).

- Source Address (32 bits): địa chỉ của trạm nguồn.

- Destination Address (32 bits): địa chỉ của trạm đích.

- Option (có độ dài thay đổi) sử dụng trong một số trường hợp, nhưng thực tế chúng rất ít dùng. Option bao gồm bảo mật, chức năng định tuyến đặc biệt

- Padding (độ dài thay đổi): vùng đệm, được dùng để đảm bảo cho phần header luôn kết thúc ở một mốc 32 bits

- Data (độ dài thay đổi): vùng dữ liệu có độ dài là bội của 8 bits, tối đa là 65535 bytes.

1.2.3. Phân mảnh và hợp nhất các gói IP

Các gói dữ liệu IP phải được nhúng trong khung dữ liệu ở tầng liên kết dữ liệu tương ứng, trước khi chuyển tiếp trong mạng. Quá trình nhận một gói dữ liệu IP diễn ra ngược lại. Ví dụ, với mạng Ethernet ở tầng liên kết dữ liệu quá trình chuyển một gói dữ liệu diễn ra như sau. Khi gửi một gói dữ liệu IP cho mức Ethernet, IP chuyển cho mức liên kết dữ liệu các thông số địa chỉ Ethernet đích, kiểu khung Ethernet (chỉ dữ liệu mà Ethernet đang mang là của IP) và cuối cùng là gói IP. Tầng liên kết số liệu đặt địa chỉ Ethernet nguồn là địa chỉ kết nối mạng của mình và tính toán giá trị checksum. Trường type chỉ ra kiểu khung là 0x0800 đối với dữ liệu IP. Mức liên kết dữ liệu sẽ chuyển khung dữ liệu theo thuật toán truy nhập Ethernet.

Một gói dữ liệu IP có độ dài tối đa 65536 byte, trong khi hầu hết các tầng liên kết dữ liệu chỉ hỗ trợ các khung dữ liệu nhỏ hơn độ lớn tối đa của gói dữ liệu IP nhiều lần (ví dụ độ dài lớn nhất của một khung dữ liệu Ethernet là 1500 byte). Vì vậy cần thiết phải có cơ chế phân mảnh khi phát và hợp nhất khi thu đối với các gói dữ liệu IP.

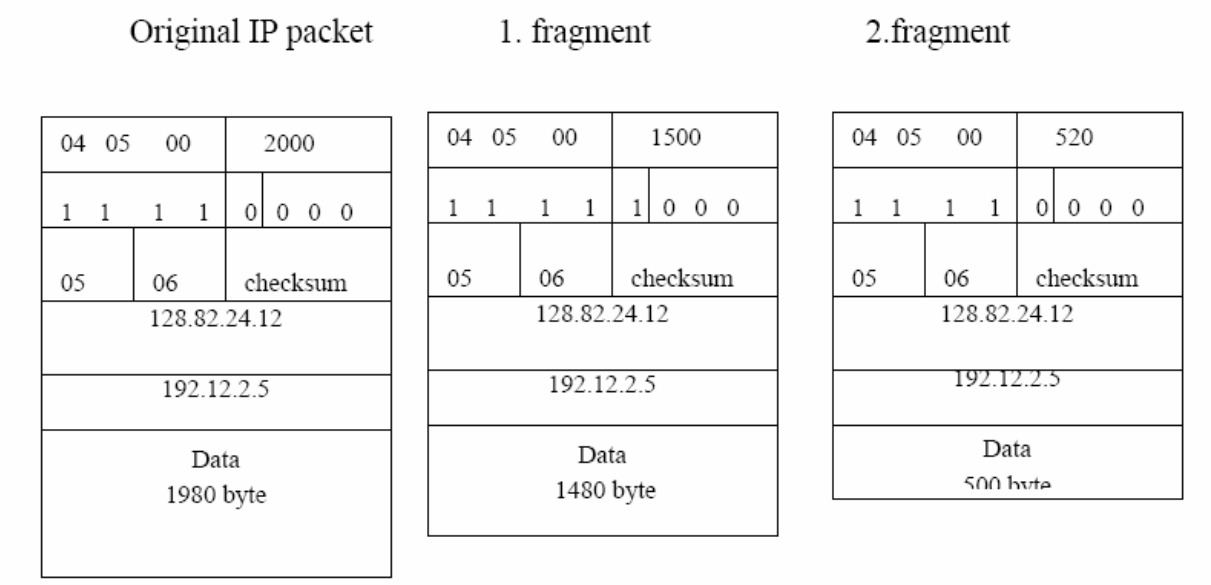
Độ dài tối đa của một gói dữ liệu liên kết là MTU (Maximum Transmit Unit). Khi cần chuyển một gói dữ liệu IP có độ dài lớn hơn MTU của một mạng cụ thể, cần phải chia gói số liệu IP đó thành những gói IP nhỏ hơn để độ dài của nó nhỏ hơn hoặc bằng MTU gọi chung là mảnh (fragment). Trong phần tiêu đề của gói dữ liệu IP có thông tin về phân mảnh và xác định các mảnh có quan hệ phụ thuộc để hợp thành sau này.

Ví dụ Ethernet chỉ hỗ trợ các khung có độ dài tối đa là 1500 byte. Nếu muốn gửi một gói dữ liệu IP gồm 2000 byte qua Ethernet, phải chia thành hai gói nhỏ hơn, mỗi gói không quá giới hạn MTU của Ethernet.

P dùng cờ MF (3 bit thấp của trường Flags trong phần đầu của gói IP) và trường Flagment offset của gói IP (đã bị phân đoạn) để định danh gói IP đó là một phân đoạn và vị trí của phân đoạn này trong gói IP gốc. Các gói cùng trong chuỗi phân mảnh đều có trường này giống nhau. Cờ MF bằng 1 nếu là gói đầu của chuỗi phân mảnh và 0 nếu là gói cuối của gói đã được phân mảnh.

Quá trình hợp nhất diễn ra ngược lại với quá trình phân mảnh. Khi IP nhận được một gói phân mảnh, nó giữ phân mảnh đó trong vùng đệm, cho đến khi nhận được hết các gói IP trong chuỗi phân mảnh có cùng trường định danh. Khi phân mảnh đầu tiên được nhận, IP khởi động một bộ đếm thời gian (giá trị ngầm định là 15s). IP phải nhận hết các phân mảnh kế tiếp trước khi đồng hồ tắt. Nếu không IP phải huỷ tất cả các phân mảnh trong hàng đợi hiện thời có cùng trường định danh.

Khi IP nhận được hết các phân mảnh, nó thực hiện hợp nhất các gói phân mảnh thành các gói IP gốc và sau đó xử lý nó như một gói IP bình thường. IP thường chỉ thực hiện hợp nhất các gói tại hệ thống đích của gói.



Hình 3.12. Nguyên tắc phân mảnh gói dữ liệu

1.2.4. Định tuyến IP

Có hai loại định tuyến:

- Định tuyến trực tiếp: Định tuyến trực tiếp là việc xác định đường nối giữa hai trạm làm việc trong cùng một mạng vật lý.

- Định tuyến không trực tiếp. Định tuyến không trực tiếp là việc xác định đường nối giữa hai trạm làm việc không nằm trong cùng một mạng vật lý và vì vậy, việc truyền tin giữa chúng phải được thực hiện thông qua các trạm trung gian là các gateway.

Để kiểm tra xem trạm đích có nằm trên cùng mạng vật lý với trạm nguồn hay không, người gửi phải tách lấy phần địa chỉ mạng trong phần địa chỉ IP. Nếu hai địa chỉ này có địa chỉ mạng giống nhau thì datagram sẽ được truyền đi trực tiếp; ngược lại phải xác định một gateway, thông qua gateway này chuyển tiếp các datagram.

Khi một trạm muốn gửi các gói dữ liệu đến một trạm khác thì nó phải đóng gói datagram vào một khung (frame) và gửi các frame này đến gateway gần nhất. Khi một frame đến một gateway, phần datagram đã được đóng gói sẽ được tách ra và IP routing sẽ chọn gateway tiếp dọc theo đường dẫn đến đích. Datagram sau đó lại được đóng gói vào một frame khác và gửi đến mạng vật lý để gửi đến gateway tiếp theo trên đường truyền và tiếp tục như thế cho đến khi datagram được truyền đến trạm đích.

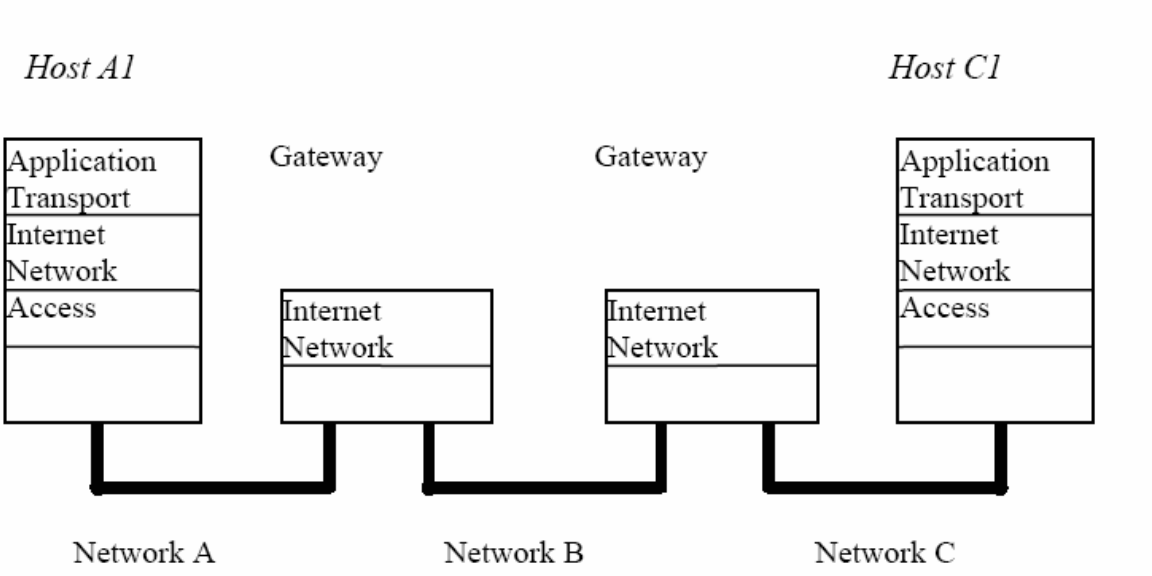
Chiến lược định tuyến:

Trong thuật ngữ truyền thống của TCP/IP chỉ có hai kiểu thiết bị, đó là các cổng truyền (gateway) và các trạm (host). Các cổng truyền có vai trò gửi các gói dữ liệu, còn các trạm thì không. Tuy nhiên khi một trạm được nối với nhiều mạng thì nó cũng có thể định hướng cho việc lưu chuyển các gói dữ liệu giữa các mạng và lúc này nó đóng vai trò hoàn toàn như một gateway.

Các trạm làm việc lưu chuyển các gói dữ liệu xuyên suốt qua cả bốn lớp, trong khi các cổng truyền chỉ chuyển các gói đến lớp Internet là nơi quyết định tuyến đường tiếp theo để chuyển tiếp các gói dữ liệu.

Các máy chỉ có thể truyền dữ liệu đến các máy khác nằm trên cùng một mạng vật lý. Các gói từ A1 cần chuyển cho C1 sẽ được hướng đến gateway G1 và G2. Trạm A1 đầu tiên sẽ truyền các gói đến gateway G1 thông qua mạng A. Sau đó G1 truyền tiếp đến G2 thông qua mạng B và cuối cùng G2 sẽ truyền các gói trực tiếp đến trạm C1, bởi vì chúng được nối trực tiếp với nhau thông qua mạng C. Trạm A1 không hề biết đến các gateway nằm ở sau G1. A1 gửi các gói số liệu cho các mạng B và C đến gateway cục bộ G1 và dựa vào gateway này để định hướng tiếp cho các gói dữ liệu đi đến đích. Theo cách này thì trạm C1 trước tiên sẽ gửi các gói của mình đến cho G2 và G2 sẽ gửi đi tiếp cho các trạm ở trên mạng A cũng như ở trên mạng B.

Hình vẽ sau mô tả việc dùng các gateway để gửi các gói dữ liệu:



Hình 3.13. Định tuyến giữa hai hệ thống

Việc phân mảnh các gói dữ liệu: Trong quá trình truyền dữ liệu, một gói dữ liệu (datagram) có thể được truyền đi thông qua nhiều mạng khác nhau. Một gói dữ liệu (datagram) nhận được từ một mạng nào đó có thể quá lớn để truyền đi trong gói đơn ở trên một mạng khác, bởi mỗi loại cấu trúc mạng cho phép một đơn vị truyền cực đại (Maximum Transmit Unit - MTU), khác nhau. Đây chính là kích thước lớn nhất của một gói mà chúng có thể truyền. Nếu như một gói dữ liệu nhận được từ một mạng nào đó mà lớn hơn MTU của một mạng khác thì nó cần được phân mảnh ra thành các gói nhỏ hơn, gọi là fragment. Quá trình này gọi là quá trình phân mảnh. Dạng của một fragment cũng giống như dạng của một gói dữ liệu thông thường. Từ thứ hai trong phần header chứa các thông tin để xác định mỗi fragment và cung cấp các thông tin để hợp nhất các fragment này lại thành các gói như ban đầu. Trường identification dùng để xác định fragment này là thuộc về gói dữ liệu nào.

I.6. Một số giao thức điều khiển

I.6.1. Giao thức ICMP

ICMP ((Internet Control Message Protocol) là một giao thức điều khiển của mức IP, được dùng để trao đổi các thông tin điều khiển dòng số liệu, thông báo lỗi và các thông tin trạng thái khác của bộ giao thức TCP/IP. Ví dụ:

- Điều khiển lưu lượng dữ liệu (Flow control): khi các gói dữ liệu đến quá nhanh, thiết bị đích hoặc thiết bị định tuyến ở giữa sẽ gửi một thông điệp ICMP trở lại thiết bị gửi, yêu cầu thiết bị gửi tạm thời ngừng việc gửi dữ liệu.

- Thông báo lỗi: trong trường hợp địa chỉ đích không tới được thì hệ thống sẽ gửi một thông báo lỗi "Destination Unreachable".

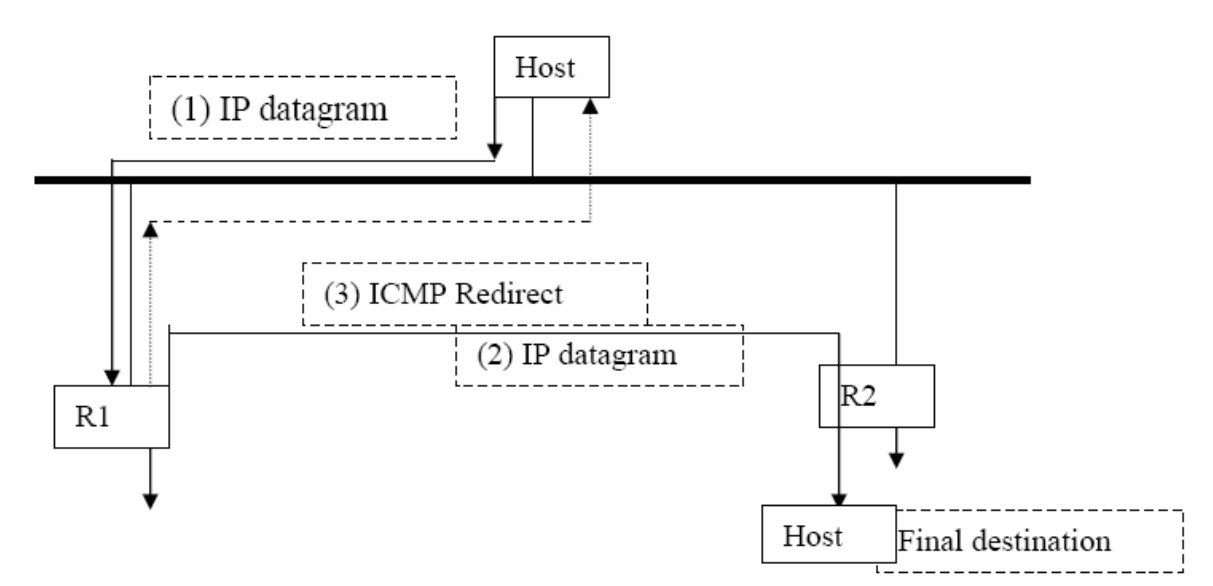
- Định hướng lại các tuyến đường: một thiết bị định tuyến sẽ gửi một thông điệp ICMP "định tuyến lại" (Redirect Router) để thông báo với một trạm là nên dùng thiết bị định tuyến khác để tới thiết bị đích. Thông điệp này có thể chỉ được dùng khi trạm nguồn ở trên cùng một mạng với cả hai thiết bị định tuyến.

- Kiểm tra các trạm ở xa: một trạm có thể gửi một thông điệp ICMP "Echo" để kiểm tra xem một trạm có hoạt động hay không.

Sau đây là mô tả một ứng dụng của giao thức ICMP thực hiện việc định tuyến lại (Redirect):

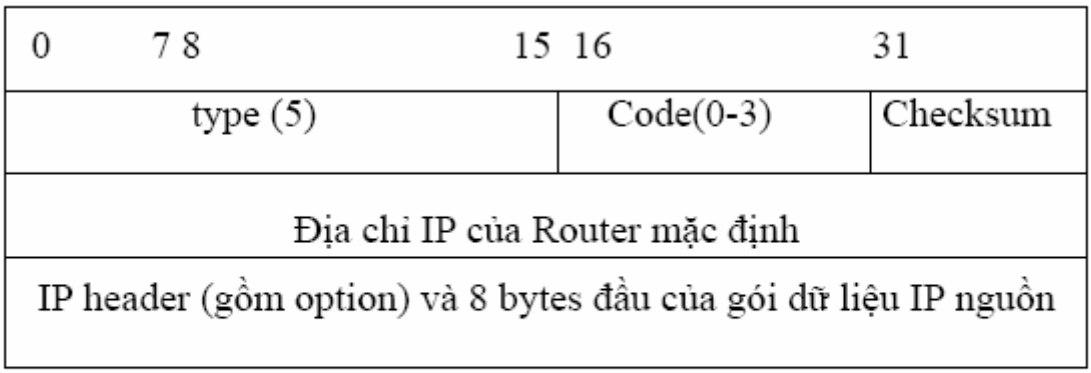
Ví dụ: giả sử host gửi một gói dữ liệu IP tới Router R1. Router R1 thực hiện việc quyết định tuyến vì R1 là router mặc định của host đó. R1 nhận gói dữ liệu và tìm trong bảng định tuyến và nó tìm thấy một tuyến tới R2. Khi R1 gửi gói dữ liệu tới R2 thì R1 phát hiện ra rằng nó đang gửi gói dữ liệu đó ra ngoài trên cùng một giao diện mà gói dữ liệu đó đã đến (là giao diện mạng LAN mà cả host và hai Router nối đến). Lúc này R1 sẽ gửi một thông báo ICMP Redirect Error tới host, thông báo cho host nên gửi các gói dữ liệu tiếp theo đến R2 thì tốt hơn.

Tác dụng của ICMP Redirect là để cho mọt host với nhận biết tối thiểu về định tuyến xây dựng lên một bảng định tuyến tốt hơn theo thời gian. Host đó có thể bắt đầu với một tuyến mặc định (có thể R1 hoặc R2 như ví dụ trên) và bất kỳ lần nào tuyến mặc định này được dùng với host đó đến R2 thì nó sẽ được Router mặc định gửi thông báo Redirect để cho phép host đó cập nhật bảng định tuyến của nó một cách phù hợp hơn. Khuôn dạng của thông điệp ICMP redirect như sau:



Hình 3.8 Thực hiện việc định tuyến lại

Có bốn loại thông báo ICMP redirect khác nhau với các giá trị mã (code) như bảng sau:



Hình 3.9 Dạng thông điệp ICMP redirect

Code	Description
0	Redirect cho mạng
1	Redirect cho host
2	Redirect cho loại dịch vụ (TOS) và mạng
3	Redirect cho loại dịch vụ và host

Các loại định hướng lại của gói dữ liệu ICMP

Redirect chỉ xảy ra khi cả hai Router R1 và R2 cùng nằm trên một mạng với host nhận direct đó.

I.6.2. Giao thức ARP và giao thức RARP

Địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm đó trên một mạng cục bộ (Ethernet, Token Ring,...). Trên một mạng cục bộ hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. Như vậy vấn đề đặt ra là phải thực hiện ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý (48 bits) của một trạm. Giao thức ARP (Address Resolution Protocol) đã được xây dựng để chuyển đổi từ địa chỉ IP sang địa chỉ vật lý khi cần thiết. Ngược lại, giao thức RARP (Reverse Address Resolution Protocol) được dùng để chuyển đổi địa chỉ vật lý sang địa chỉ IP. Các giao thức ARP và RARP không phải là bộ phận của IP mà IP sẽ dùng đến chúng khi cần.

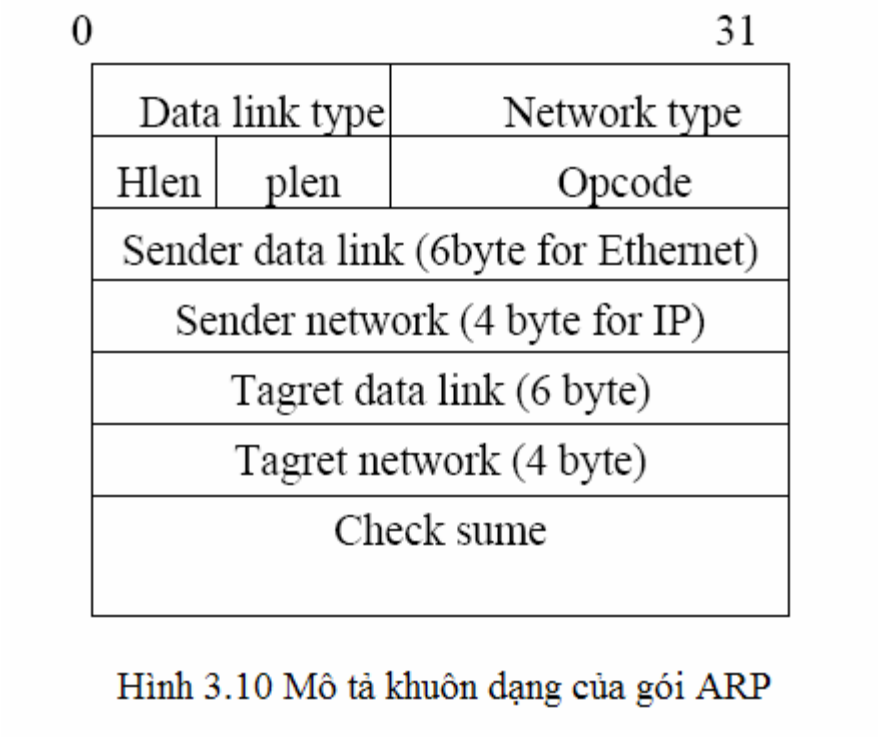
Giao thức ARP

Giao thức TCP/IP sử dụng ARP để tìm địa chỉ vật lý của trạm đích. Ví dụ khi cần gửi một gói dữ liệu IP cho một hệ thống khác trên cùng một mạng vật lý Ethernet, hệ thông gửi cần biết địa chỉ Ethernet của hệ thống đích để tầng liên kết dữ liệu xây dựng khung gói dữ liệu.

Thông thường, mỗi hệ thống lưu giữ và cập nhật bảng thích ứng địa chỉ IP-MAC tại chỗ (còn được gọi là bảng ARP cache). Bảng thích ứng địa chỉ được cập nhật bởi người quản trị hệ thống hoặc tự động bởi giao thức ARP sau mỗi lần ánh xạ được một địa chỉ thích ứng mới. Khuôn dạng của gói dữ liệu ARP được mô tả trong hình

- Data link type: cho biết loại công nghệ mạng mức liên kết (ví dụ đối với mạng Ethernet trường này có giá trị 01).

- Network type: cho biết loại mạng (ví dụ đối với mạng IPv4, trường này có giá



Hình 3.10 Mô tả khuôn dạng của gói ARP

trị 0800_{16}).

- Hlen (hardware length): độ dài địa chỉ mức liên kết (6 byte).

- Plen (Protocol length): cho biết độ dài địa chỉ mạng (4 byte)

- Opcode (operation code): mã lệnh yêu cầu: ; mã lệnh trả lời .

- Sender data link: địa chỉ mức liên kết của thiết bị phát gói dữ liệu này.

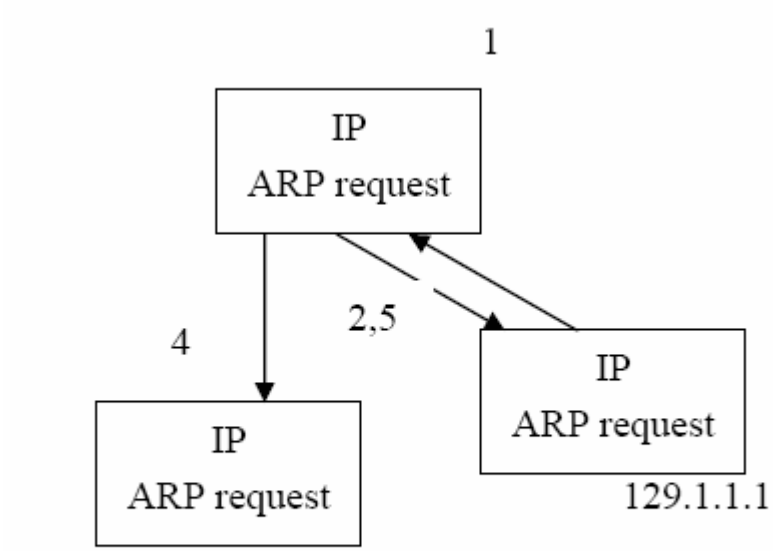
- Sender network : địa chỉ IP của thiết bị phát.

- Tagret data link: trong yêu cầu đây là địa chỉ mức liên kết cần tìm (thông thường được điền 0 bởi thiết bị gửi yêu cầu); trong trả lời đây là địa chỉ mức liên kết của thiết bị gửi yêu cầu.

- Tagret network : trong yêu cầu đây là địa chỉ IP mà địa chỉ mức liên kết tương ứng cần tìm; trong trả lời đây là địa chỉ IP của thiết bị gửi yêu cầu.

Mỗi khi cần tìm thích ứng địa chỉ IP - MAC, có thể tìm địa chỉ MAC tương ứng với địa IP đó trước tiên trong bảng địa chỉ IP - MAC ở mỗi hệ thống. Nếu không tìm thấy, có thể sử dụng giao thức ARP để làm việc này. Trạm làm việc gửi yêu cầu ARP (ARP_Request) tìm thích ứng địa chỉ IP -MAC đến máy phục vụ ARP - server. Máy phục vụ ARP tìm trong bảng thích ứng địa chỉ IP - MAC của mình và trả lời bằng ARP_Response cho trạm làm việc. Nếu không, máy phục vụ chuyển tiếp yêu cầu nhận được dưới dạng quảng bá cho tất cả các trạm làm việc trong mạng. Trạm nào có trùng địa chỉ IP được yêu cầu sẽ trả lời với địa chỉ MAC của mình. Tóm lại tiến trình của ARP được mô tả như sau

1. IP yêu cầu địa chỉ MAC.



Hình 3.11 Tiến trình ARP

2. Tìm kiếm trong bảng ARP.

3. Nếu tìm thấy sẽ trả lại địa chỉ MAC.

4. Nếu không tìm thấy, tạo gói ARP yêu cầu và gửi tới tất cả các trạm.

5. Tuỳ theo gói dữ liệu trả lời, ARP cập nhật vào bảng ARP và gửi địa chỉ MAC đó cho IP.

Giao thức RARP

Reverse ARP (Reverse Address Resolution Protocol) là giao thức giải thích ứng địa chỉ AMC - IP. Quá trình này ngược lại với quá trình giải thích ứng địa chỉ IP - MAC mô tả ở trên, nghĩa là cho trước địa chỉ mức liên kết, tìm địa chỉ IP tương ứng.

II. Giao thức lớp chuyển tải (Transport Layer)

2.1. Giao thức TCP

TCP (Transmission Control Protocol) là một giao thức "có liên kết" (connection - oriented), nghĩa là cần thiết lập liên kết (logic), giữa một cặp thực thể TCP trước khi chúng trao đổi dữ liệu với nhau.

TCP cung cấp khả năng truyền dữ liệu một cách an toàn giữa các máy trạm trong hệ thống các mạng. Nó cung cấp thêm các chức năng nhằm kiểm tra tính chính xác của dữ liệu khi đến và bao gồm cả việc gửi lại dữ liệu khi có lỗi xảy ra. TCP cung cấp các chức năng chính sau:

1. Thiết lập, duy trì, kết thúc liên kết giữa hai quá trình.

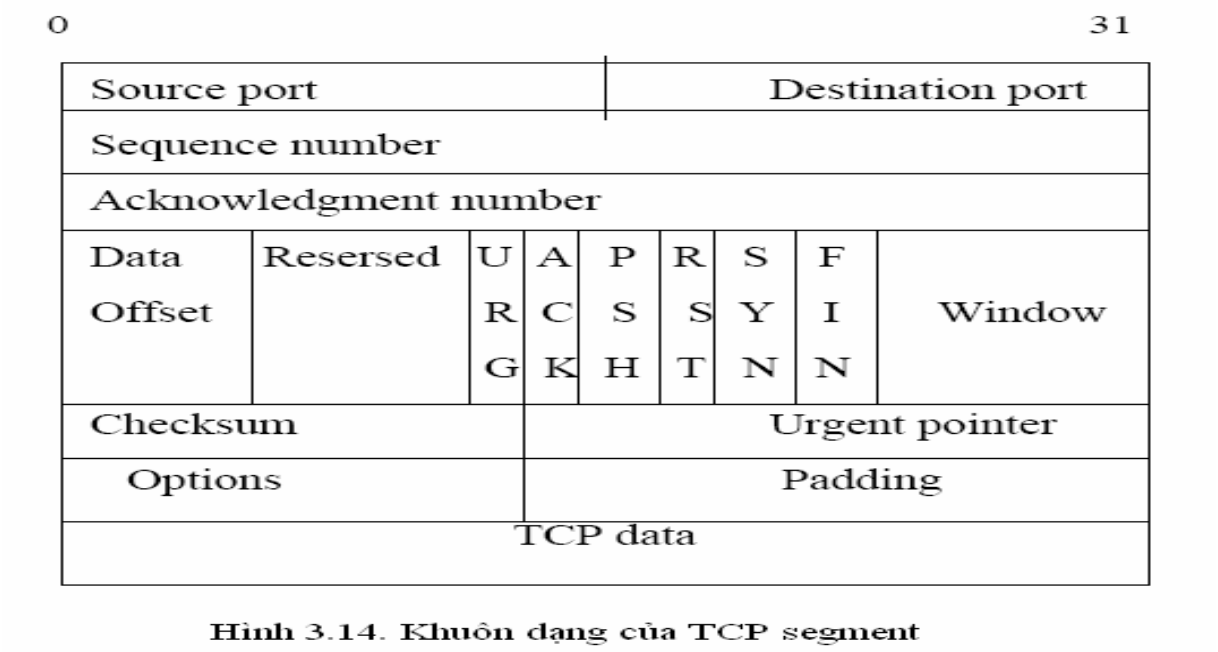
2. Phân phát gói tin một cách tin cậy.

3. Đánh số thứ tự (sequencing) các gói dữ liệu nhằm truyền dữ liệu một cách tin cậy.

4. Cho phép điều khiển lỗi.

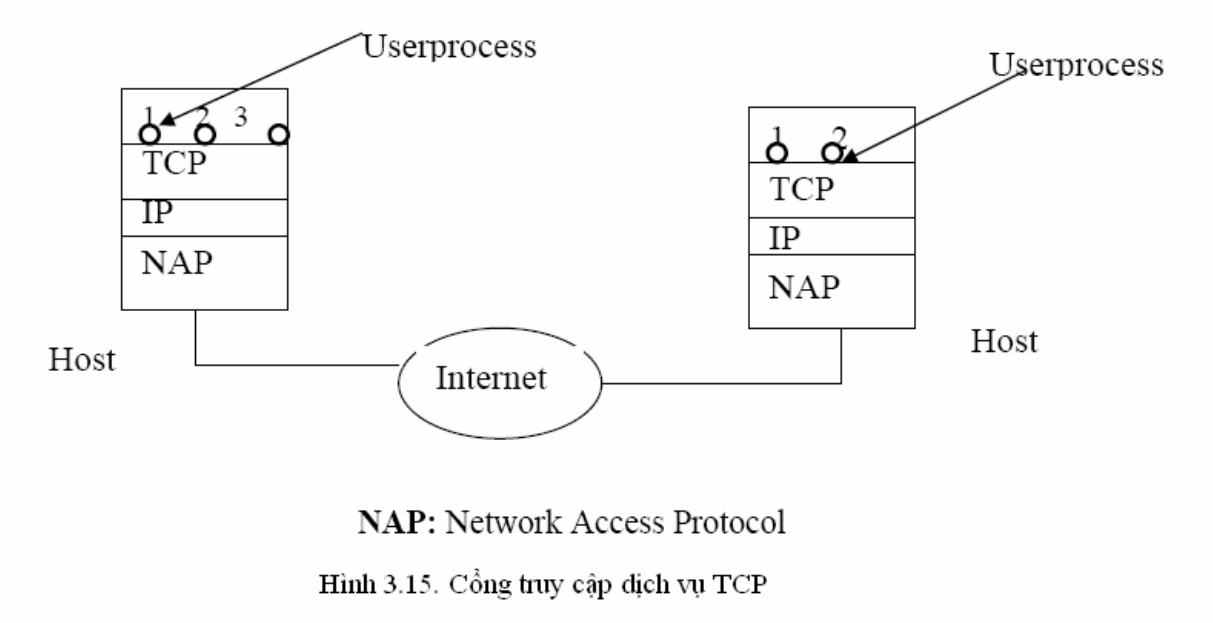
5. Cung cấp khả năng đa kết nối với các quá trình khác nhau giữa trạm nguồn và trạm đích nhất định thông qua việc sử dụng các cổng.

6. Truyền dữ liệu sử dụng cơ chế song công (full-duplex).



Hình 3.14. Khuôn dạng của TCP segment

2.2 Cấu trúc gói dữ liệu TCP



NAP: Network Access Protocol

Hình 3.15. Cổng truy cập dịch vụ TCP

- Source port (16 bits) : số hiệu cổng của trạm nguồn

- Destination port (16 bits) : số hiệu cổng của trạm đích

- Sequence Number (32 bits): số hiệu của byte đầu tiên của segment trừ khi bit SYN được thiết lập. Nếu bit SYN được thiết lập thì Sequence Number là số hiệu tuần tự khởi đầu (ISN) và byte dữ liệu đầu tiên là ISN +1.

- Acknowlegment: vị trí tương đối của byte cuối cùng đã nhận đúng bởi thực thể gửi gói ACK cộng thêm 1. Giá trị của trường này còn được gọi là số tuần tự thu. Trường này được kiểm tra chỉ khi bit ACK=1.

- Data offset (4 bits) : số tượng từ 32 bit trong TCP header. Tham số này chỉ ra vị trí bắt đầu của vùng dữ liệu

- Reserved (6 bits) : dành để dùng trong tương lai. Phải được thiết lập là 0.

- Control bits : các bit điều khiển

  - URG : vùng con trỏ khẩn (Urgent Pointer) có hiệu lực.

  - ACK : vùng báo nhận (ACK number) có hiệu lực.

- PSH : chức năng Push. PSH=1 thực thể nhận phải chuyển dữ liệu này cho ứng dụng tức thời.

  - RST : thiết lập lại (reset) kết nối.

  - SYN : đồng bộ hoá các số hiệu tuần tự, dùng để thiết lập kết nối TCP.

  - FIN : thông báo thực thể gửi đã kết thúc gửi dữ liệu.

- Window (16 bits): cấp phát credit để kiểm soát luồng dữ liệu (cơ chế của sổ). Đây chính là số lượng các byte dữ liệu, bắt đầu từ byte được chỉ ra trong vùng ACK number, mà trạm nguồn đã sẵn sàng để nhận

- Checksum (16 bits) : mã kiểm soát lỗi (theo phương pháp CRC) cho toàn bộ segment (header + data)

- Urgent pointer (16 bits) : con trỏ này trỏ tới số hiệu tuần tự của byte đi theo sau dữ liệu khẩn, cho phép bên nhận biết được độ dài của dữ liệu khẩn. Vùng này chỉ có hiệu lực khi bit URG được thiết lập

- Options (độ dài thay đổi): khai báo các option của TCP, trong đó có độ dài tối đa của vùng TCP data trong một segment

- Padding (độ dài thay đổi) : phần chèn thêm vào header để bảo đảm phần header luôn kết thúc ở một mốc 32 bits. Phần thêm này gồm toàn số 0.

- TCP data (độ dài thay đổi) : chứa dữ liệu của tầng trên, có độ dài tối đa ngầm định là 536 bytes. Giá trị này có thể điều chỉnh bằng cách khai báo trong vùng options.

Một tiến trình ứng dụng trong một host truy nhập vào các dịch vụ của TCP cung cấp thông qua một cổng (port) như sau:

Một cổng kết hợp với một địa chỉ IP tạo thành một socket duy nhất trong liên mạng. TCP được cung cấp nhờ một liên kết logic giữa một cặp socket. Một socket có thể tham gia nhiều liên kết với các socket ở xa khác nhau. Trước khi truyền dữ liệu giữa hai trạm cần phải thiết lập một liên kết TCP giữa chúng và khi kết thúc phiên truyền dữ liệu thì liên kết đó sẽ được giải phóng. Cũng giống như ở các giao thức khác, các thực thể ở tầng trên sử dụng TCP thông qua các hàm dịch vụ nguyên thuỷ (service primitives), hay còn gọi là các lời gọi hàm (function call).

2.3. Thiết lập và kết thúc kết nối TCP

Thiết lập kết nối

Thiết lập kết nối TCP được thực hiện trên cơ sở phương thức bắt tay ba bước (Tree - way Handsake) hình 3.14. Yêu cầu kết nối luôn được tiến trình trạm khởi tạo, bằng cách gửi một gói TCP với cờ SYN=1 và chứa giá trị khởi tạo số tuần tự ISN của client. Giá trị ISN này là một số 4 byte không dấu và được tăng mỗi khi kết nối được yêu cầu (giá trị này quay về 0 khi nó tới giá trị 2^{32}). Trong thông điệp SYN này còn chứa số hiệu cổng TCP của phần mềm dịch vụ mà tiến trình trạm muốn kết nối (bước 1).

Mỗi thực thể kết nối TCP đều có một giá trị ISN mới số này được tăng theo thời gian. Vì một kết nối TCP có cùng số hiệu cổng và cùng địa chỉ IP được dùng lại nhiều lần, do đó việc thay đổi giá trị INS ngăn không cho các kết nối dùng lại các dữ liệu đã cũ (stale) vẫn còn được truyền từ một kết nối cũ và có cùng một địa chỉ kết nối.

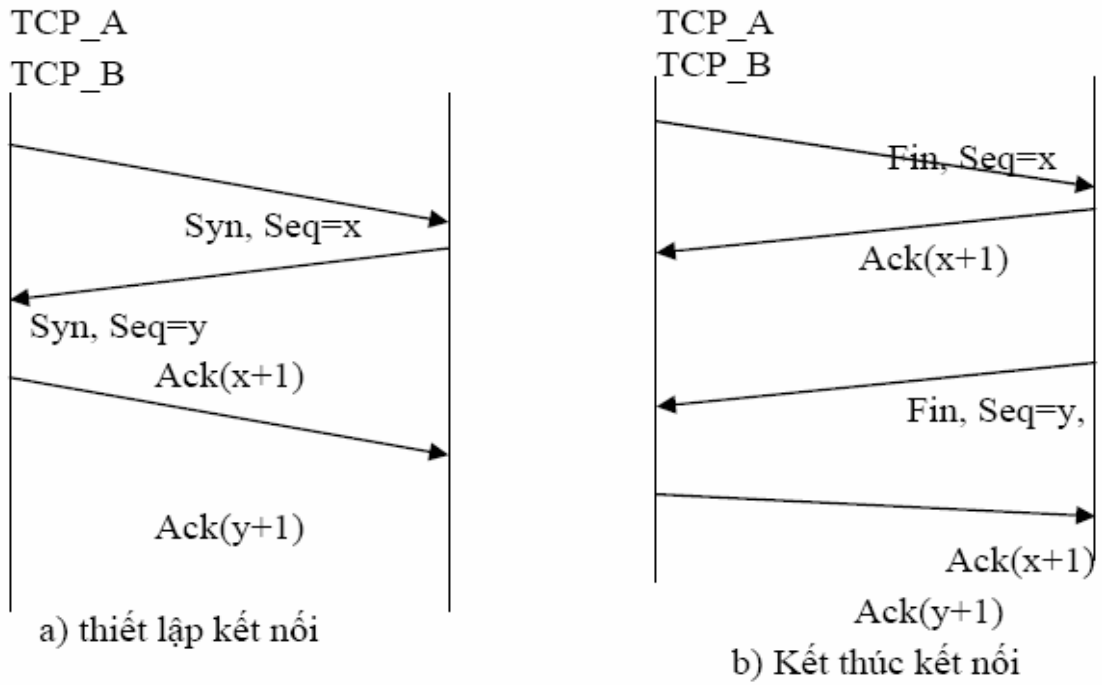
Khi thực thể TCP của phần mềm dịch vụ nhận được thông điệp SYN, nó gửi lại gói SYN cùng giá trị ISN của nó và đặt cờ ACK=1 trong trường hợp sẵn sàng nhận kết nối. Thông điệp này còn chứa giá trị ISN của tiến trình trạm trong

trường hợp số tuần tự thu để báo rằng thực thể dịch vụ đã nhận được giá trị ISN của tiến trình trạm (bước 2).

Tiến trình trạm trả lời lại gói SYN của thực thể dịch vụ bằng một thông báo trả lời ACK cuối cùng. Bằng cách này, các thực thể TCP trao đổi một cách tin cậy các giá trị ISN của nhau và có thể bắt đầu trao đổi dữ liệu. Không có thông điệp nào trong ba bước trên chứa bất kỳ dữ liệu gì; tất cả thông tin trao đổi đều nằm trong phần tiêu đề của thông điệp TCP (bước 3).

Kết thúc kết nối

Khi có nhu cầu kết thúc kết nối, thực thể TCP, ví dụ cụ thể A gửi yêu cầu kết thúc kết nối với FIN=1. Vì kết nối TCP là song công (full-duplex) nên mặc dù nhận được yêu cầu kết thúc kết nối của A (A thông báo hết số liệu gửi) thực thể B vẫn có thể tiếp tục truyền số liệu cho đến khi B không còn số liệu để gửi và thông báo cho A bằng yêu cầu kết thúc kết nối với FIN=1 của mình. Khi thực thể TCP đã nhận được thông điệp FIN và sau khi đã gửi thông điệp FIN của chính mình, kết nối TCP thực sụ kết thúc.



Hình 3.16. Quá trình kết nối theo ba bước

Chương 1: Tổng quan về công nghệ mạng máy tính và mạng cục bộ

Chương này cung cấp các khái niệm, các kiến thức cơ bản nhất về mạng máy tính và phân loại mạng máy tính. Các nội dung giới thiệu mang tính tổng quan về mạng cục

bộ, kiến trúc mạng cục bộ, phương pháp truy cập trong mạng cục bộ và các chuẩn vật lý về các thiết bị mạng. Đây là những kiến thức cơ bản rất hữu ích do phạm vi sử dụng của mạng cục bộ là đang phổ biến hiện nay. Hầu hết các cơ quan, tổ chức, công ty có sử dụng công nghệ thông tin đều thiết lập mạng cục bộ riêng.

Các khái niệm, nội dung cơ bản trong chương 1 cần phải nắm vững đối với tất cả các học viên vì chúng sẽ được sử dụng nhiều trong các chương tiếp theo.

Mục 1: Mạng máy tính

I. Lịch sử mạng máy tính

Internet bắt nguồn từ đề án ARPANET (Advanced Research Project Agency Network) khởi sự trong năm 1969 bởi Bộ Quốc phòng Mỹ (American Department of Defense). Đề án ARPANET với sự tham gia của một số trung tâm nghiên cứu, đại học tại Mỹ (UCLA, Stanford, . . .) nhằm mục đích thiết kế một mạng WAN (Wide Area Network) có khả năng tự bảo tồn chống lại sự phá hoại một phân mạng bằng chiến tranh nguyên tử. Đề án này dẫn tới sự ra đời của nghi thức truyền IP (Internet Protocol). Theo nghi thức này, thông tin truyền sẽ được đóng thành các gói dữ liệu và truyền trên mạng theo nhiều đường khác nhau từ người gửi tới nơi người nhận. Một hệ thống máy tính nối trên mạng gọi là **Router** làm nhiệm vụ tìm đường đi tối ưu cho các gói dữ liệu, tất cả các máy tính trên mạng đều tham dự vào việc truyền dữ liệu, nhờ vậy nếu một phân mạng bị phá huỷ các **Router** có thể tìm đường khác để truyền thông tin tới người nhận. Mạng ARPANET được phát triển và sử dụng trước hết trong các trường đại học, các cơ quan nhà nước Mỹ, tiếp theo đó, các trung tâm tính toán lớn, các trung tâm truyền vô tuyến điện và vệ tinh được nối vào mạng, . . . trên cơ sở này, ARPANET được nối với khắp các vùng trên thế giới.

Tới năm 1983, trước sự thành công của việc triển khai mạng ARPANET, Bộ quốc phòng Mỹ tách một phân mạng giành riêng cho quân đội Mỹ (MILNET). Phần còn lại, gọi là NSFnet, được quản lý bởi NSF (National Science Foundation) NSF dùng 5 siêu máy tính để làm **Router** cho mạng, và lập một tổ chức không chính phủ để quản lý mạng, chủ yếu dùng cho đại học và nghiên cứu cơ bản trên toàn thế giới. Tới năm 1987, NSFnet mở cửa cho cá nhân và cho các công ty tư nhân (BITnet), tới năm 1988 siêu mạng được mang tên INTERNET.

Tuy nhiên cho tới năm 1988, việc sử dụng INTERNET còn hạn chế trong các dịch vụ truyền mạng (FTP), thư điện tử(E-mail), truy nhập từ xa (TELNET) không thích ứng với nhu cầu kinh tế và đời sống hàng ngày. INTERNET chủ yếu được dùng trong môi trường nghiên cứu khoa học và giảng dạy đại học. Trong năm 1988, tại trung tâm nghiên cứu nguyên tử của Pháp CERN (Centre Européen de Recherche Nuclaire) ra đời đề án Mạng nhện thế giới WWW (World Wide Web). Đề án này,

nhằm xây dựng một phương thức mới sử dụng INTERNET, gọi là phương thức Siêu văn bản (HyperText). Các tài liệu và hình ảnh được trình bày bằng ngôn ngữ HTML (HyperText Markup Language) và được phát hành trên INTERNET qua các hệ chủ làm việc với nghi thức HTTP (HyperText Transport Protocol). Từ năm 1992, phương thức làm việc này được đưa ra thử nghiệm trên INTERNET, rất nhanh chóng, các công ty tư nhân tìm thấy qua phương thức này cách sử dụng INTERNET trong kinh tế và đời sống. Vốn đầu tư vào INTERNET được nhân lên hàng chục lần. Từ năm 1994 INTERNET trở thành siêu mạng kinh doanh. Số các công ty sử dụng INTERNET vào việc kinh doanh và quảng cáo lên gấp hàng nghìn lần kể từ năm 1995. Doanh số giao dịch thương mại qua mạng INTERNET lên hàng chục tỉ USD trong năm 1996 . . .

Với phương thức siêu văn bản, người sử dụng, qua một phần mềm truy đọc (Navigator, Browser), có thể tìm đọc tất cả các tài liệu siêu văn bản công bố tại mọi nơi trên thế giới (kể cả hình ảnh và tiếng nói). Với công nghệ WWW, chúng ta bước vào giai đoạn mà mọi thông tin có thể có ngay trên bàn làm việc của mình. Mỗi công ty hoặc người sử dụng, được phân phối một trang cội nguồn (Home Page) trên hệ chủ HTTP. Trang cội nguồn, là siêu văn bản gốc, để tự do có thể tìm tới tất cả các siêu văn bản khác mà người sử dụng muốn phát hành. Địa chỉ của trang cội nguồn được tìm thấy từ khắp mọi nơi trên thế giới. Vì vậy, đối với một xí nghiệp, trang cội nguồn trở thành một văn phòng đại diện điện tử trên INTERNET. Từ khắp mọi nơi, khách hàng có thể xem các quảng cáo và liên hệ trực tiếp với xí nghiệp qua các dòng siêu liên (HyperLink) trong siêu văn bản.

Tới năm 1994, một điểm yếu của INTERNET là không có khả năng lập trình cục bộ, vì các máy nối vào mạng không đồng bộ và không tương thích. Thiếu khả năng này, INTERNET chỉ được dùng trong việc phát hành và truyền thông tin chứ không dùng để xử lý thông tin được. Trong năm 1994, hãng máy tính SUN Corporation công bố một ngôn ngữ mới, gọi là JAVA (cafe), cho phép lập trình cục bộ trên INTERNET, các chương trình JAVA được gọi thẳng từ các siêu văn bản qua các siêu liên (Applet). Vào mùa thu năm 1995, ngôn ngữ JAVA chính thức ra đời, đánh dấu một bước tiến quan trọng trong việc sử dụng INTERNET. Trước hết, một chương trình JAVA, sẽ được chạy trên máy khách (Workstation) chứ không phải trên máy chủ (Server). Điều này cho phép sử dụng công suất của tất cả các máy khách vào việc xử lý số liệu. Hàng triệu máy tính (hoặc vi tính) có thể thực hiện cùng một lúc một chương trình ghi trên một siêu văn bản trong máy chủ. Việc lập trình trên INTERNET cho phép truy nhập từ một trang siêu văn bản vào các chương trình xử lý thông tin, đặc biệt là các chương trình điều hành và quản lý thông tin của một xí nghiệp. phương thức làm việc này, được gọi là INTRANET. Chỉ trong năm 1995-1996, hàng trăm nghìn dịch vụ phần mềm INTRANET được phát triển. Nhiều hãng máy tính và phần mềm như Microsoft, SUN, IBM, Oracle, Netscape,... đã phát triển và kinh doanh hàng

loạt phần mềm hệ thống và phần mềm cơ bản để phát triển các ứng dụng INTERNET / INTRANET.

II. Giới thiệu mạng máy tính

I.1. I.Định nghĩa mạng máy tính và mục đích của việc kết nối mạng

I.1.1. Nhu cầu của việc kết nối mạng máy tính

Việc nối máy tính thành mạng từ lâu đã trở thành một nhu cầu khách quan vì :

- Có rất nhiều công việc về bản chất là phân tán hoặc về thông tin, hoặc về xử lý hoặc cả hai đòi hỏi có sự kết hợp truyền thông với xử lý hoặc sử dụng phương tiện từ xa.

- Chia sẻ các tài nguyên trên mạng cho nhiều người sử dụng tại một thời điểm (ổ cứng, máy in, ổ CD ROM . . .)

- Nhu cầu liên lạc, trao đổi thông tin nhờ phương tiện máy tính.

- Các ứng dụng phần mềm đòi hòi tại một thời điểm cần có nhiều người sử dụng, truy cập vào cùng một cơ sở dữ liệu.

I.1.2. Định nghĩa mạng máy tính

Nói một cách ngắn gọn thì mạng máy tính là tập hợp các máy tính độc lập (autonomous) được kết nối với nhau thông qua các đường truyền vật lý và tuân theo các quy ước truyền thông nào đó.

Khái niệm máy tính độc lập được hiểu là các máy tính không có máy nào có khả năng khởi động hoặc đình chỉ một máy khác.

Các đường truyền vật lý được hiểu là các môi trường truyền tín hiệu vật lý (có thể là hữu tuyến hoặc vô tuyến).

Các quy ước truyền thông chính là cơ sở để các máy tính có thể "nói chuyện" được với nhau và là một yếu tố quan trọng hàng đầu khi nói về công nghệ mạng máy tính.

I.2. Đặc trưng kỹ thuật của mạng máy tính

Một mạng máy tính có các đặc trưng kỹ thuật cơ bản như sau:

I.2.1. Đường truyền

Là thành tố quan trọng của một mạng máy tính, là phương tiện dùng để truyền các tín hiệu điện tử giữa các máy tính. Các tín hiệu điệu tử đó chính là các thông tin, dữ liệu được biểu thị dưới dạng các xung nhị phân (ON_OFF), mọi tín hiệu truyền giữa các máy tính với nhau đều thuộc sóng điện từ, tuỳ theo tần số mà ta có thể dùng các đường truyền vật lý khác nhau

Đặc trưng cơ bản của đường truyền là giải thông nó biểu thị khả năng truyền tải tín hiệu của đường truyền.

Thông thuường người ta hay phân loại đường truyền theo hai loại:

- Đường truyền hữu tuyến (các máy tính được nối với nhau bằng các dây cáp mạng).

- Đường truyền vô tuyến: các máy tính truyền tín hiệu với nhau thông qua các sóng vô tuyền với các thiết bị điều chế/giải điều chế ở các đầu mút.

I.2.2. Kỹ thuật chuyển mạch:

Là đặc trưng kỹ thuật chuyển tín hiệu giữa các nút trong mạng, các nút mạng có chức năng hướng thông tin tới đích nào đó trong mạng, hiện tại có các kỹ thuật chuyển mạch như sau:

- Kỹ thuật chuyển mạch kênh: Khi có hai thực thể cần truyền thông với nhau thì giữa chúng sẽ thiết lập một kênh cố định và duy trì kết nối đó cho tới khi hai bên ngắt liên lạc. Các dữ liệu chỉ truyền đi theo con đường cố định đó.

- Kỹ thuật chuyển mạch thông báo: thông báo là một đơn vị dữ liệu của người sử dụng có khuôn dạng được quy định trước. Mỗi thông báo có chứa các thông tin điều khiển trong đó chỉ rõ đích cần truyền tới của thông báo. Căn cứ vào thông tin điều khiển này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp trên con đường dẫn tới đích của thông báo

- Kỹ thuật chuyển mạch gói: ở đây mỗi thông báo được chia ra thành nhiều gói nhỏ hơn được gọi là các gói tin (packet) có khuôn dạng qui định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của cùng một thông báo có thể được gởi đi qua mạng tới đích theo nhiều con đường khác nhau.

I.2.3. Kiến trúc mạng

Kiến trúc mạng máy tính (network architecture) thể hiện cách nối các máy tính với nhau và tập hợp các quy tắc, quy ước mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để đảm bảo cho mạng hoạt động tốt.

Khi nói đến kiến trúc của mạng người ta muốn nói tới hai vấn đề là hình trạng mạng (Network topology) và giao thức mạng (Network protocol)

- Network Topology: Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng. Các hình trạng mạng cơ bản đó là: hình sao, hình bus, hình vòng

- Network Protocol: Tập hợp các quy ước truyền thông giữa các thực thể truyền thông mà ta gọi là giao thức (hay nghi thức) của mạng. Các giao thức thường gặp nhất là : TCP/IP, NETBIOS, IPX/SPX, . . .

I.2.4. Hệ điều hành mạng

Hệ điều hành mạng là một phần mềm hệ thống có các chức năng sau:

- Quản lý tài nguyên của hệ thống, các tài nguyên này gồm:

+ Tài nguyên thông tin (về phương diện lưu trữ) hay nói một cách đơn giản là quản lý tệp. Các công việc về lưu trữ tệp, tìm kiếm, xoá, copy, nhóm, đặt các thuộc tính đều thuộc nhóm công việc này

+ Tài nguyên thiết bị. Điều phối việc sử dụng CPU, các ngoại vi... để tối ưu hoá việc sử dụng

- Quản lý người dùng và các công việc trên hệ thống.

Hệ điều hành đảm bảo giao tiếp giữa người sử dụng, chương trình ứng dụng với thiết bị của hệ thống.

- Cung cấp các tiện ích cho việc khai thác hệ thống thuận lợi (ví dụ FORMAT đĩa, sao chép tệp và thư mục, in ấn chung ...)

Các hệ điều hành mạng thông dụng nhất hiện nay là: WindowsNT, Windows9X, Windows 2000, Unix, Novell.

I.3. Phân loại mạng máy tính

Có nhiều cách phân loại mạng khác nhau tuỳ thuộc vào yếu tố chính được chọn dùng để làm chỉ tiêu phân loại, thông thường người ta phân loại mạng theo các tiêu chí như sau

- Khoảng cách địa lý của mạng

- Kỹ thuật chuyển mạch mà mạng áp dụng

- Kiến trúc mạng

- Hệ điều hành mạng sử dụng ...

Tuy nhiên trong thực tế người ta thường chỉ phân loại theo hai tiêu chí đầu tiên

I.3.1. Phân loại mạng theo khoảng cách địa lý :

Nếu lấy khoảng cách địa lý làm yếu tố phân loại mạng thì ta có mạng cục bộ, mạng đô thị, mạng diện rộng, mạng toàn cầu.

Mạng cục bộ (LAN - Local Area Network) : là mạng được cài đặt trong phạm vi tương đối nhỏ hẹp như trong một toà nhà, một xí nghiệp...với khoảng cách lớn nhất giữa các máy tính trên mạng trong vòng vài km trở lại.

Mạng đô thị (MAN - Metropolitan Area Network) : là mạng được cài đặt trong phạm vi một đô thị, một trung tâm văn hoá xã hội, có bán kính tối đa khoảng 100 km trở lại.

Mạng diện rộng (WAN - Wide Area Network) : là mạng có diện tích bao phủ rộng lớn, phạm vi của mạng có thể vượt biên giới quốc gia thậm chí cả lục địa.

Mạng toàn cầu (GAN - Global Area Network) : là mạng có phạm vi trải rộng toàn cầu.

I.3.2. Phân loại theo kỹ thuật chuyển mạch:

Nếu lấy kỹ thuật chuyển mạch làm yếu tố chính để phân loại sẽ có: mạng chuyển mạch kênh, mạng chuyển mạch thông báo và mạng chuyển mạch gói.

Mạch chuyển mạch kênh (circuit switched network) : Khi có hai thực thể cần truyền thông với nhau thì giữa chúng sẽ thiết lập một kênh cố định và duy trì kết nối đó cho tới khi hai bên ngắt liên lạc. Các dữ liệu chỉ truyền đi theo con đường cố định đó. Nhược điểm của chuyển mạch kênh là tiêu tốn thời gian để thiết lập kênh truyền cố định và hiệu suất sử dụng mạng không cao.

Mạng chuyển mạch thông báo (message switched network) : Thông báo là một đơn vị dữ liệu của người sử dụng có khuôn dạng được quy định trước. Mỗi thông báo có chứa các thông tin điều khiển trong đó chỉ rõ đích cần truyền tới của thông báo. Căn cứ vào thông tin điều khiển này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp trên con đường dẫn tới đích của thông báo. Như vậy mỗi nút cần phải lưu giữ tạm thời để đọc thông tin điều khiển trên thông báo, nếu thấy thông báo không gửi cho mình thì tiếp tục chuyển tiếp thông báo đi. Tuỳ vào điều kiện của mạng mà thông báo có thể được chuyển đi theo nhiều con đường khác nhau.

Ưu điểm của phương pháp này là :

- Hiệu suất sử dụng đường truyền cao vì không bị chiếm dụng độc quyền mà được phân chia giữa nhiều thực thể truyền thông.

- Mỗi nút mạng có thể lưu trữ thông tin tạm thời sau đó mới chuyển thông báo đi, do đó có thể điều chỉnh để làm giảm tình trạng tắc nghẽn trên mạng.

- Có thể điều khiển việc truyền tin bằng cách sắp xếp độ ưu tiên cho các thông báo.

- Có thể tăng hiệu suất sử dụng giải thông của mạng bằng cách gắn địa chỉ quảng bá (broadcast addressing) để gửi thông báo đồng thời tới nhiều đích.

Nhược điểm của phương pháp này là:

- Không hạn chế được kích thước của thông báo dẫn đến phí tổn lưu giữ tạm thời cao và ảnh hưởng đến thời gian trả lời yêu cầu của các trạm .

Mạng chuyển mạch gói (packet switched network) : ở đây mỗi thông báo được chia ra thành nhiều gói nhỏ hơn được gọi là các gói tin (packet) có khuôn dạng qui định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của cùng một thông báo có thể được gởi đi qua mạng tới đích theo nhiều con đường khác nhau.

Phương pháp chuyển mạch thông báo và chuyển mạch gói là gần giống nhau. Điểm khác biệt là các gói tin được giới hạn kích thước tối đa sao cho các nút mạng (các nút chuyển mạch) có thể xử lý toàn bộ gói tin trong bộ nhớ mà không phải lưu giữ tạm thời trên đĩa. Bởi vậy nên mạng chuyển mạch gói truyền dữ liệu hiệu quả hơn so với mạng chuyển mạch thông báo.

Tích hợp hai kỹ thuật chuyển mạch kênh và chuyển mạch gói vào trong một mạng thống nhất được mạng tích hợp số ISDN (Integated Services Digital Network).

I.3.3. Phân loại theo kiến trúc mạng sử dụng

Kiến trúc của mạng bao gồm hai vấn đề: hình trạng mạng (Network topology) và giao thức mạng (Network protocol)

Hình trạng mạng: Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng

Giao thức mạng: Tập hợp các quy ước truyền thông giữa các thực thể truyền thông mà ta gọi là giao thức (hay nghi thức) của mạng

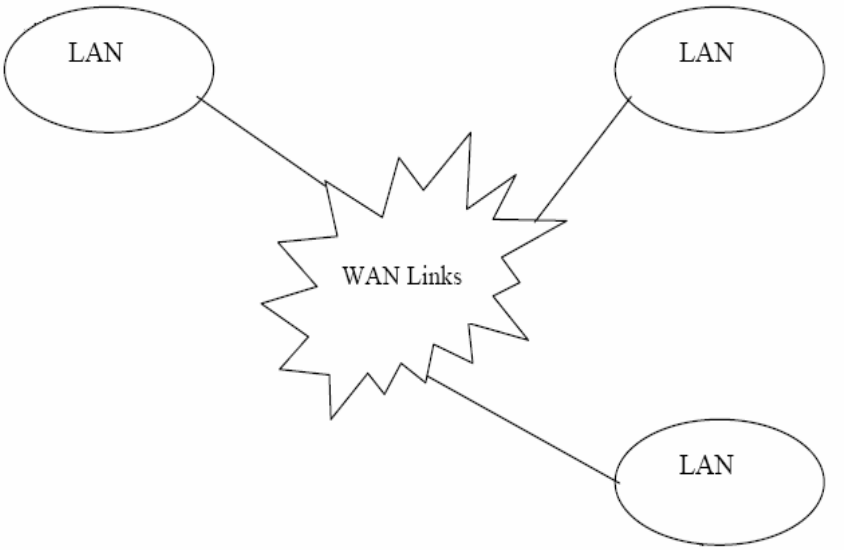
Khi phân loại theo topo mạng người ta thường có phân loại thành: mạng hình sao, tròn, tuyến tính

Phân loại theo giao thức mà mạng sử dụng người ta phân loại thành mạng : TCP/IP, mạng NETBIOS . ..

Tuy nhiên cách phân loại trên không phổ biến và chỉ áp dụng cho các mạng cục bộ.

I.3.4. Phân loại theo hệ điều hàng mạng

Nếu phân loại theo hệ điều hành mạng người ta chia ra theo mô hình mạng ngang hàng, mạng khách/chủ hoặc phân loại theo tên hệ điều hành mà mạng sử dụng: Windows NT, Unix, Novell . . .



I.4. Giới thiệu các mạng máy tính thông dụng nhất

I.4.1. Mạng cục bộ

Một mạng cục bộ là sự kết nối một nhóm máy tính và các thiết bị kết nối mạng được lắp đặt trên một phạm vị địa lý giới hạn, thường trong một toà nhà hoặc một khu công sở nào đó.

Mạng cục bộ có các đặc tính sau:

- Tốc độ truyền dữ liệu cao

- Phạm vi địa lý giới hạn

-Sở hữu của một cơ quan/tổ chức

I.4.2. Mạng diện rộng với kết nối LAN TO LAN

Mạng diện rộng bao giờ cũng là sự kết nối của các mạng LAN, mạng diện rộng có thể trải trên phạm vi một vùng, quốc gia hoặc cả một lục địa thậm chí trên phạm vi toàn cầu.

- Tốc độ truyền dữ liệu không cao

- Phạm vi địa lý không giới hạn

- Thường triển khai dựa vào các công ty truyền thông, bưu điện và dùng các hệ thống truyền thông này để tạo dựng đường truyền

- Một mạng WAN có thể là sở hữu của một tập đoàn/tổ chức hoặc là mạng kết nối của nhiều tập đoàn/tỗ chức 16

I.4.3. Liên mạng INTERNET

Với sự phát triển nhanh chóng của công nghệ là sự ra đời của liên mạng INTERNET,

- Là một mạng toàn cầu

- Là sự kết hợp của vô số các hệ thống truyền thông, máy chủ cung cấp thông tin và dịch vụ, các máy trạm khai thác thông tin

- Dựa trên nhiều nền tảng truyền thông khác nhau, nhưng đều trên nền giao thức TCP/IP

- Là sở hữu chung của toàn nhân loại

- Càng ngày càng phát triển mãnh liệt

I.4.4. Mạng INTRANET

Thực sự là một mạng INTERNET thu nhỏ vào trong một cơ quan/công ty/tổ chức hay một bộ/nghành . . ., giới hạn phạm vi người sử dụng, có sử dụng các công nghệ kiểm soát truy cập và bảo mật thông tin .

Được phát triển từ các mạng LAN, WAN dùng công nghệ INTERNET

II. Mạng cục bộ, kiến trúc mạng cục bộ

II.1. Mạng cục bộ

Tên gọi "mạng cục bộ" được xem xét từ quy mô của mạng. Tuy nhiên, đó không phải là đặc tính duy nhất của mạng cục bộ nhưng trên thực tế, quy mô của mạng quyết định nhiều đặc tính và công nghệ của mạng. Sau đây là một số đặc điểm của mạng cục bộ:

Đặc điểm của mạng cục bộ

- Mạng cục bộ có quy mô nhỏ, thường là bán kính dưới vài km. Đặc điểm này cho phép không cần dùng các thiết bị dẫn đường với các mối liên hệ phức tạp

- Mạng cục bộ thường là sở hữu của một tổ chức. Điều này dường như có vẻ ít quan trọng nhưng trên thực tế đó là điều khá quan trọng để việc quản lý mạng có hiệu quả.

- Mạng cục bộ có tốc độ cao và ít lỗi. Trên mạng rộng tốc độ nói chung chỉ đạt vài Kbit/s. Còn tốc độ thông thường trên mạng cục bộ là 10, 100 Kb/s và tới nay với Gigabit Ethernet, tốc độ trên mạng cục bộ có thể đạt 1Gb/s. Xác suất lỗi rất thấp.

II.2. Kiến trúc mạng cục bộ

II.2.1. Đồ hình mạng (Network Topology)

*** Định nghĩa Topo mạng:**

Cách kết nối các máy tính với nhau về mặt hình học mà ta gọi là tô pô của mạng

Có hai kiểu nối mạng chủ yếu đó là :

- Nối kiểu điểm - điểm (point - to - point).

- Nối kiểu điểm - nhiều điểm (point - to - multipoint hay broadcast).

Theo kiểu điểm - điểm, các đường truyền nối từng cặp nút với nhau và mỗi nút đều có trách nhiệm lưu giữ tạm thời sau đó chuyển tiếp dữ liệu đi cho tới đích. Do cách làm việc như vậy nên mạng kiểu này còn được gọi là mạng "lưu và chuyển tiếp" (store and forward).

Theo kiểu điểm - nhiều điểm, tất cả các nút phân chia nhau một đường truyền vật lý chung. Dữ liệu gửi đi từ một nút nào đó sẽ được tiếp nhận bởi tất cả các nút còn lại trên mạng, bởi vậy cần chỉ ra địa chỉ đích của dữ liệu để căn cứ vào đó các nút kiểm tra xem dữ liệu đó có phải gửi cho mình không.

*** Phân biệt kiểu tô pô của mạng cục bộ và kiểu tô pô của mạng rộng.**

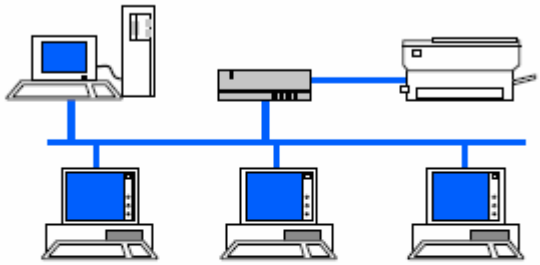
Tô pô của mạng rộng thông thường là nói đến sự liên kết giữa các mạng cục bộ thông qua các bộ dẫn đường (router). Đối với mạng rộng topo của mạng là hình trạng hình học của các bộ dẫn đường và các kênh viễn thông còn khi nói tới tô pô của mạng cục bộ người ta nói đến sự liên kết của chính các máy tính.

a) Mạng hình sao

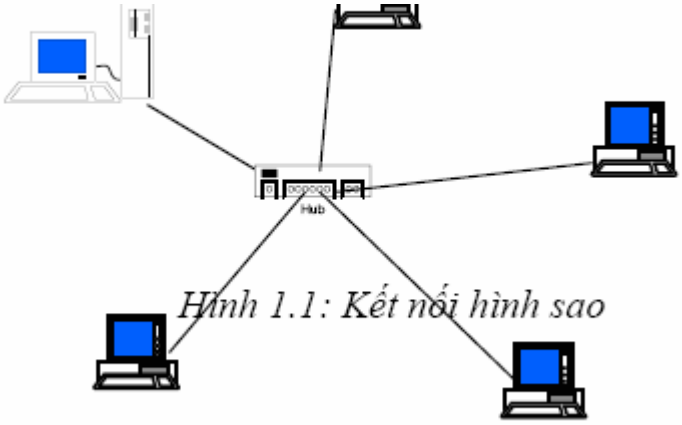
Mạng hình sao có tất cả các trạm được kết nối với một thiết bị trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển đến trạm đích. Tuỳ theo yêu cầu truyền thông trên mạng mà thiết bị trung tâm có thể là bộ chuyển mạch (switch), bộ chọn đường (router) hoặc là bộ phân kênh (hub). Vai trò của thiết bị trung tâm này là thực hiện việc thiết lập các liên kết điểm-điểm (point-to-point) giữa các trạm.

Ưu điểm: Thiết lập mạng đơn giản, dễ dàng cấu hình lại mạng (thêm, bớt các trạm), dễ dàng kiểm soát và khắc phục sự cố, tận dụng được tối đa tốc độ truyền của đường truyền vật lý.

Nhược điểm: Độ dài đường truyền nối một trạm với thiết bị trung tâm bị hạn chế (trong vòng 100m, với công nghệ hiện nay). Hub



Hình 1.2. Kết nối kiểu bus



Hình 1.1: Kết nối hình sao

b) Mạng trục tuyến tính (Bus):

Trong mạng trục tất cả các trạm phân chia một đường truyền chung (bus). Đường truyền chính được giới hạn hai đầu bằng hai đầu nối đặc biệt gọi là terminator. Mỗi trạm được nối với trục chính qua một đầu nối chữ T (T-connector) hoặc một thiết bị thu phát (transceiver).

Khi một trạm truyền dữ liệu tín hiệu được quảng bá trên cả hai chiều của bus, tức là mọi trạm còn lại đều có thể thu được tín hiệu đó trực tiếp. Đối với các bus một chiều thì tín hiệu chỉ đi về một phía, lúc đó các terminator phải được thiết kế sao cho các tín hiệu đó phải được dội lại trên bus để cho các trạm trên mạng đều có thể thu nhận được tín hiệu đó. Như vậy với topo mạng trục dữ liệu được truyền theo các liên kết điểm-đa điểm (point-to-multipoint) hay quảng bá (broadcast).

Ưu điểm : Dễ thiết kế, chi phí thấp

Nhược điểm: Tính ổn định kém, chỉ một nút mạng hỏng là toàn bộ mạng bị ngừng hoạt động

c) Mạng hình vòng

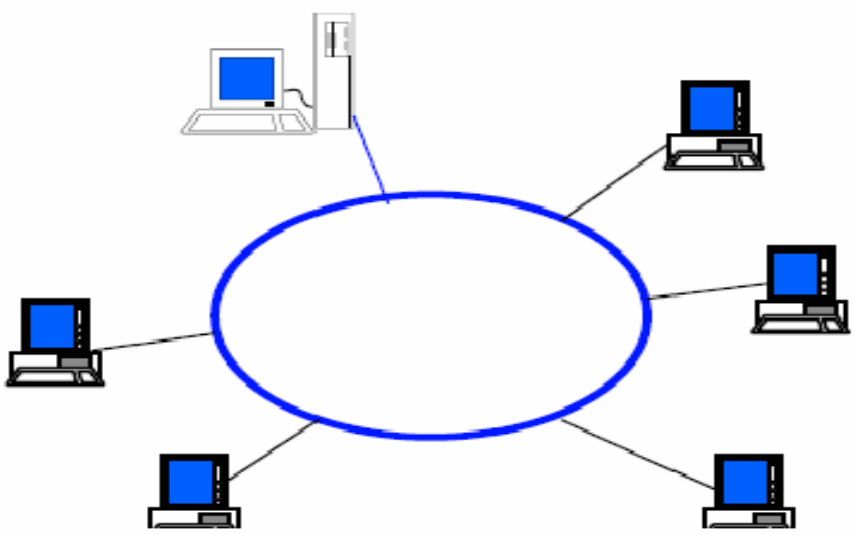
Trên mạng hình vòng tín hiệu được truyền đi trên vòng theo một chiều duy nhất. Mỗi trạm của mạng được nối với vòng qua một bộ chuyển tiếp (repeater) có nhiệm vụ nhận tín hiệu rồi chuyển tiếp đến trạm kế tiếp trên vòng. Như vậy tín hiệu được lưu chuyển trên vòng theo một chuỗi liên tiếp các liên kết điểm-điểm giữa các repeater do đó cần có giao thức điều khiển việc cấp phát quyền được truyền dữ liệu trên vòng mạng cho trạm có nhu cầu.

Để tăng độ tin cậy của mạng ta có thể lắp đặt thêm các vòng dự phòng, nếu vòng chính có sự cố thì vòng phụ sẽ được sử dụng.

Mạng hình vòng có ưu nhược điểm tương tự mạng hình sao, tuy nhiên mạng hình vòng đòi hỏi giao thức truy nhập mạng phức tạp hơn mạng hình sao.

d) Kết nối hỗn hợp

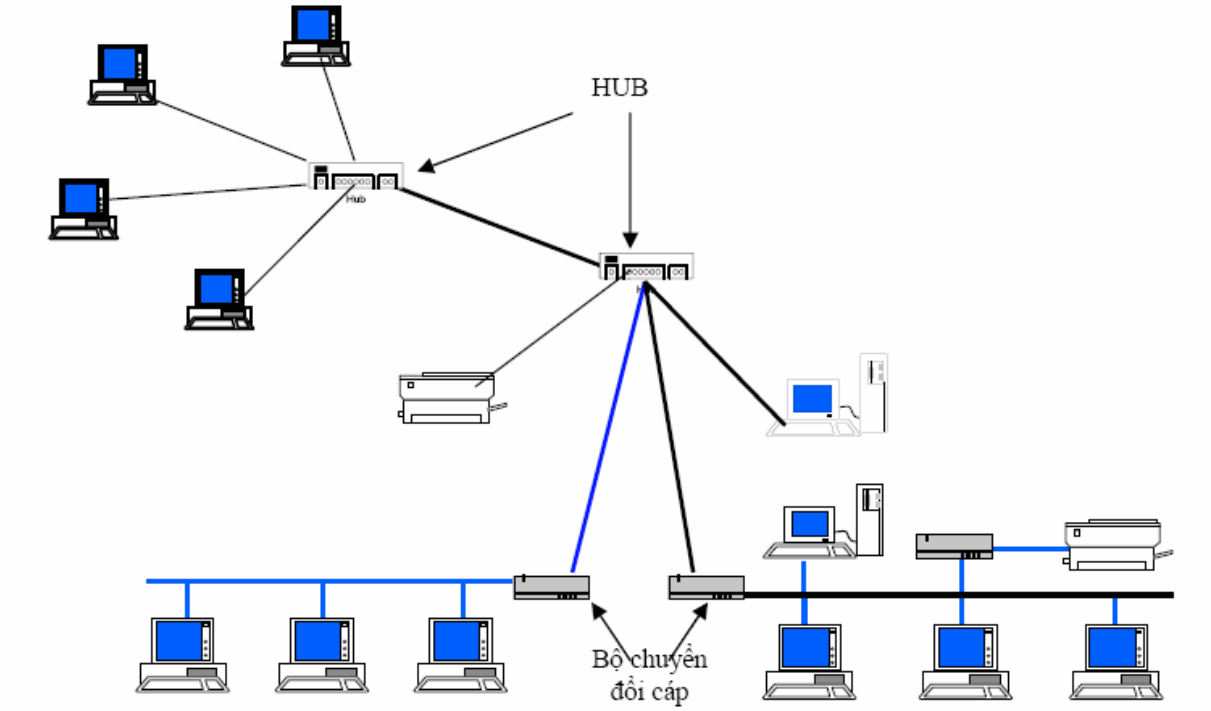
Là sự phối hợp các kiểu kết nối khác nhau, ví du hình cây là cấu trúc phân tầng của kiểu hình sao hay các HUB có thể được nối với nhau theo kiểu bus còn từ các HUB nối với các máy theo hình sao.



Hình 1.3 Kết nối kiểu vòng

II.3. Các phương pháp truy cập đường truyền vật lý

Trong mạng cục bộ, tất cả các trạm kết nối trực tiếp vào đường truyền chung. Vì vậy tín hiệu từ một trạm đưa lên đường truyền sẽ được các trạm khác "nghe thấy". Một vấn đề khác là, nếu nhiều trạm cùng gửi tín hiệu lên đường truyền đồng thời thì tín hiệu sẽ chồng lên nhau và bị hỏng. Vì vậy cần phải có một phương pháp tổ chức chia sẻ đường truyền để việc truyền thông được đúng đắn.



Hình 1.4. Một kết nối hỗn hợp

Có hai phương pháp chia sẻ đường truyền chung thường được dùng trong các mạng cục bộ:

- Truy nhập đường truyền một cách ngẫu nhiên, theo yêu cầu. Đương nhiên phải có tính đến việc sử dụng luân phiên và nếu trong trường hợp do có nhiều trạm cùng truyền tin dẫn đến tín hiệu bị trùm lên nhau thì phải truyền lại.

- Có cơ chế trọng tài để cấp quyền truy nhập đường truyền sao cho không xảy ra xung đột

II.3.1 Phương pháp đa truy nhập sử dụng sóng mang có phát hiện xung đột CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

Giao thức CSMA (Carrier Sense Multiple Access) - đa truy nhập có cảm nhận sóng mang được sử dụng rất phổ biến trong các mạng cục bộ. Giao thức này sử dụng

phương pháp thời gian chia ngăn theo đó thời gian được chia thành các khoảng thời gian đều đặn và các trạm chỉ phát lên đường truyền tại thời điểm đầu ngăn.

Mỗi trạm có thiết bị nghe tín hiệu trên đường truyền (tức là cảm nhận sóng mang). Trước khi truyền cần phải biết đường truyền có rỗi không. Nếu rỗi thì mới được truyền. Phương pháp này gọi là LBT (Listening before talking). Khi phát hiện xung đột, các trạm sẽ phải phát lại. Có một số chiến lược phát lại như sau:

- Giao thức CSMA không kiên trì.Trạm nghe đường, nếu kênh rỗi thì truyền, nếu không thì ngừng nghe một khoảng thời gian ngẫu nhiên rồi mới thực hiện lại thủ tục. Cách này có hiệu suất dùng kênh cao hơn. (1)

- Giao thức CSMA 1-kiên trì. Khi trạm phát hiện kênh rỗi trạm truyền ngay. Nhưng nếu có xung đột, trạm đợi khoảng thời gian ngẫu nhiên rồi truyền lại. Do vậy xác suất truyền khi kênh rỗi là 1. Chính vì thế mà giao thức có tên là CSMA 1-kiên trì. (2)

- Giao thức CSMA p-kiên trì. Khi đã sẵn sàng truyền, trạm cảm nhận đường, nếu đường rỗi thì thực hiện việc truyền với xác suất là $p < 1$ (tức là ngay cả khi đường rỗi cũng không hẳn đã truyền mà đợi khoảng thời gian tiếp theo lại tiếp tục thực hiện việc truyền với xác suất còn lại $q=1-p$. (3)

• Ta thấy giải thuật (1) có hiệu quả trong việc tránh xung đột vì hai trạm cần truyền thấy đường truyền bận sẽ cùng rút lui chờ trong những khoảng thời gian ngẫu nhiên khác nhau sẽ quay lại tiếp tục nghe đường truyền. Nhược điểm của nó là có thể có thời gian không sử dụng đường truyền sau mỗi cuộc gọi.

• Giải thuật (2) cố gắng làm giảm thời gian "chết" bằng cách cho phép một trạm có thể được truyền dữ liệu ngay sau khi một cuộc truyền kết thúc. Tuy nhiên nếu lúc đó lại có nhiều trạm đang đợi để truyền dữ liệu thì khả năng xảy ra xung đột sẽ rất lớn.

• Giải thuật (3) với giá trị p được họn hợp lý có thể tối thiểu hoá được cả khả năng xung đột lẫn thời gian "chết" của đường truyền.

• Xảy ra xung đột thường là do độ trễ truyền dẫn, mấu chốt của vấn đề là: các trạm chỉ "nghe" trước khi truyền dữ liệu mà không "nghe" trong khi truyền, cho nên thực tế có xung đột thế nhưng các trạm không biết do đó vẫn truyền dữ liệu.

• Để có thể phát hiện xung đột, CSMA/CD đã bổ xung thêm các quy tắc sau đây:

- Khi một trạm truyền dữ liệu, nó vẫn tiếp tục "nghe" đường truyền . Nếu phát hiện xung đột thì nó ngừng ngay việc truyền, nhờ đó mà tiết kiệm được thời gian và giải thông, nhưng nó vẫn tiếp tục gửi tín hiệu thêm một thời gian nữa để đảm bảo rằng tất cả các trạm trên mạng đều "nghe" được sự kiện

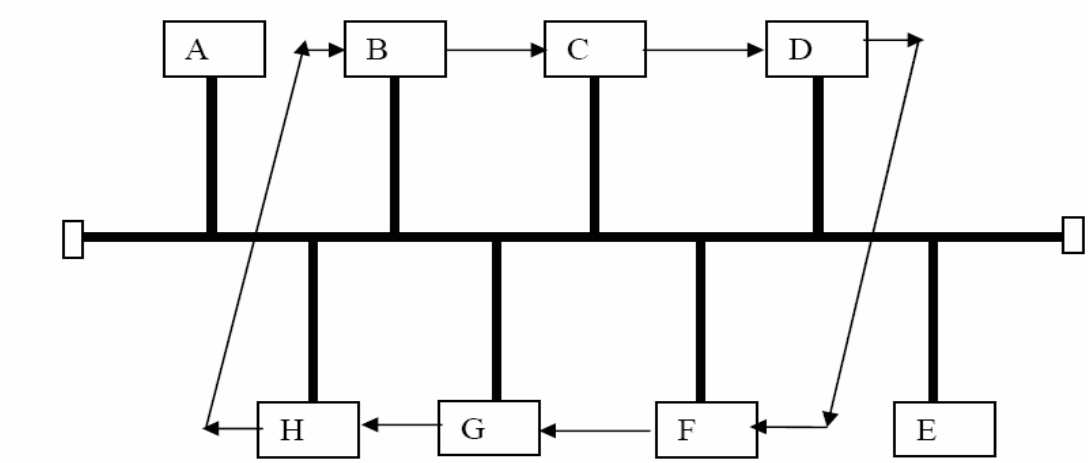
này.(như vậy phải tiếp tục nghe đường truyền trong khi truyền để phát hiện đụng độ (Listening While Talking))

- Sau đó trạm sẽ chờ trong một khoảng thời gian ngẫu nhiên nào đó rồi thử truyền lại theo quy tắc CSMA.

Giao thức này gọi là **CSMA có phát hiện xung đột** (Carrier Sense Multiple Access with Collision Detection viết tắt là CSMA/CD), dùng rộng rãi trong LAN và MAN.

II.3.2. Phương pháp Token Bus

Nguyên lý chung của phương pháp này là để cấp phát quyền truy nhập đường truyền cho các trạm đang có nhu cầu truyền dữ liệu, một thẻ bài được lưu chuyển trên một vòng logic được thiết lập bởi các trạm đó. Khi một trạm nhận được thẻ bài thì sẽ được phép sử dụng đường truyền trong một thời gian nhất định. Trong khoảng thời gian đó nó có thể truyền một hay nhiều đơn vị dữ liệu. Khi đã truyền xong dữ liệu hoặc thời gian đã hết thì trạm đó phải chuyển thẻ bài cho trạm tiếp theo. Như vậy,



Hình 1.5. Ví dụ về vòng logic

công việc đầu tiên là thiết lập vòng logic (hay còn gọi là vòng ảo) bao gồm các trạm đang có nhu cầu truyền dữ liệu được xác định vị trí theo một chuỗi thứ tự mà trạm cuối cùng của chuỗi sẽ tiếp liền sau bởi trạm đầu tiên. Mỗi trạm sẽ biết địa chỉ của trạm liền trước và kề sau nó. Thứ tự của các trạm trên vòng logic có thể độc lập với thứ tự vật lý. Các trạm không hoặc chưa có nhu cầu truyền dữ liệu không được vào trong vòng logic.

Trong ví dụ trên, các trạm A, E nằm ngoài vòng logic do đó chỉ có thể tiếp nhận được dữ liệu dành cho chúng.

Việc thiết lập vòng logic không khó nhưng việc duy trì nó theo trạng thái thực tế của mạng mới là khó. Cụ thể phải thực hiện các chức năng sau:

a) Bổ xung một trạm vào vòng logic : các trạm nằm ngoài vòng logic cần được xem xét một cách định kỳ để nếu có nhu cầu truyền dữ liệu thì được bổ xung vào vòng logic.

b) Loại bỏ một vòng khỏi vòng logic : khi một trạm không có nhu cầu truyền dữ liệu thì cần loại bỏ nó ra khỏi vòng logic để tối ưu hoá việc truyền dữ liệu bằng thẻ bài

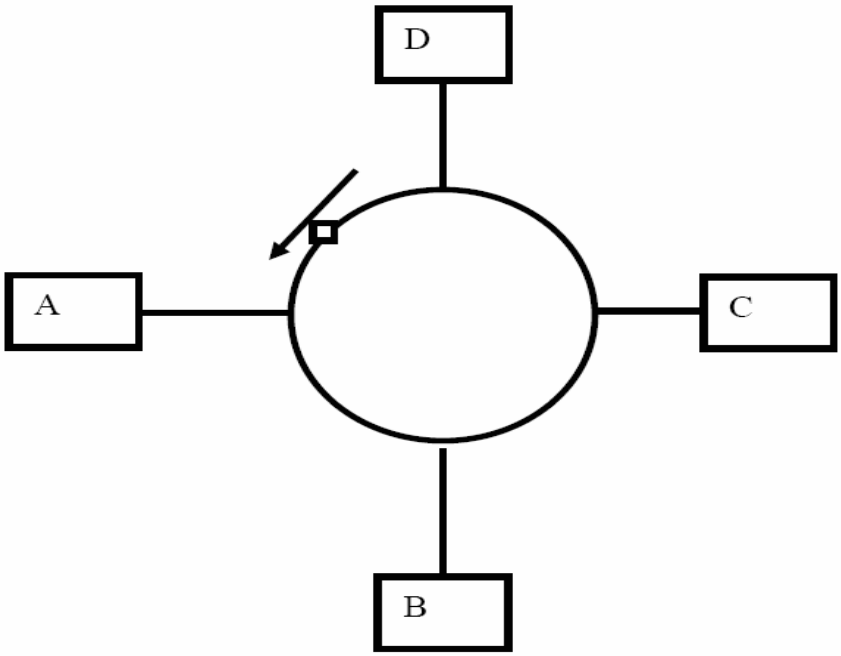
c) Quản lý lỗi : một số lỗi có thể xảy ra như trùng hợp địa chỉ, hoặc đứt vòng logic.

d) Khởi taọ vòng logic : khi khởi tạo mạng hoặc khi đứt vòng logic cần phải khởi tạo lại vòng logic.

II.3.2. Phương pháp Token Ring

Phương pháp này cũng dựa trên nguyên tắc dùng thẻ bài để cấp phát quyền truy nhập đường truyền. Nhưng ở đây thẻ bài lưu chuyển theo theo vòng vật lý chứ không theo vòng logic như dối với phương pháp token bus.

Thẻ bài là một đơn vị truyền dữ liệu đặc biệt trong đó có một bit biểu diễn trạng thái của thẻ (bận hay rỗi). Một trạm muốn truyền dữ liệu phải chờ cho tới khi nhận được thẻ bài "rỗi". Khi đó trạm sẽ đổi bit trạng thái thành "bận" và truyền một đơn vị dữ liệu đi cùng với thẻ bài đi theo chiều của vòng. Lúc này không còn thẻ bài "rỗi " nữa do đó các trạm muốn truyền dữ liệu phải đợi. Dữ liệu tới trạm đích được sao chép



Hình 1.6. Thẻ bài trong mạng Ring

lại, sau đó cùng với thẻ bài trở về trạm nguồn. Trạm nguồn sẽ xoá bỏ dữ liệu đổi bit trạng thái thành "rỗi" và cho lưu chuyển thẻ trên vòng để các trạm khác có nhu cầu truyền dữ liệu được phép truyền

Sự quay trở lại trạm nguồn của dữ liệu và thẻ bài nhằm tạo khả năng báo nhận tự nhiên: trạm đích có thể gửi vào đơn vị dữ liệu (phần header) các thông tin về kết quả tiếp nhận dữ liệu của mình. Chẳng hạn các thông tin đó có thể là: trạm đích không tồn tại hoặc không hoạt động, trạm đích tồn tại nhưng dữ liệu không được sao chép, dữ liệu đã được tiếp nhận, có lỗi...

Trong phương pháp này cần giải quyết hai vấn đề có thể dẫn đến phá vỡ hệ thống đó là mất thẻ bài và thẻ bài "bận" lưu chuyển không dừng trên vòng .Có nhiều phương pháp giải quyết các vấn đề trên, dưới đây là một phương pháp được khuyến nghị:

Đối với vấn đề mất thẻ bài có thể quy định trước một trạm điều khiển chủ động. Trạm này sẽ theo dõi, phát hiện tình trạng mất thẻ bài bằng cách dùng cơ chế ngưỡng thời gian (time - out) và phục hồi bằng cách phát đi một thẻ bài "rỗi" mới.

Đối với vấn đề thẻ bài bận lưu chuyển không dừng, trạm điều khiển sử dụng một bit trên thẻ bài để đánh dấu khi gặp một thẻ bài "bận" đi qua nó. Nếu nó gặp lại thẻ bài bận với bit đã đánh dấu đó có nghĩa là trạm nguồn đã không nhận lại được đơn vị dữ liệu của mình do đó thẻ bài "bận" cứ quay vòng mãi. Lúc đó trạm điều khiển sẽ chủ động đổi bit trạng thái "bận" thành "rỗi" và cho thẻ bài chuyển tiếp trên vòng. Trong phương pháp này các trạm còn lại trên mạng sẽ đóng vai trò bị động, chúng theo dõi phát hiện tình trạng sự cố trên trạm chủ động và thay thế trạm chủ động nếu cần.

III. Chuẩn hoá mạng máy tính

III.1. Vấn đề chuẩn hoá mạng và các tổ chức chuẩn hoá mạng

Khi thiết kế, các nhà thiết kế tự do lựa chọn kiến trúc mạng cho riêng mình. Từ đó dẫn tới tình trạng không tương thích giữa các mạng máy tính với nhau. Nhu cầu trao đổi thông tin càng lớn thúc đẩy việc xây dựng khung chuẩn về kiến trúc mạng để làm căn cứ cho các nhà thiết kế và chế tạo thiết bị mạng.

Chính vì lý do đó, tổ chức tiêu chuẩn hoá quốc tế ISO (Internatinal Organnization for Standarzation) đã xây dựng mô hình tham chiếu cho việc kết nối các hệ thống mở OSI (reference model for Open Systems Interconnection). Mô hình này là cơ sở cho việc kết nối các hệ thống mở phục vụ cho các ứng dụng phân tán.

Có hai loại chuẩn cho mạng đó là :

- Truyền tin dạng bit qua kênh vật lý.
- Có thể có nhiều kênh.

b) Lớp liên kết dữ liệu

Lớp này đảm bảo việc biến đổi các tin dạng bit nhận được từ lớp dưới (vật lý) sang khung số liệu, thông báo cho hệ phát, kết quả thu được sao cho các thông tin truyền lên cho mức 3 không có lỗi. Các thông tin truyền ở mức 1 có thể làm hỏng các thông tin khung số liệu (frame error). Phần mềm mức hai sẽ thông báo cho mức một truyền lại các thông tin bị mất / lỗi. Đồng bộ các hệ có tốc độ xử lý tính khác nhau, một trong những phương pháp hay sử dụng là dùng bộ đệm trung gian để lưu giữ số liệu nhận được. Độ lớn của bộ đệm này phụ thuộc vào tương quan xử lý của các hệ thu và phát. Trong trường hợp đường truyền song công toàn phần, lớp datalink phải đảm bảo việc quản lý các thông tin số liệu và các thông tin trạng thái.

c) Lớp mạng

Nhiệm vụ của lớp mạng là đảm bảo chuyển chính xác số liệu giữa các thiết bị cuối trong mạng. Để làm được việc đó, phải có chiến lược đánh địa chỉ thống nhất trong toàn mạng. Mỗi thiết bị cuối và thiết bị mạng có một địa chỉ mạng xác định. Số liệu cần trao đổi giữa các thiết bị cuối được tổ chức thành các gói (packet) có độ dài thay đổi và được gán đầy đủ địa chỉ nguồn (source address) và địa chỉ đích (destination address).

Lớp mạng đảm bảo việc tìm đường tối ưu cho các gói dữ liệu bằng các giao thức chọn đường dựa trên các thiết bị chọn đường (router). Ngoài ra, lớp mạng có chức năng điều khiển lưu lượng số liệu trong mạng để tránh xảy ra tắc ngẽn bằng cách chọn các chiến lược tìm đường khác nhau để quyết định việc chuyển tiếp các gói số liệu.

d) Lớp chuyển vận

Lớp này thực hiện các chức năng nhận thông tin từ lớp phiên (**session**) chia thành các gói nhỏ hơn và truyền xuống lớp dưới, hoặc nhận thông tin từ lớp dưới chuyển lên phục hồi theo cách chia của hệ phát (Fragmentation and Reassembly). Nhiệm vụ quan trọng nhất của lớp vận chuyển là đảm bảo chuyển số liệu chính xác giữa hai thực thể thuộc lớp phiên (end-to-end control). Để làm được việc đó, ngoài chức năng kiểm tra số tuần tự phát, thu, kiểm tra và phát hiện, xử lý lỗi.Lớp vận chuyển còn có chức năng điều khiển lưu lượng số liệu để đồng bộ giữa thể thu và phát, tránh tắc nghẽn số liệu khi chuyển qua lớp mạng. Ngoài ra, nhiều thực thể lớp phiên có thể trao đổi số liệu trên cùng một kết nối lớp mạng (multiplexing).

e) Lớp phiên

Liên kết giữa hai thực thể có nhu cầu trao đổi số liệu, ví dụ người dùng và một máy tính ở xa, được gọi là một phiên làm việc. Nhiệm vụ của lớp phiên là quản lý việc trao đổi số liệu, ví dụ: thiết lập giao diện giữa người dùng và máy, xác định thông số điều khiển trao đổi số liệu (tốc độ truyền, số bit trong một byte, có kiểm tra lỗi parity hay không, v.v.), xác định loại giao thức mô phỏng thiết bị cuối (terminal emulation), v.v. Chức năng quan trọng nhất của lớp phiên là đảm bảo đồng bộ số liệu bằng cách thực hiện các điểm kiểm tra. Tại các điểm kiểm tra này, toàn bộ trạng thái và số liệu của phiên làm việc được lưu trữ trong bộ nhớ đệm. Khi có sự cố, có thể khởi tạo lại phiên làm việc từ điểm kiểm tra cuối cùng (không phải khởi tạo lại từ đầu).

f) Lớp thể hiện

Nhiệm vụ của lớp thể hiện là thích ứng các cấu trúc dữ liệu khác nhau của người dùng với cấu trúc dữ liệu thống nhất sử dụng trong mạng. Số liệu của người dùng có thể được nén và mã hoá ở lớp thể hiện, trước khi chuyển xuống lớp phiên. Ngoài ra, lớp thể hiện còn chứa các thư viện các yêu cầu của người dùng, thư viện tiện ích, ví dụ thay đổi dạng thể hiện của các tệp, nén tệp...

g) Lớp ứng dụng

Lớp ứng dụng cung cấp các phương tiện để người sử dụng có thể truy nhập được vào môi trường OSI, đồng thời cung cấp các dịch vụ thông tin phân tán. Lớp mạng cho phép người dùng khai thác các tài nguyên trong mạng tương tự như tài nguyên tại chỗ.

III.3. Các chuẩn kết nối thông dụng nhất IEEE 802.X và ISO 8802.X

Bên cạnh việc chuẩn hoá cho mạng nối chung dẫn đến kết quả cơ bản nhất là mô hình tham chiếu OSI như đã giới thiệu. Việc chuẩn hoá mạng cục bộ nói riêng đã được thực hiện từ nhiều năm nay để đáp ứng sự phát triển của mạng cục bộ.

Cũng như đối với mạng nói chung, có hai loại chuẩn cho mạng cục bộ, đó là :

- Các chuẩn chính thức (de jure) do các tổ chức chuẩn quốc gia và quốc tế ban hành.

- Các chuẩn tực tiễn (de facto) do các hãng soản xuất, các tổ chức người sử dụng xây dựng và được dùng rộng rãi trong thực tế

- Các chuẩn IEEE 802.x và ISO 8802.x

IEEE là tổ chức đi tiên phong trong lĩnh vực chuẩn hoá mạng cục bộ với đề án IEEE 802 với kết quả là một loạt các chuẩn thuộc họ IEEE 802.x ra đời. Cuối những năm 80, tổ chức ISO đã tiếp nhận họ chuẩn này và ban hành thành chuẩn quốc tế dưới mã hiệu tương ứng là ISO 8802.x.

IEEE 802.: là chuẩn đặc tả kiến trúc mạng, kết nối giữa các mạng và việc quản trị mạng đối với mạng cục bộ.

IEEE 802.2: là chuẩn đặc tả tầng dịch vụ giao thức của mạng cục bộ.

IEEE 802.3: là chuẩn đặc tả một mạng cục bộ dựa trên mạng Ethernet nổi tiếng của Digital, Intel và Xerox hợp tác xây dựng từ năm 1980.

Tầng vật lý của IEEE 802.3 có thể dùng các phương án sau để xây dựng:

- 10BASE5 : tốc độ 10Mb/s, dùng cáp xoắn đôi không bọc kim UTP (Unshield Twisted Pair), với phạm vi tín hiệu lên tới 500m, topo mạng hình sao.

- 10BASE2 : tốc độ 10Mb/s, dùng cáp đồng trục thin-cable với trở kháng 50 Ohm, phạm vi tín hiệu 200m,topo mạng dạng bus.

- 10BASE5 : tốc độ 10Mb/s, dùng cáp đồng trục thick-cable (đường kính 10mm) với trở kháng 50 Ohm, phạm vi tín hiệu 500m, topo mạng dạng bus.

- 10BASE-F: dùng cáp quang, tốc độ 10Mb/s phạm vi cáp 2000m.

IEEE 802.4: là chuẩn đặc tả mạng cục bộ với topo mạng dạng bus dùng thẻ bài để điều việc truy nhập đường truyền.

IEEE 802.5: là chuẩn đặc tả mạng cục bộ với topo mạng dạng vòng (ring) dùng thẻ bài để điều việc truy nhập đường truyền.

IEEE 802.6: là chuẩn đặc tả mạng tốc độ cao kết nối với nhiều mạng cục bộ thuộc các khu vực khác nhau của một đô thị (còn được gọi là mạng MAN - Metropolitan Area Network)

IEEE 802.9: là chuẩn đặc tả mạng tích hợp dữ liệu và tiếng nói bao gồm 1 kênh dị bộ 10 Mb/s cùng với 96 kênh 64Kb/s. Chuẩn này được thiết kế cho môi trường có lượng lưu thông lớn và cấp bách.

IEEE 802.10: là chuẩn đặc tả về an toàn thông tin trong các mạng cục bộ có khả năng liên tác .

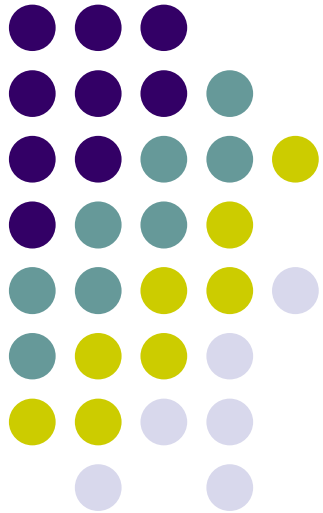
IEEE 802.11: là chuẩn đặc tả mạng cục bộ không dây (Wireless LAN) hiện đang được tiếp tục phát triển.

IEEE 802.12: là chuẩn đặc tả mạng cục bộ dựa trên công nghệ được đề xuất bởi AT&T, IBM và HP gọi là 100 VG - AnyLAN. Mạng này có topo mạng hình sao và một phương pháp truy nhập đường truyền có điều khiển tranh chấp. Khi có nhu cầu truyền dữ liệu, một trạm sẽ gửi yêu cầu đến hub và trạm chỉ có truyền dữ liệu khi hub cho phép.

Mạng máy tính

Giảng viên: Ngô Hồng Sơn

Bộ môn Truyền thông và Mạng máy tính
Khoa CNTT- ĐHBK Hà Nội

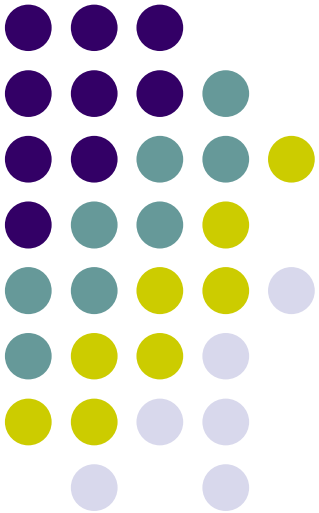


Nội dung

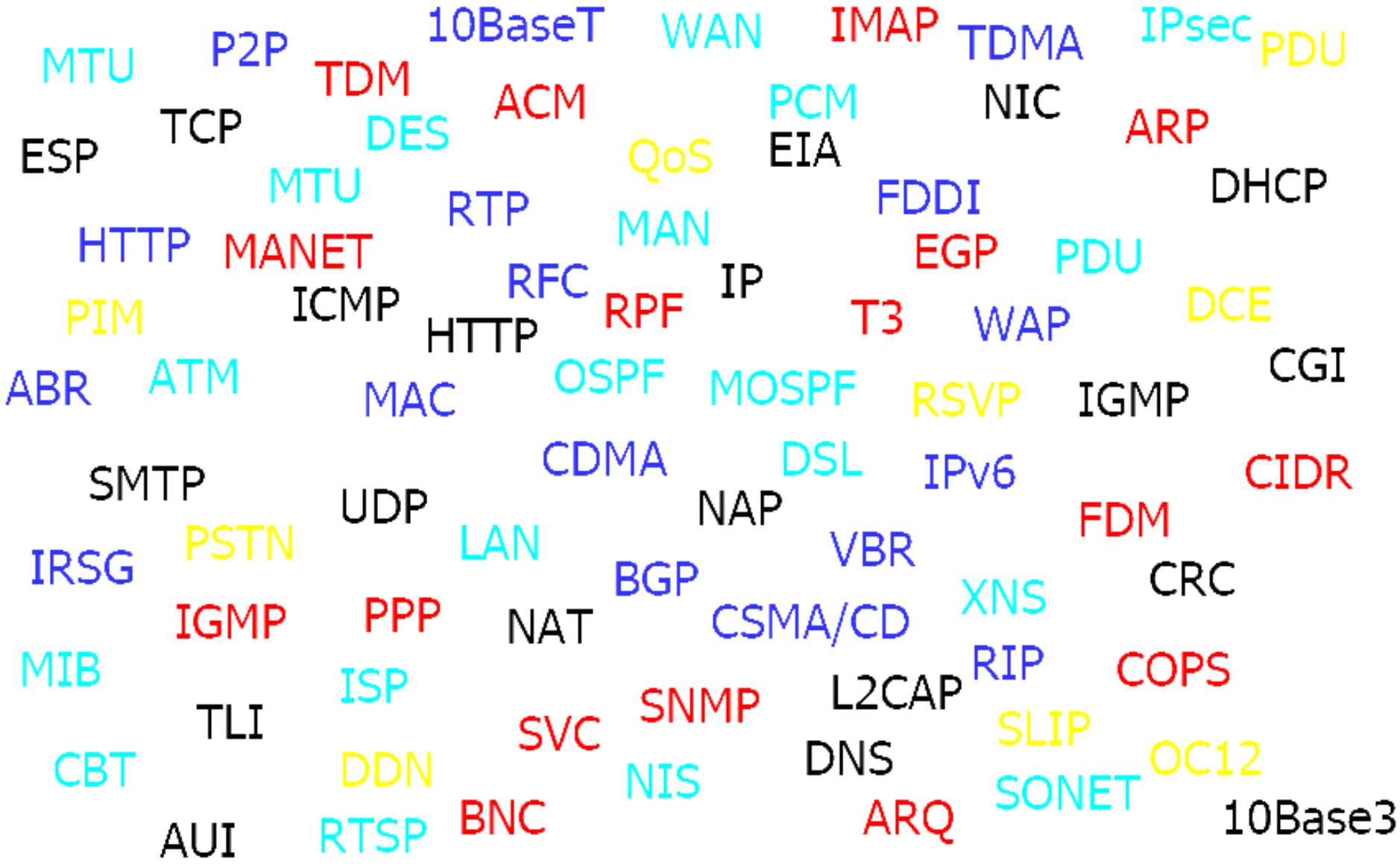
- Giới thiệu môn học
- Cơ bản về mạng máy tính
- Lược sử mạng máy tính và Internet
- Internet ở Việt Nam

Giới thiệu môn học

Mục đích
Chủ đề và lịch học
Đánh giá
Liên hệ giáo viên

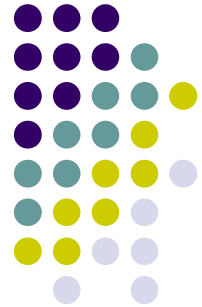


Mục đích môn học



Source: Hung Q Ngo' course

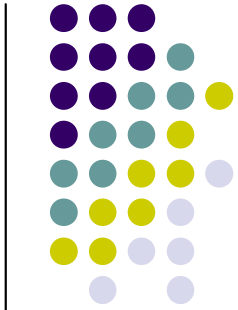
Mục đích môn học



Kết thúc môn học này, các sinh viên ngành CNTT sẽ có khả năng:

- Nêu và giải thích các công nghệ liên quan đến mạng máy tính và Internet
  - Nguyên lý cơ bản của mạng máy tính
  - Họ giao thức TCP/IP
- Giải thích được Internet hoạt động như thế nào
- Sử dụng hiệu quả Internet, vận dụng để có thể cài đặt các công nghệ và dịch vụ mới

Lịch học dự kiến



1	22-Aug-08	Giới thiệu môn học, lịch sử mạng máy tính
2	29-Aug-08	Cơ bản về mạng máy tính
3	5-Sep-08	Tầng mạng, IP
4	12-Sep-08	Bài toán và các giao thức chọn đường đi
5	19-Sep-08	Tầng giao vận, TCP, UDP
6	26-Sep-08	Tầng ứng dụng, Web, Mail, FTP, DNS
7	3-Oct-08	Tầng liên kết dữ liệu

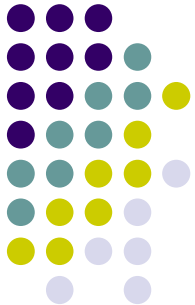
Lịch học dự kiến

8	10-Oct-08	LAN (VLAN, WLAN), WAN (...)
9	17-Oct-08	Tầng vật lí, các vấn đề về truyền số liệu
10	24-Oct-08	Advanced topic: Mạng thế hệ mới
11	31-Oct-08	Advanced topic: An toàn an ninh mạng
12	7-Nov-08	*Topic presentation*
13	14-Nov-08	*Topic presentation*
14	21-Nov-08	*Topic presentation*
15	28-Nov-08	Tổng kết và ôn tập

Đánh giá kết quả

- Bài tập lớn 40%
  - Hai bài
- Thi cuối kỳ 60%

Cách làm việc



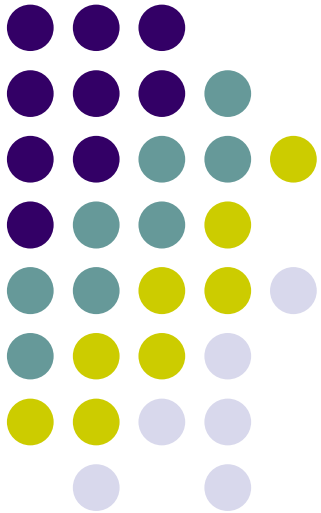
- Để học tốt
  - Đọc tài liệu trước khi đến lớp
  - Tham gia tích cực vào bài giảng
    - Thảo luận, trả lời và ĐẶT câu hỏi.
  - Tìm kiếm câu trả lời trên Web hoặc thảo luận với bạn bè
- Liên hệ với giáo viên
  - 8:30 – 10:00 sáng thứ 2 hàng tuần.
  - Bộ môn TTM – Khoa CNTT, 329 C1
  - ĐT: 8680896
  - Mail: sonnh@it-hut.edu.vn

Tài liệu tham khảo

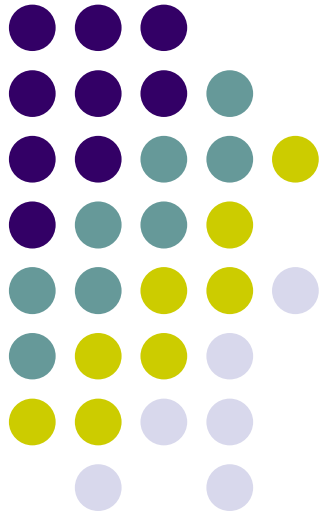
- [1] Nguyễn Thúc Hải, “Mạng máy tính và các hệ thống mở”
- [2] W. Stallings, “Data and Computer Communications”, Mac Millan,
- [3] James F. Kurose, Keith W. Ross, “Computer networks: a top-down approach featuring the Internet”, Addison Wesley.

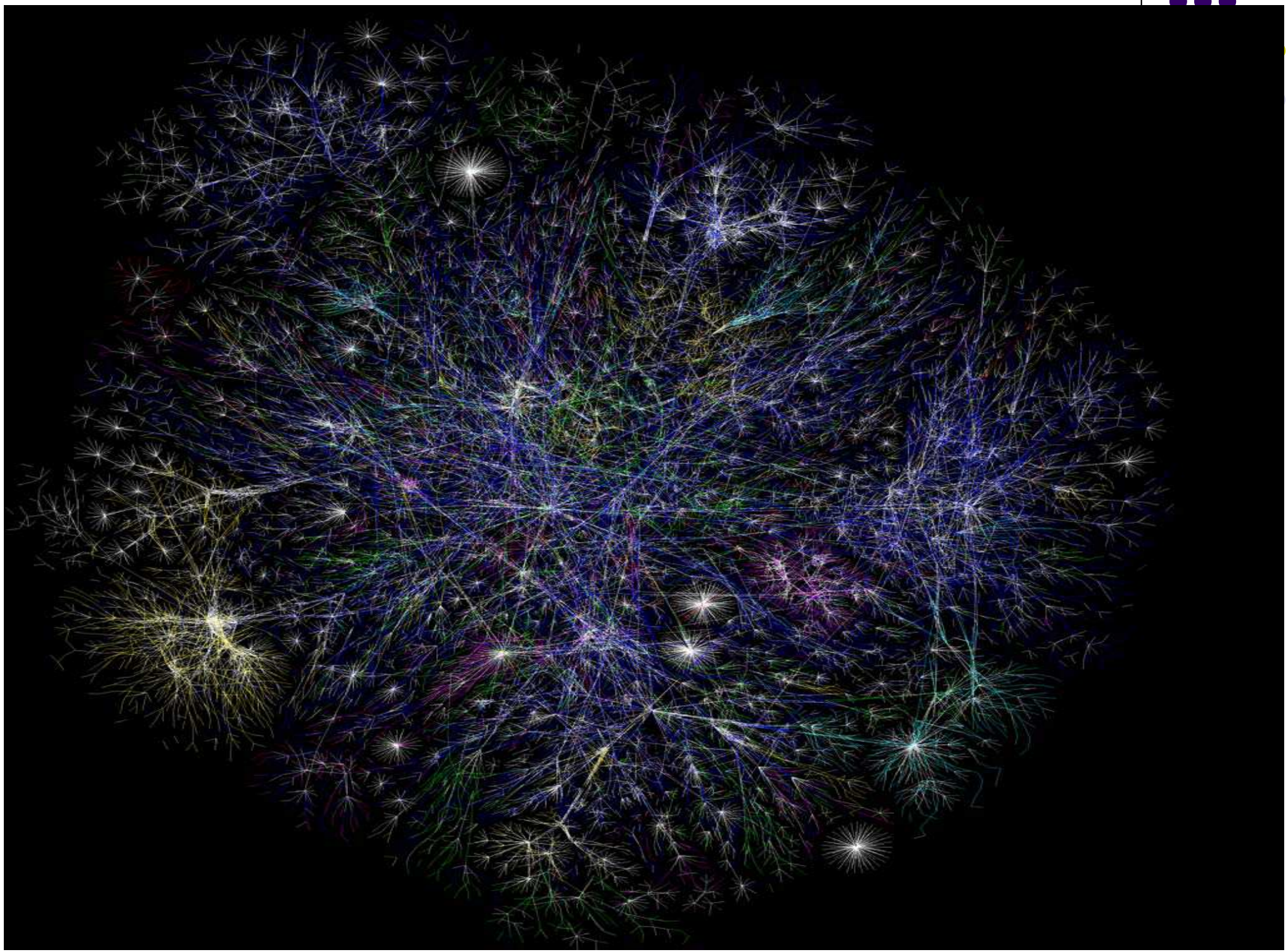
Cơ bản về mạng máy tinh

Khái niệm mạng máy tính
Kiến trúc mạng
Chuyển mạch gói vs. chuyển mạch kênh



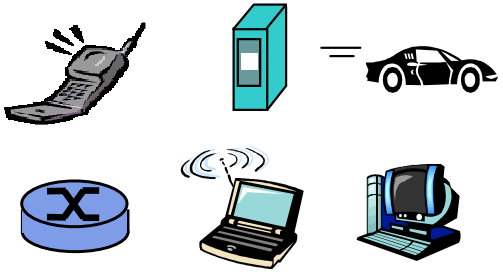
Mạng máy tính là gì

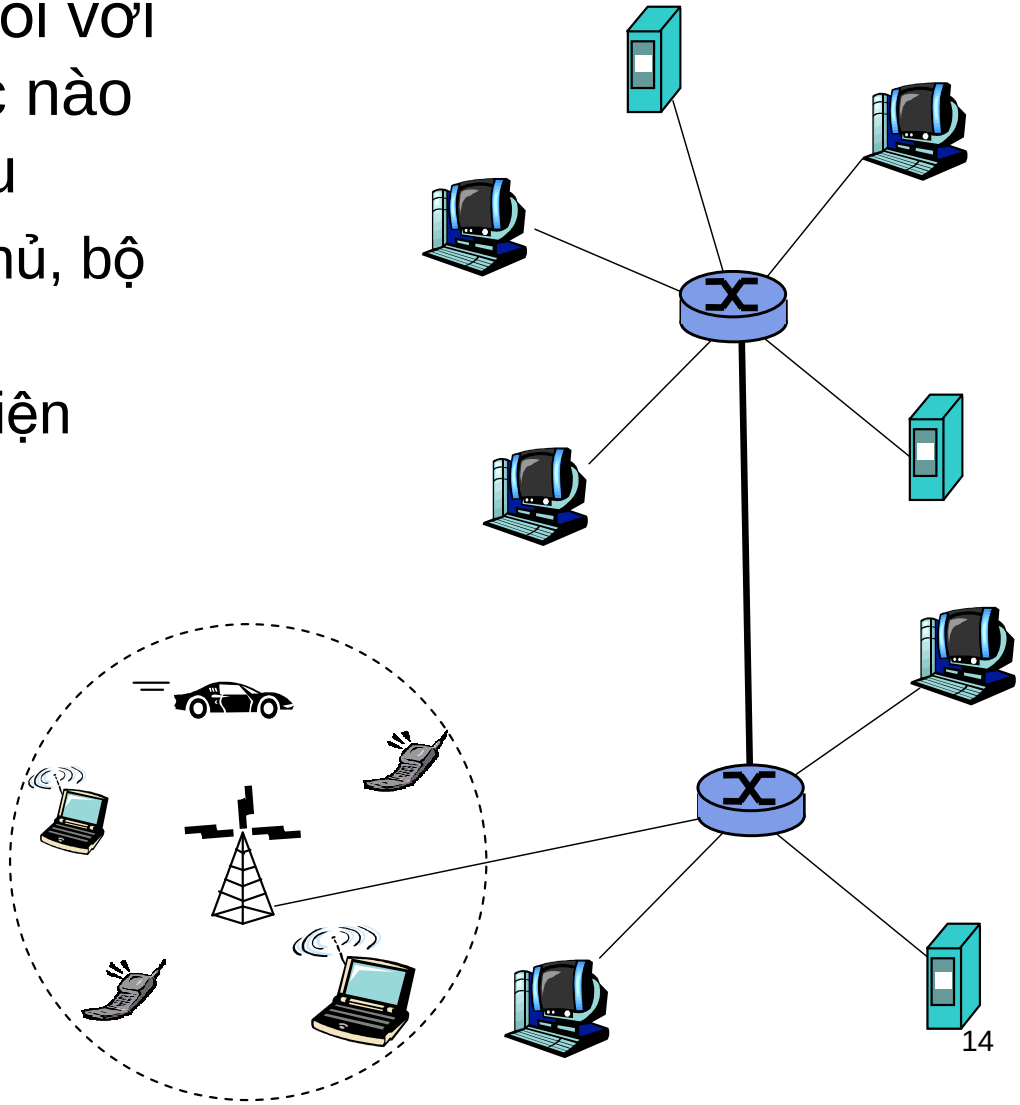




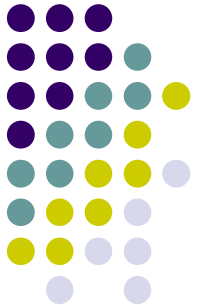
Khái niệm

- Tập hợp các máy tính kết nối với nhau dựa trên một kiến trúc nào đó để có thể trao đổi dữ liệu
  - Máy tính: máy trạm, máy chủ, bộ định tuyến
  - Kết nối bằng một phương tiện truyền
  - Theo một kiến trúc mạng
- Các dạng máy tính?



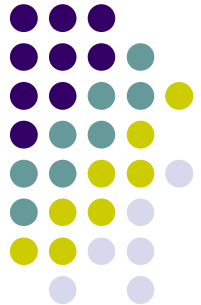


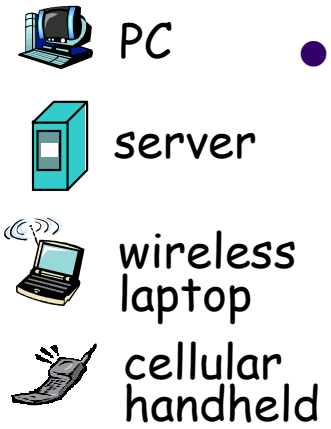
Ví dụ về mạng máy tính



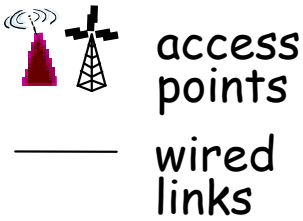
- Mạng Internet
- Mạng Ethernet
- Mạng LAN không dây: 802:11
- Hệ thống mạng ngân hàng: mạng lưới máy rút tiền
- Hệ thống bán vé tàu qua mạng
- …

Internet ngày nay



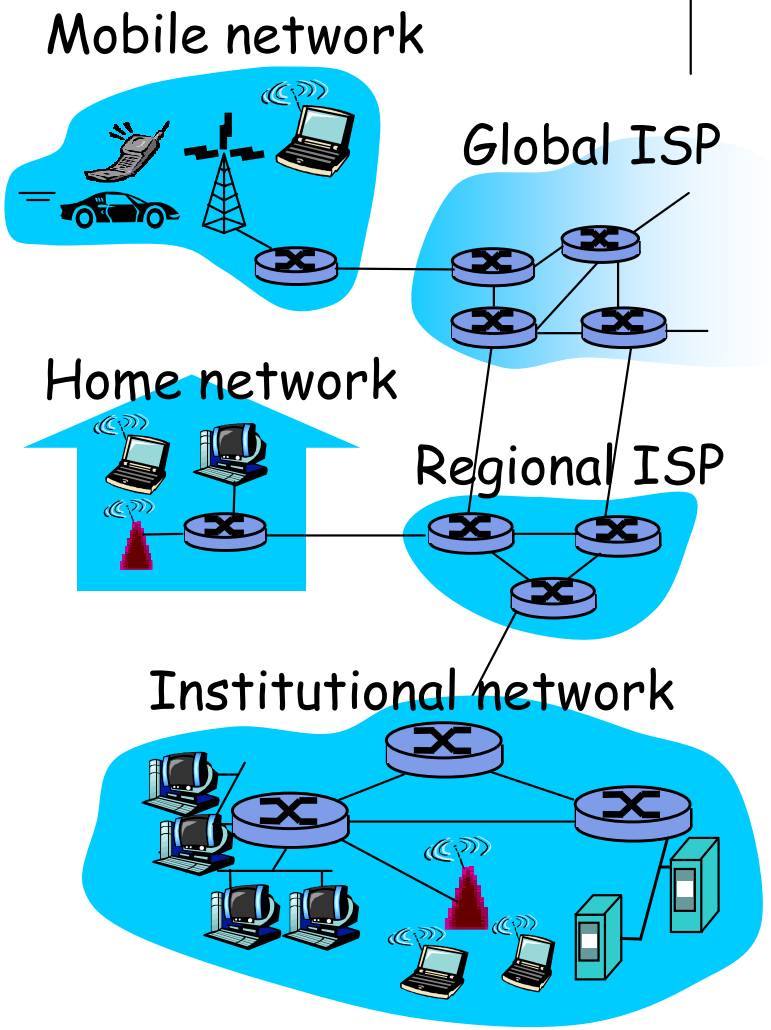


- Hàng triệu thiết bị kết nối: *hosts = end systems*
  - chạy *các ứng dụng mạng*

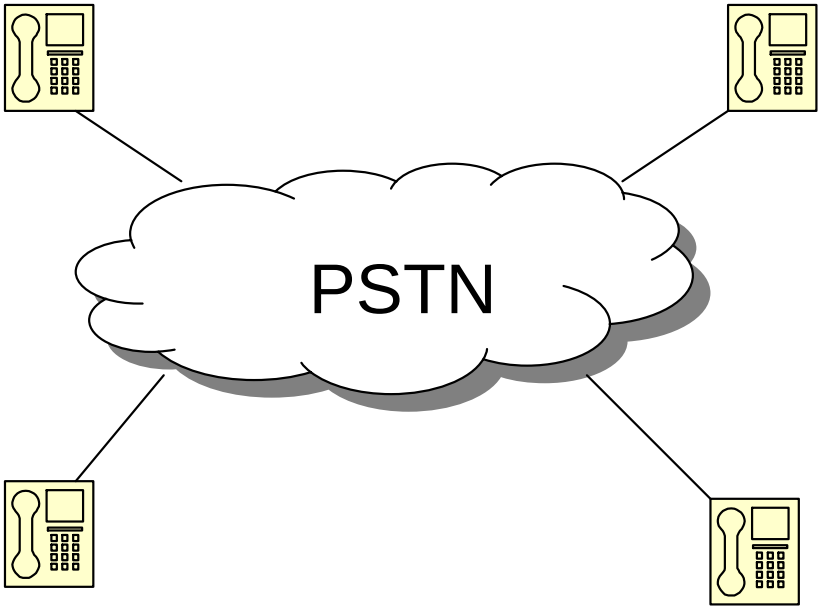


- *Đường truyền*
  - Cáp quang, đồng, vệ tinh, …
  - Tốc độ truyền = *băng thông*

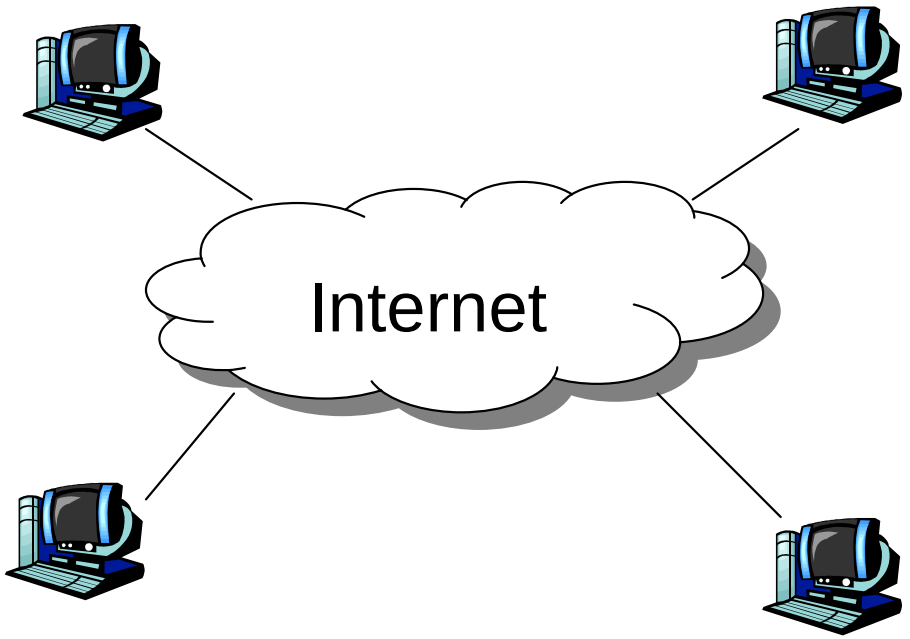
- *Bộ định tuyến:* chuyển tiếp các gói tin (dữ liệu)



Xử lý tập trung hay phân tán



- Mạng điện thoại công cộng, tập trung: mạng xử lý mọi thứ

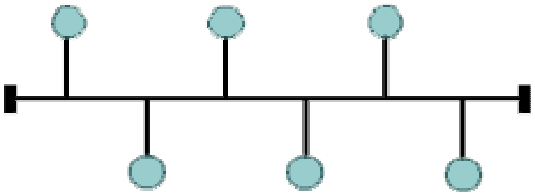


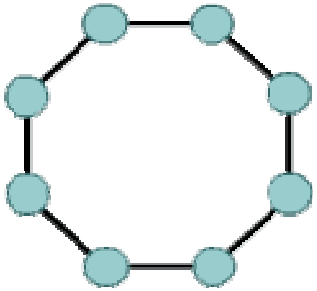
- Máy tính có khả năng lớn hơn
- Hầu hết các chức năng tập trung ở mạng máy tính
- Mạng: Truyền dữ liệu

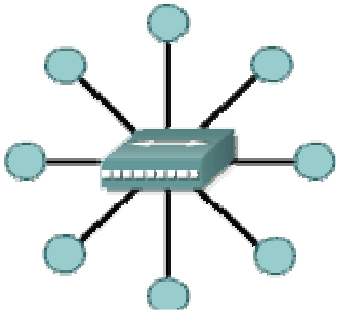
PSTN: Public Switch Telephone Network

Kiến trúc mạng

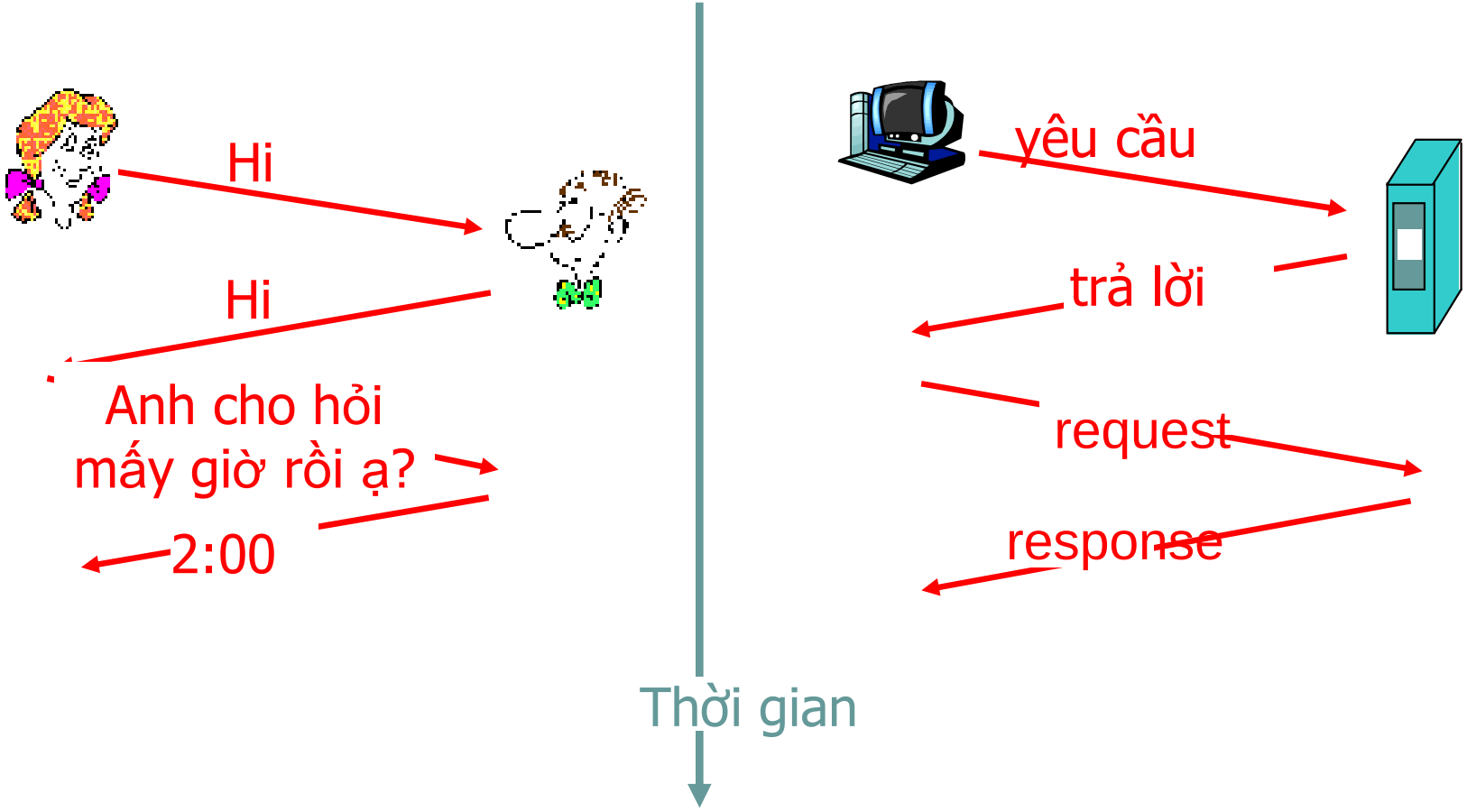
- Kiến trúc mạng: Hình trạng (topology) và giao thức (protocol)
- Hình trạng mạng
  - Trục (Bus), Vòng (Ring), Sao (Star)…
  - Thực tế là sự kết hợp của nhiều hình trạng khác nhau







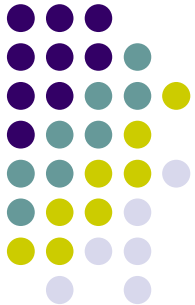
Giao thức là gì?



Giao thức người-người

Giao thức máy-máy

Giao thức mạng

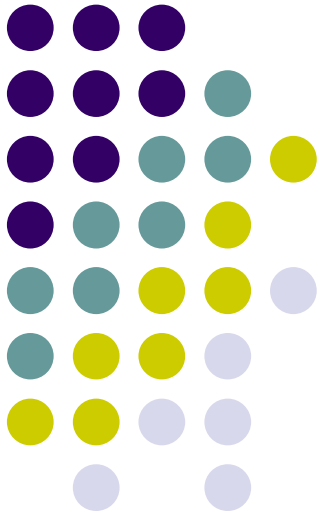


- Protocol: Quy tắc để truyền thông
  - Gửi một thông điệp với yêu cầu hoặc thông tin
  - Nhận một thông điệp với thông tin, sự kiện hoặc hành động
- Định nghĩa khuôn dạng và thứ tự truyền, nhận thông điệp giữa các thực thể trên mạng hoặc các hành động tương ứng khi nhận được thông điệp
- Ví dụ về giao thức mạng: TCP, UDP, IP, HTTP, Telnet, SSH, Ethernet, …

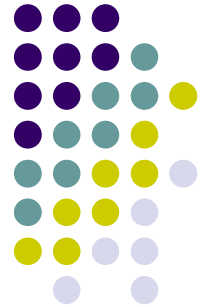
Mô hình truyền thông

Chuyển mạch gói vs. Chuyển mạch kênh

Hướng liên kết vs. Không liên kết



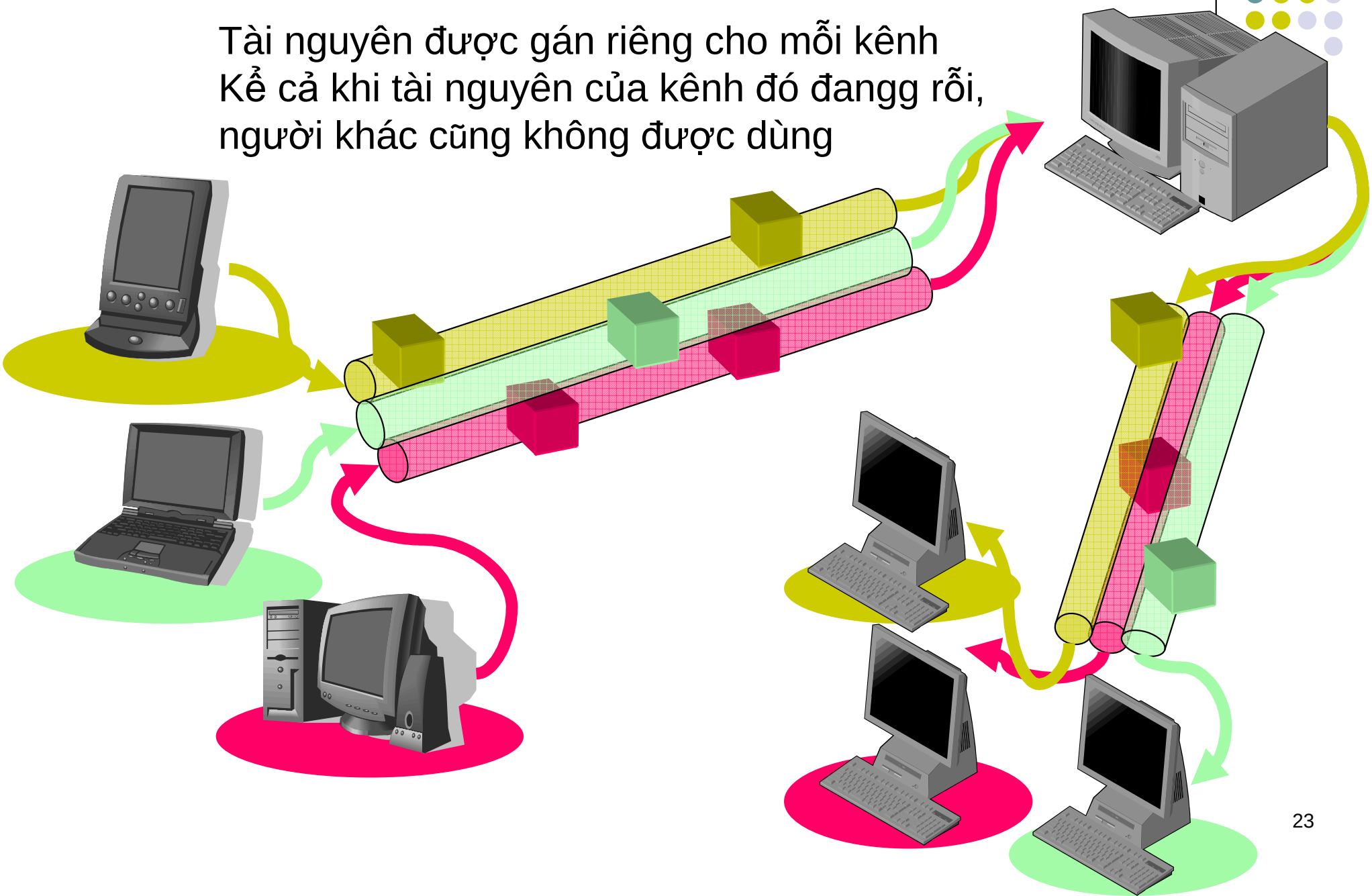
Chuyển mạch gói vs. Chuyển mạch kênh



- Chuyển mạch kênh
  - Trao đổi dữ liệu sử dụng một kênh riêng .
  - Mỗi liên kết sử dụng một kênh. Tài nguyên cho kênh đó không được sử dụng bởi người khác trừ khi đóng liên kết
- Chuyển mạch gói
  - Dữ liệu được chia thành các gói nhỏ (packets), và được chuyển qua mạng
  - Nhiều liên kết có thể chia sẻ một kênh
  - Internet (với giao thức IP – Internet Protocol) sử dụng chuyển mạch gói

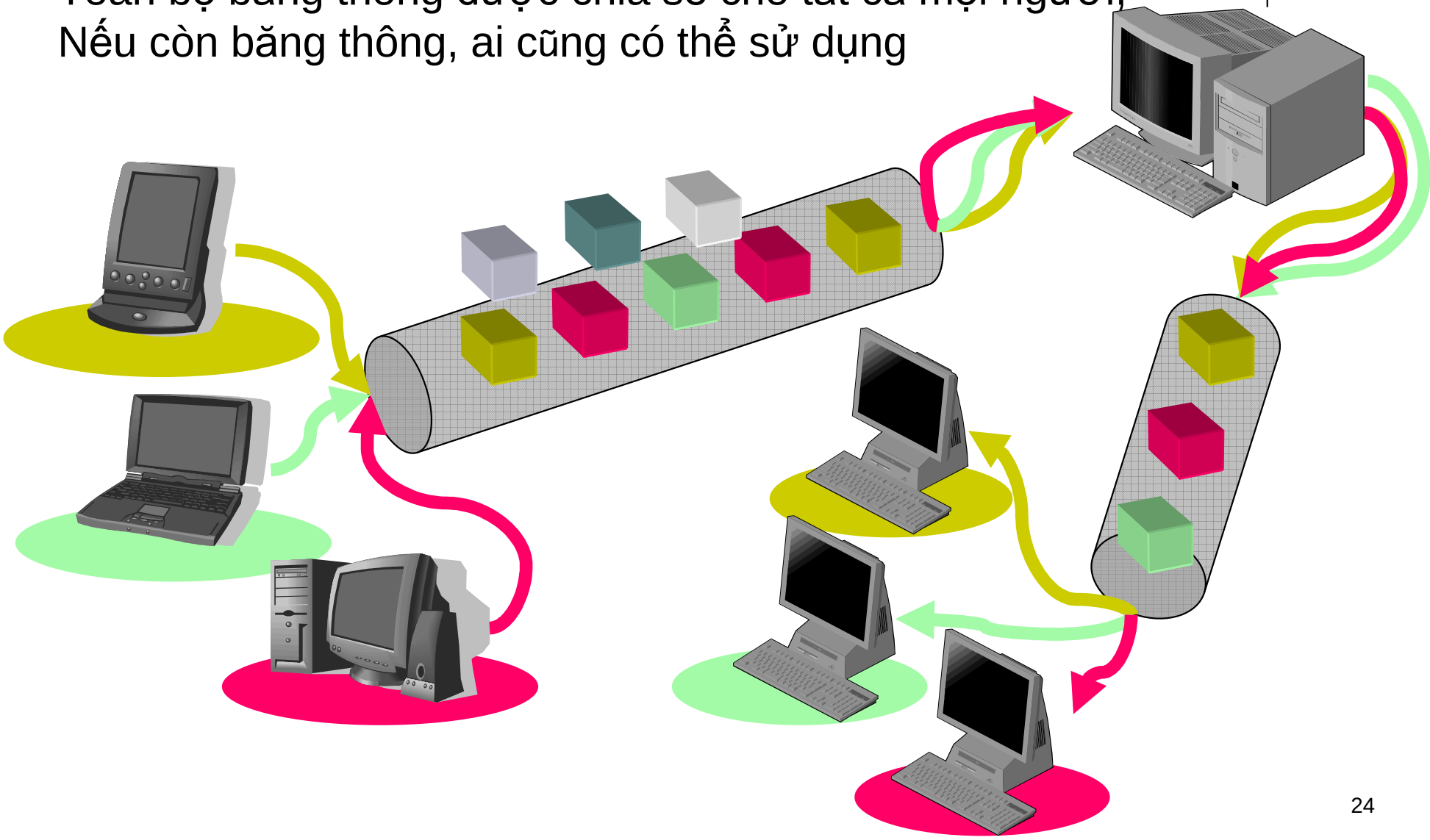
Chuyển mạch kênh

Tài nguyên được gán riêng cho mỗi kênh
Kể cả khi tài nguyên của kênh đó đangg rỗi, người khác cũng không được dùng

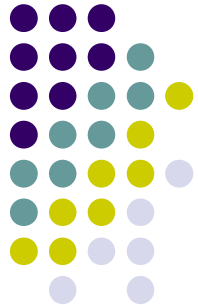


Chuyển mạch gói

Toàn bộ băng thông được chia sẻ cho tất cả mọi người,
Nếu còn băng thông, ai cũng có thể sử dụng

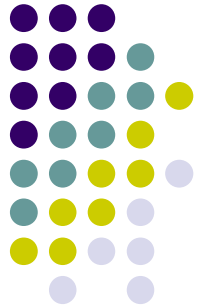


Chuyển mạch gói vs. Chuyển mạch kênh



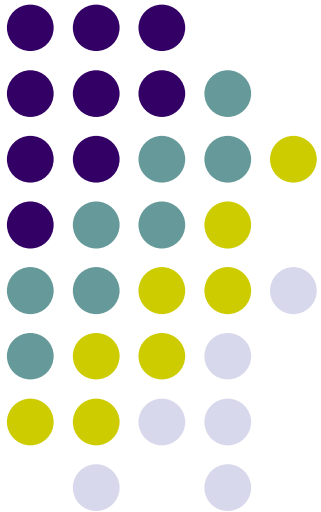
- Chuyển mạch kênh
  - Mỗi kênh chỉ dùng cho duy nhất 1 liên kết
  - Bảo đảm băng thông (cần cho các ứng dụng audio/video)
  - Lãng phí nếu liên kết đó không sử dụng hết khả năng của kênh
- Chuyển mạch gói
  - Tăng hiệu quả sử dụng băng thông
  - Tốt cho các dạng dữ liệu đến ngâu nhiên, không định trước
  - Hạn chế: Tắc nghẽn làm trễ và mất gói tin, không bảo đảm băng thông

Truyền thông hướng liên kết vs. không liên kết



- Truyền thông hướng liên kết :
  - Dữ liệu được truyền qua một liên kết đã được thiết lập
  - Ba giai đoạn: Thiết lập liên kết, truyền dữ liệu, Hủy bỏ liên kết
  - Tin cậy
- Truyền thông không liên kết
  - Không thiết lập liên kết, chỉ có giai đoạn truyền dữ liệu
  - Không tin cậy - “Best effort”

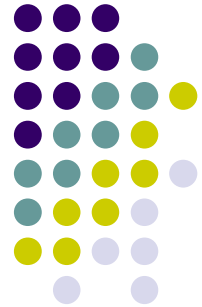
Một số tham số trong mạng



Các tham số cơ bản

- Băng thông - Bandwidth
- Thông lượng - Throughput
- Độ trễ- Delay
- Độ mất gói tin - Loss

Băng thông

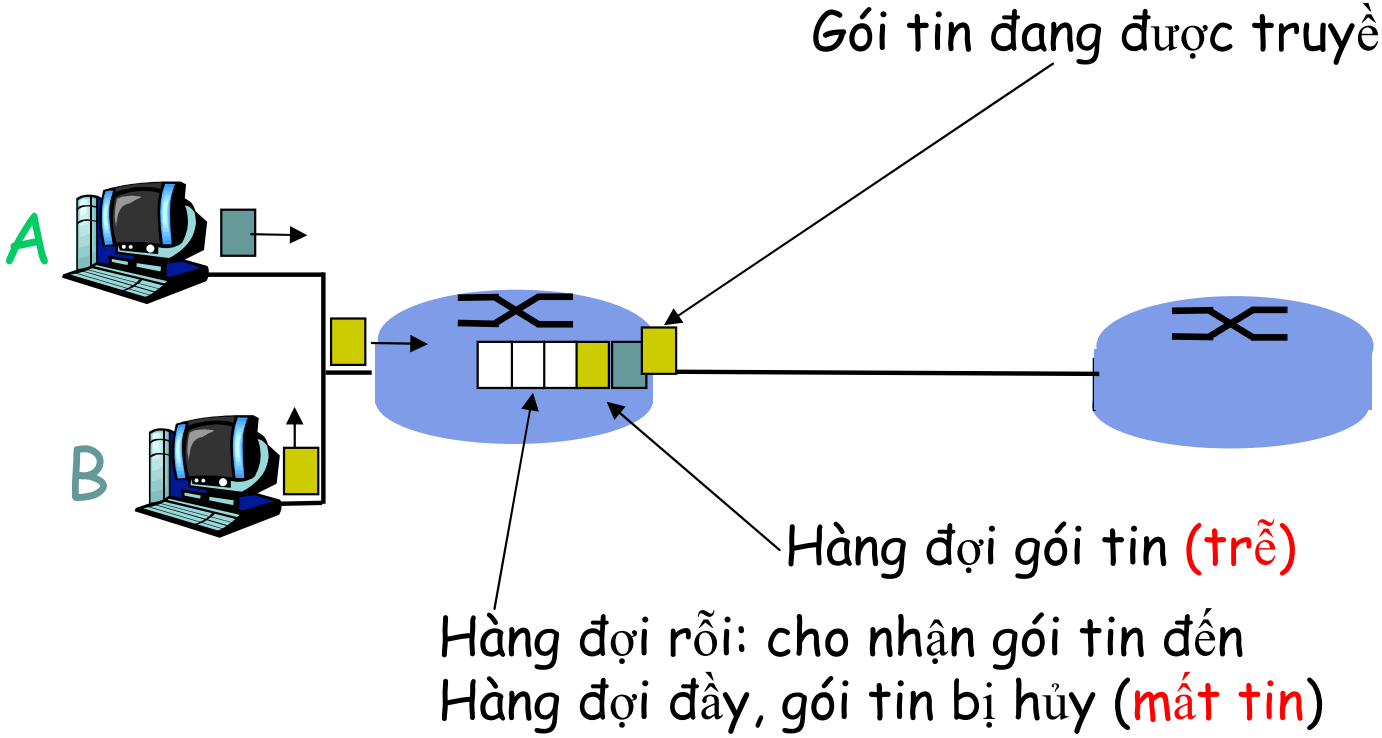


- Khái niệm
- Đơn vị
  - bps, kbps, Mbps, Gbps, Tbps
- Uplink/downlink

Vì sao có mất và trễ tin?

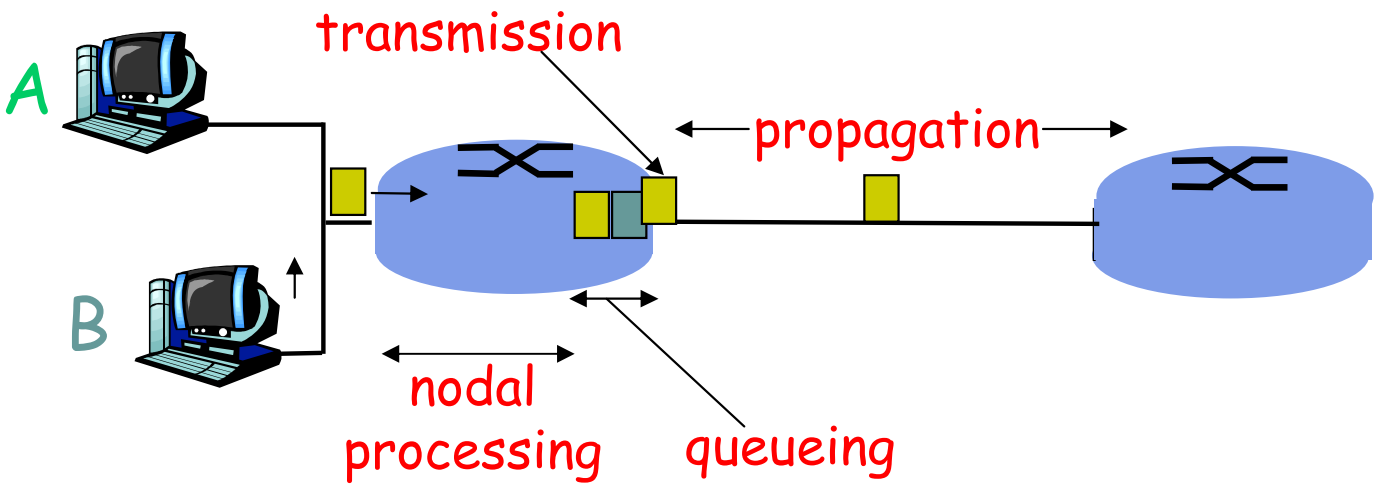
Các gói tin phải xếp hàng trong bộ định tuyến!

- Tốc độ đến của các gói tin vượt quá khả năng đường ra
- Các gói tin phải xếp hàng chờ đến lượt



4 nguyên nhân gây trễ tin

- 1. Xử lý tại nút mạng:
  - Kiểm soát lỗi
  - Tìm đường ra
- 2. Xếp hàng
  - Thời gian chờ đi ra
  - Phụ thuộc độ tắc nghẽn của router



4 nguyên nhân gây trễ tin

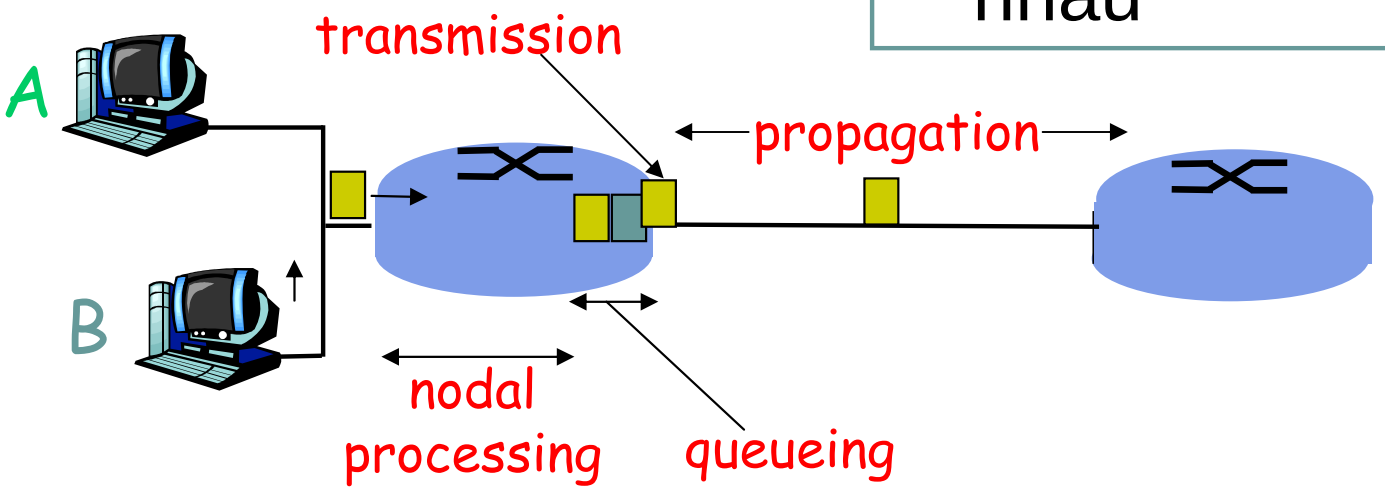
3. Trễ truyền tin:

- R= băng thông (bps)
- L= độ dài packet (bits)
- Trễ truyền tin = L/R

4. Trễ lan truyền:

- d = độ dài đường truyền
- s = tốc độ tín hiệu (~$2x10^8$ m/sec)
- Trễ lan truyền = d/s

Chú ý: s và R rất khác nhau



Tổng thời gian trễ

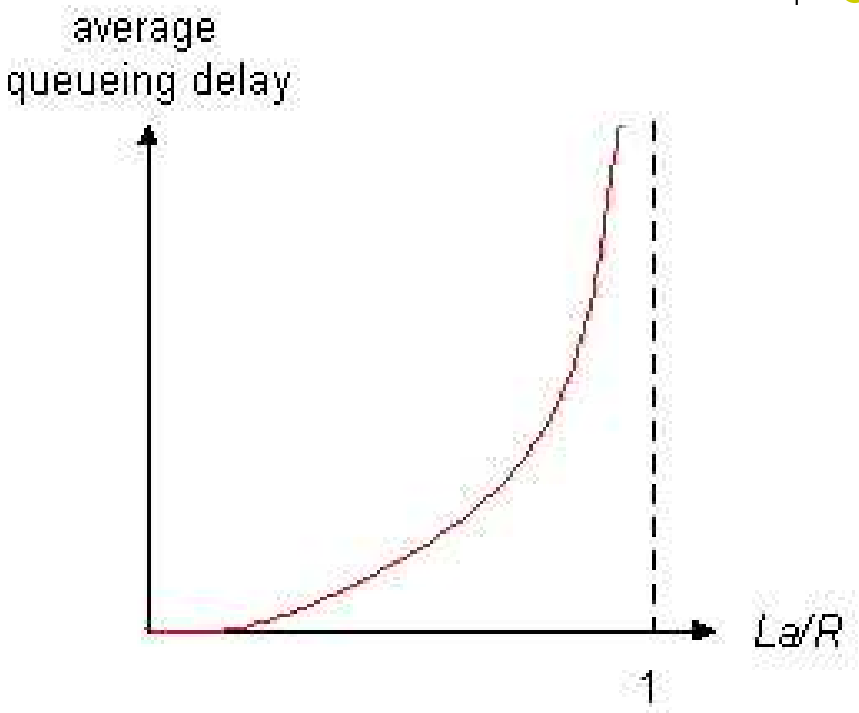
$$d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

- d_{proc} = processing delay
  - Vài microsecs hay ít hơn
- d_{queue} = queuing delay
  - Phụ thuộc vào độ tắc nghẽn
- d_{trans} = transmission delay
  - = L/R, lớn với những đường truyền tốc độ thấp
- d_{prop} = propagation delay
  - vài microsecs tới hàng trăm msecs

Trễ hàng đợi

- R= băng thông (bps)
- L= độ dài gói tin (bits)
- a= tốc độ đến của gói tin

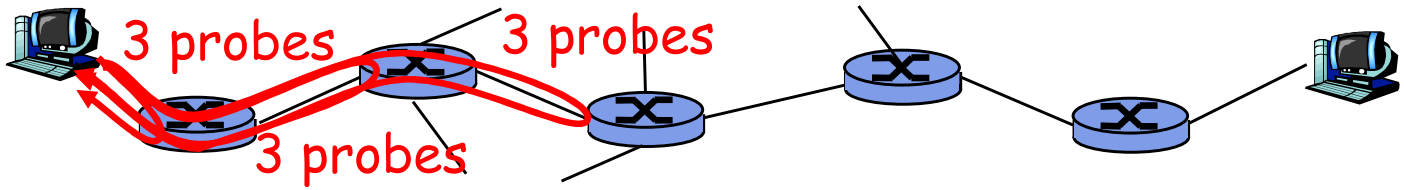
Lưu lượng đến = La/R



- La/R ~ 0: trễ hàng đợi nhỏ
- La/R -> 1: trễ lớn dần lên
- La/R > 1: quá khả năng, trễ vô cùng

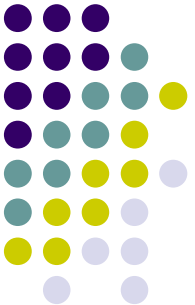
Độ trễ và đường đi thực tế trên Internet

- Làm thế nào để biết đường đi và độ trễ?
- **`Traceroute` program:** cung cấp độ trễ và đường đi end-to-end.
- For all *i:*
  - Gửi 3 gói tin tới router *i* trên đường tới đích
  - router *i* trả lại một gói tin cho người gửi
  - Bên gửi đo khoảng thời gian giữa gửi và nhận



Ví dụ

traceroute: gaia.cs.umass.edu to www.eurecom.fr



Three delay measurements from gaia.cs.umass.edu to cs-gw.cs.umass.edu

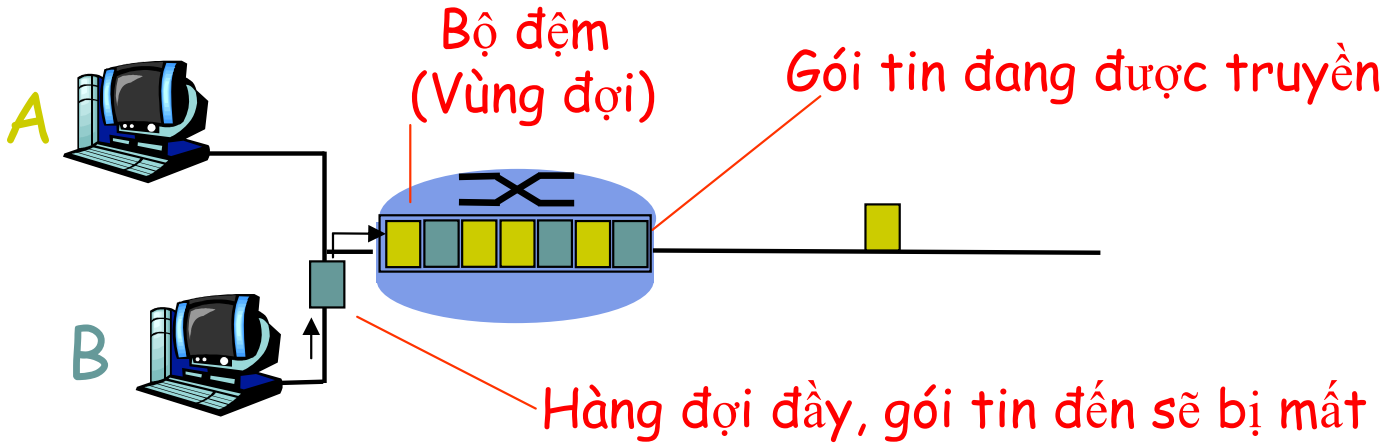
```
1  cs-gw (128.119.240.254)  1 ms  1 ms  2 ms
2  border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145)  1 ms  1 ms  2 ms
3  cht-vbns.gw.umass.edu (128.119.3.130)  6 ms 5 ms 5 ms
4  jn1-at1-0-0-19.wor.vbns.net (204.147.132.129)  16 ms 11 ms 13 ms
5  jn1-so7-0-0-0.wae.vbns.net (204.147.136.136)  21 ms 18 ms 18 ms
6  abilene-vbns.abilene.ucaid.edu (198.32.11.9)  22 ms  18 ms  22 ms
7  nycm-wash.abilene.ucaid.edu (198.32.8.46)  22 ms  22 ms  22 ms
8  62.40.103.253 (62.40.103.253)  104 ms 109 ms 106 ms
9  de2-1.de1.de.geant.net (62.40.96.129)  109 ms 102 ms 104 ms
10  de.fr1.fr.geant.net (62.40.96.50)  113 ms 121 ms 114 ms
11  renater-gw.fr1.fr.geant.net (62.40.103.54)  112 ms  114 ms  112 ms
12  nio-n2.cssi.renater.fr (193.51.206.13)  111 ms  114 ms  116 ms
13  nice.cssi.renater.fr (195.220.98.102)  123 ms  125 ms  124 ms
14  r3t2-nice.cssi.renater.fr (195.220.98.110)  126 ms  126 ms  124 ms
15  eurecom-valbonne.r3t2.ft.net (193.48.50.54)  135 ms  128 ms  133 ms
16  194.214.211.25 (194.214.211.25)  126 ms  128 ms  126 ms
17  * * *
18  * * *
19  fantasia.eurecom.fr (193.55.113.142)  132 ms  128 ms  136 ms
```

trans-oceanic link

* means no response (probe lost, router not replying)

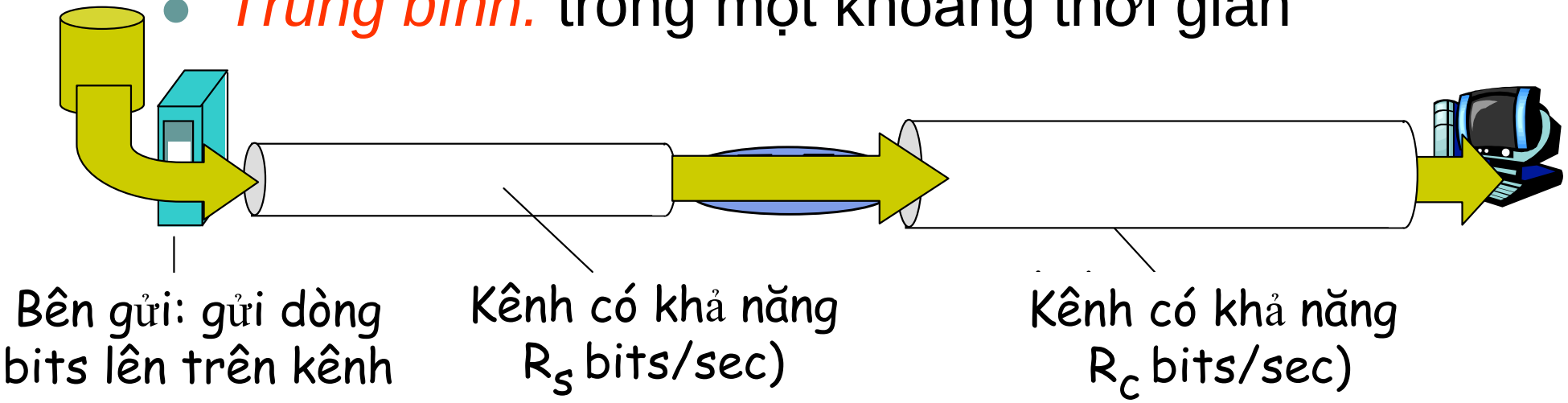
Mất tin (loss)

- Hàng đợi (vùng đệm) của mỗi đường truyền có kích thước giới hạn
- Gói tin nào tới hàng đợi đầy sẽ bị mất
- Gói tin bị mất có thể được truyền lại hoặc không.



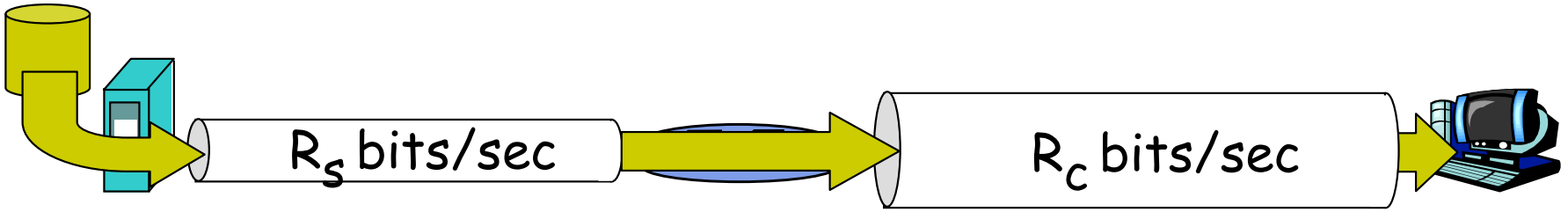
Thông lượng

- *Thông lượng:* tốc độ (đơn vị bits/sec) mà tại đó các bits được truyền giữa bên gửi/bên nhận
  - *Tức thời:* tốc độ tại một thời điểm
  - *Trung bình:* trong một khoảng thời gian

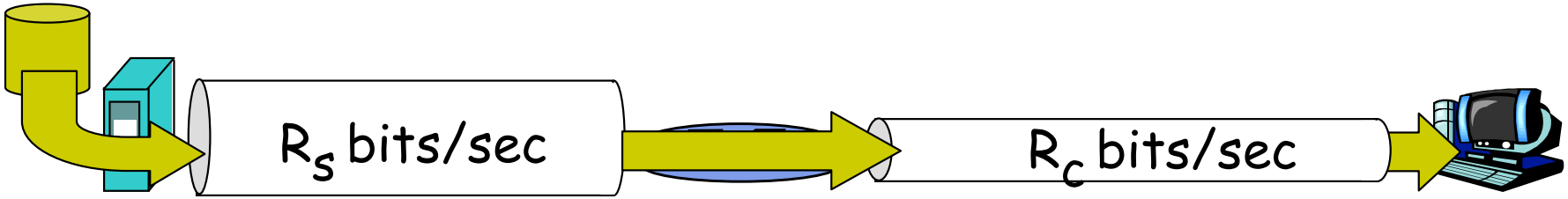


Thông lượng

- $R_s < R_c$ Thông lượng trung bình?



- $R_s > R_c$ Thông lượng trung bình?

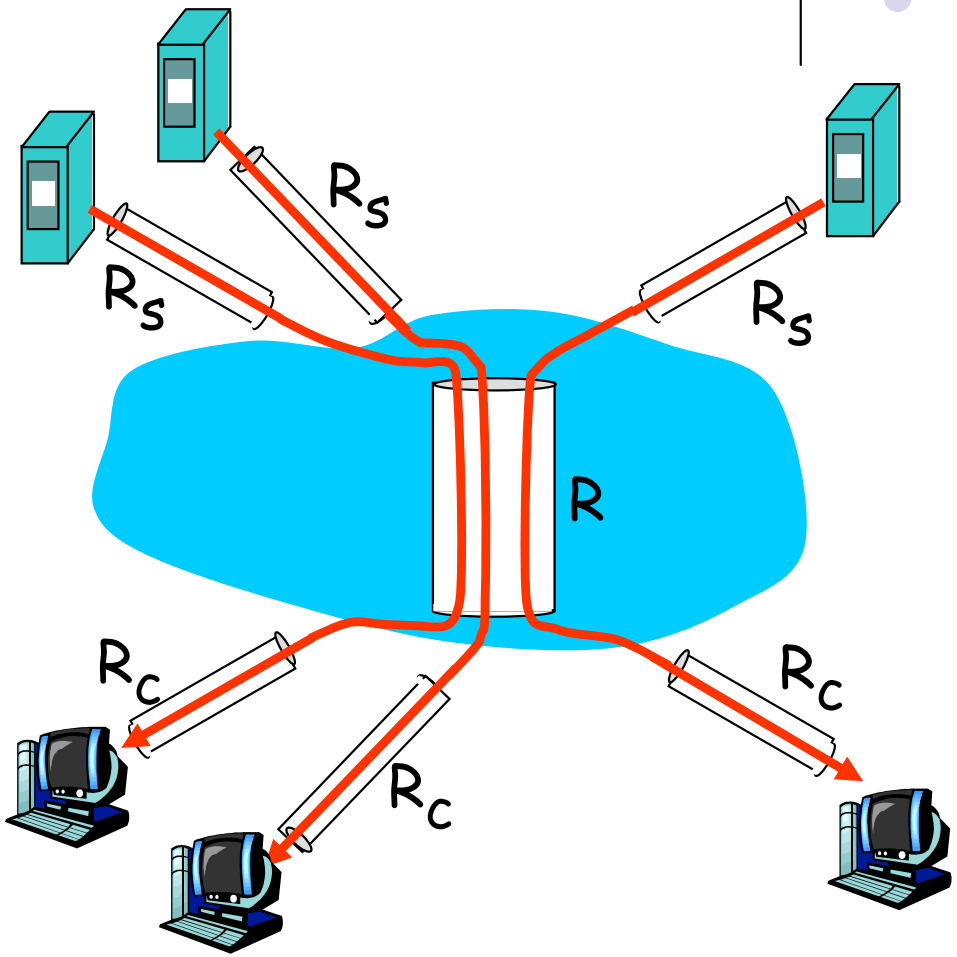


Nút thắt cổ chai

Đường truyền mà tại đó giới hạn toàn bộ băng thông của tuyến

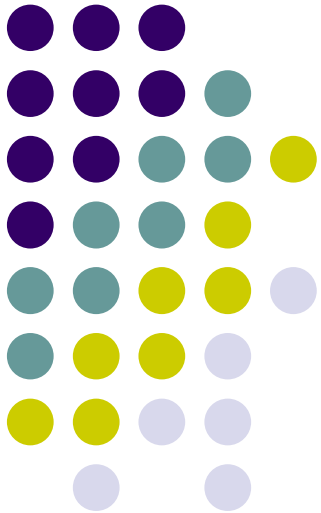
Thông lượng: Ví dụ trên Internet

- Thông lượng của mỗi kết nối min(R_c,R_s,R/10)
- Thực tế: R_c hoặc R_s thường xuyên bị thắt cổ "chai"



10 liên kết chia sẻ 1 đường R bits/sec

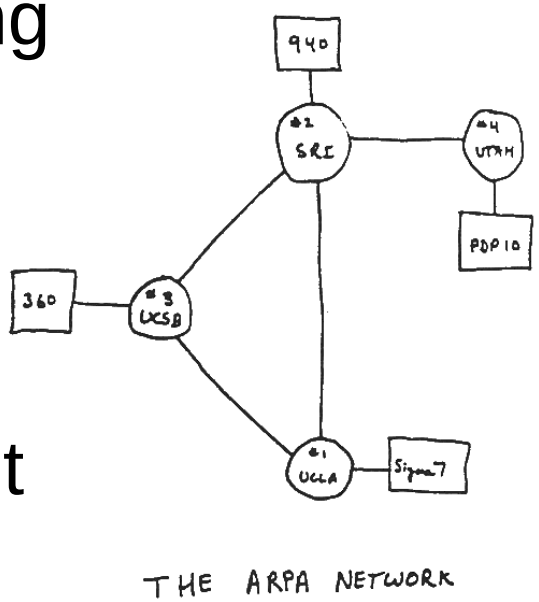
Lược sử mạng & Internet



Thời kỳ đầu

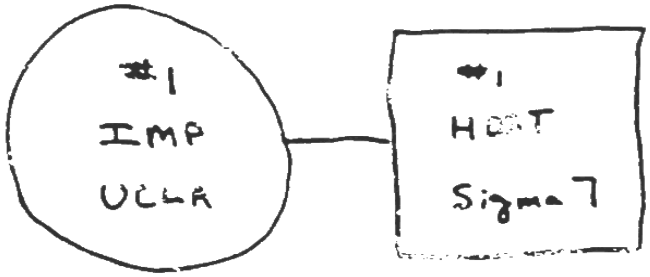
1961-1972: Các nguyên lý mạng chuyển mạch gói

- 1960s: Mạng điện thoại & sự phát triển của máy tính
- 1961: Kleinrock – Lý thuyết hàng đợi, hiệu quả của chuyển mạch gói
- 1964: Baran – mạng chuyển mạch gói
- 1967: ARPAnet được phê duyệt (Advanced Research Projects Agency)



THE ARPA NETWORK

Nguồn gốc Internet



- Bắt đầu từ một thí nghiệm của dự án của ARPA
- Một liên kết giữa hai nút mạng (IMP tại UCLA và IMP tại SRI).

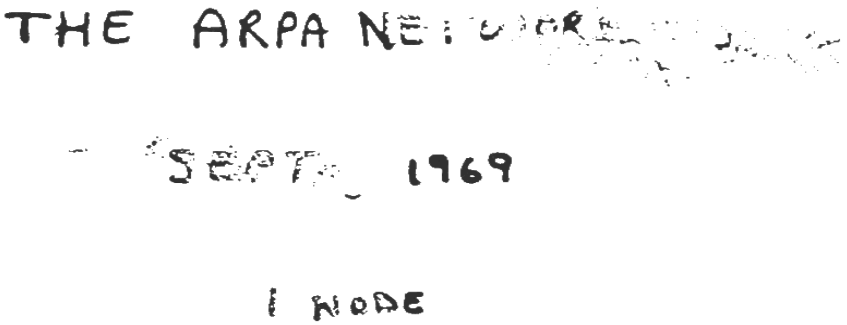
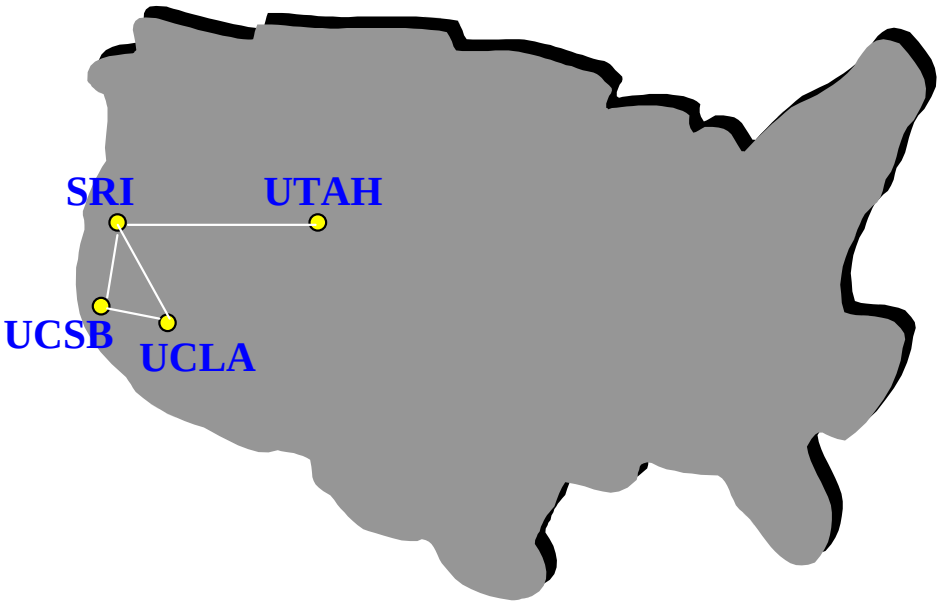


FIGURE 6.1 Drawing of September 1969 (Courtesy of Alex McKenzie)

ARPA: Advanced Research Project Agency
UCLA: University California Los Angeles
SRI: Stanford Research Institute
IMP: Interface Message Processor

Source: http://www.cybergeography.org/atlas/historical.html

3 tháng sau, 12/1969



Một mạng hoàn chỉnh với 4 nút, 56kbps

UCSB:University of California, Santa Barbara
UTAH:University of Utah

source: http://www.cybergeography.org/atlas/historical.html

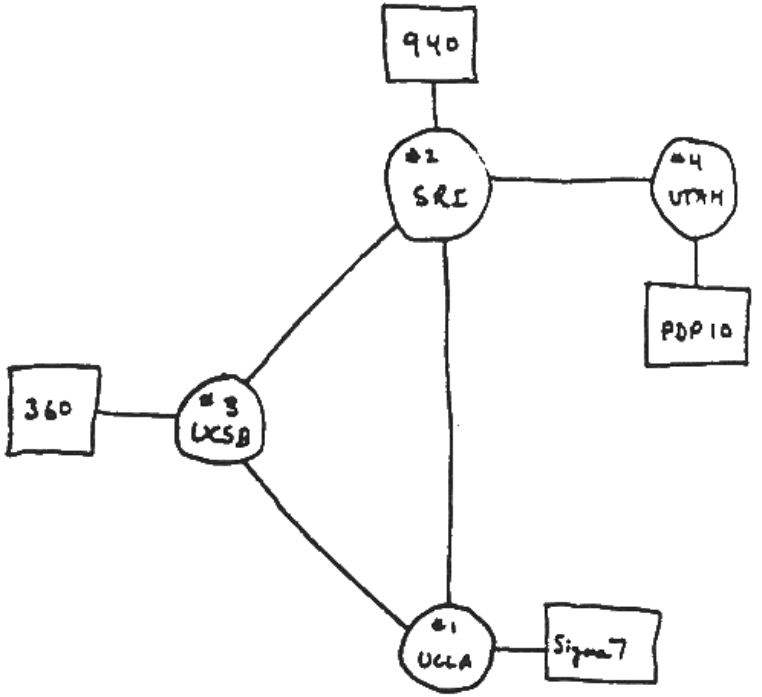
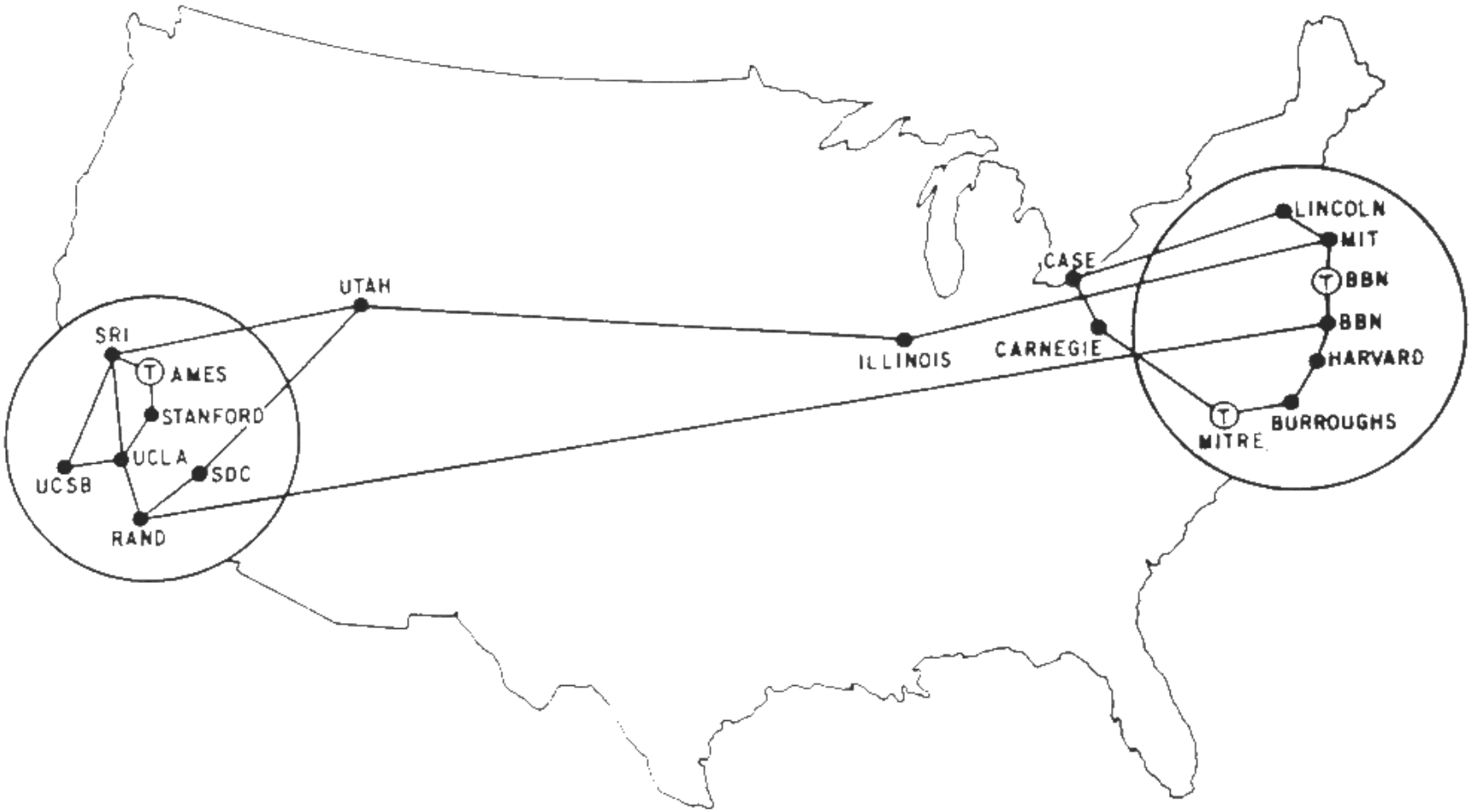


FIGURE 6.2 Drawing of 4 Node Network (Courtesy of Alex McKenzie)

ARPANET thời kỳ đầu, 1971

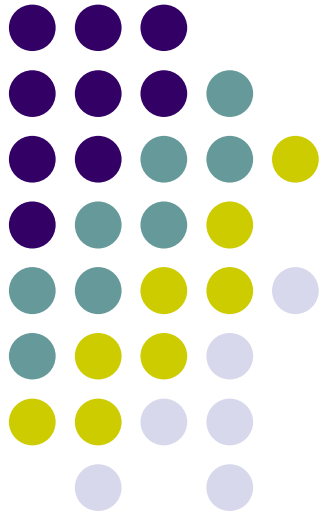


MAP 4 September 1971

Source: http://www.cybergeography.org/atlas/historical.html

Mạng phát triển với tốc độ thêm mỗi nút một tháng

Thập niên 70: Kết nối liên mạng, kiến trúc mạng mới và các mạng riêng



Sự mở rộng của ARPANET, 1974

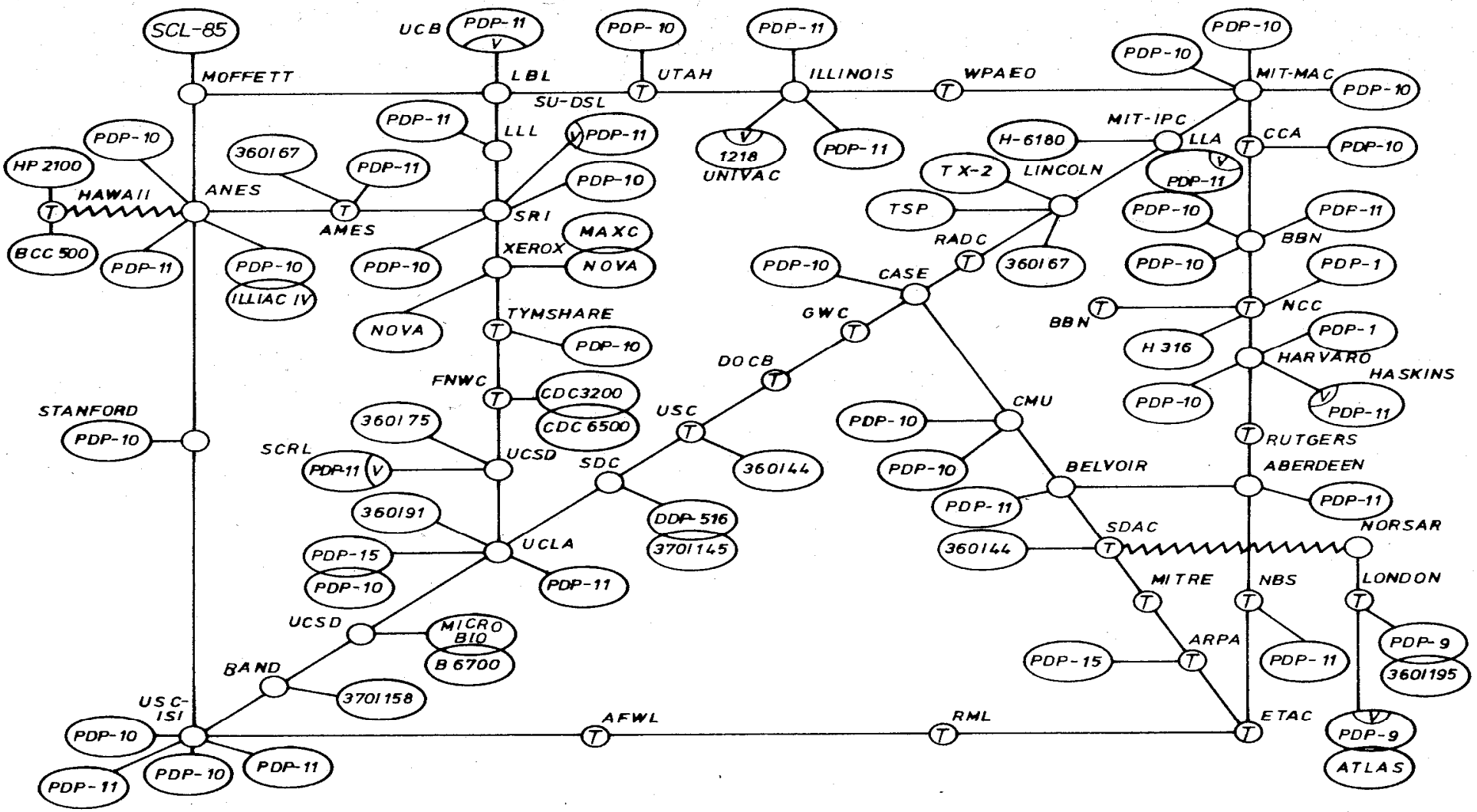
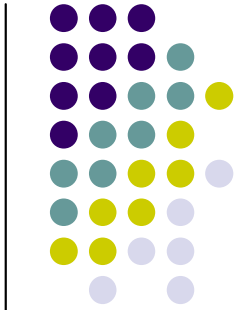


Abb. 4 ARPA NETwork, topologische Karte. Stand Juni 1974.

source:
http://www.cybergeography.org/atlas/historical.html

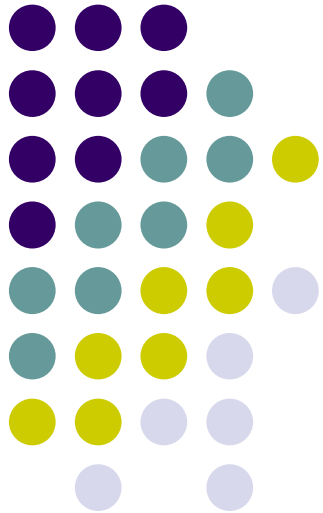
Lưu lượng mỗi ngày vượt quá 3.000.000 gói tin

Thập niên 70



- Từ đầu 1970 xuất hiện các mạng riêng:
  - ALOHAnet tại Hawaii
  - DECnet, IBM SNA, XNA
- 1974: Cerf & Kahn – nguyên lý kết nối các hệ thống mở (Turing Awards)
- 1976: Ethernet, Xerox PARC
- Cuối 1970: ATM

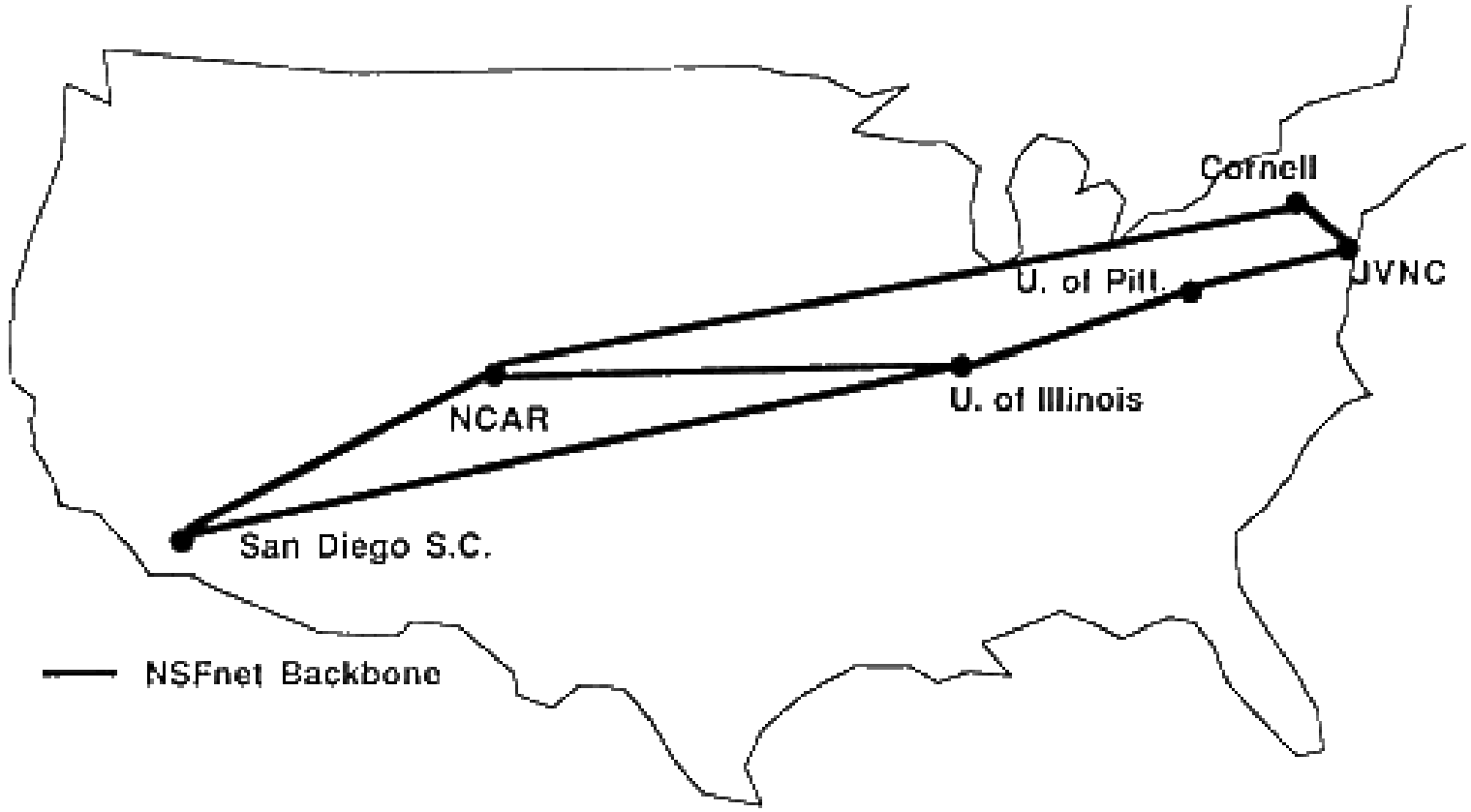
Thập niên 80: Các giao thức mới, kết nối thêm mạng mới



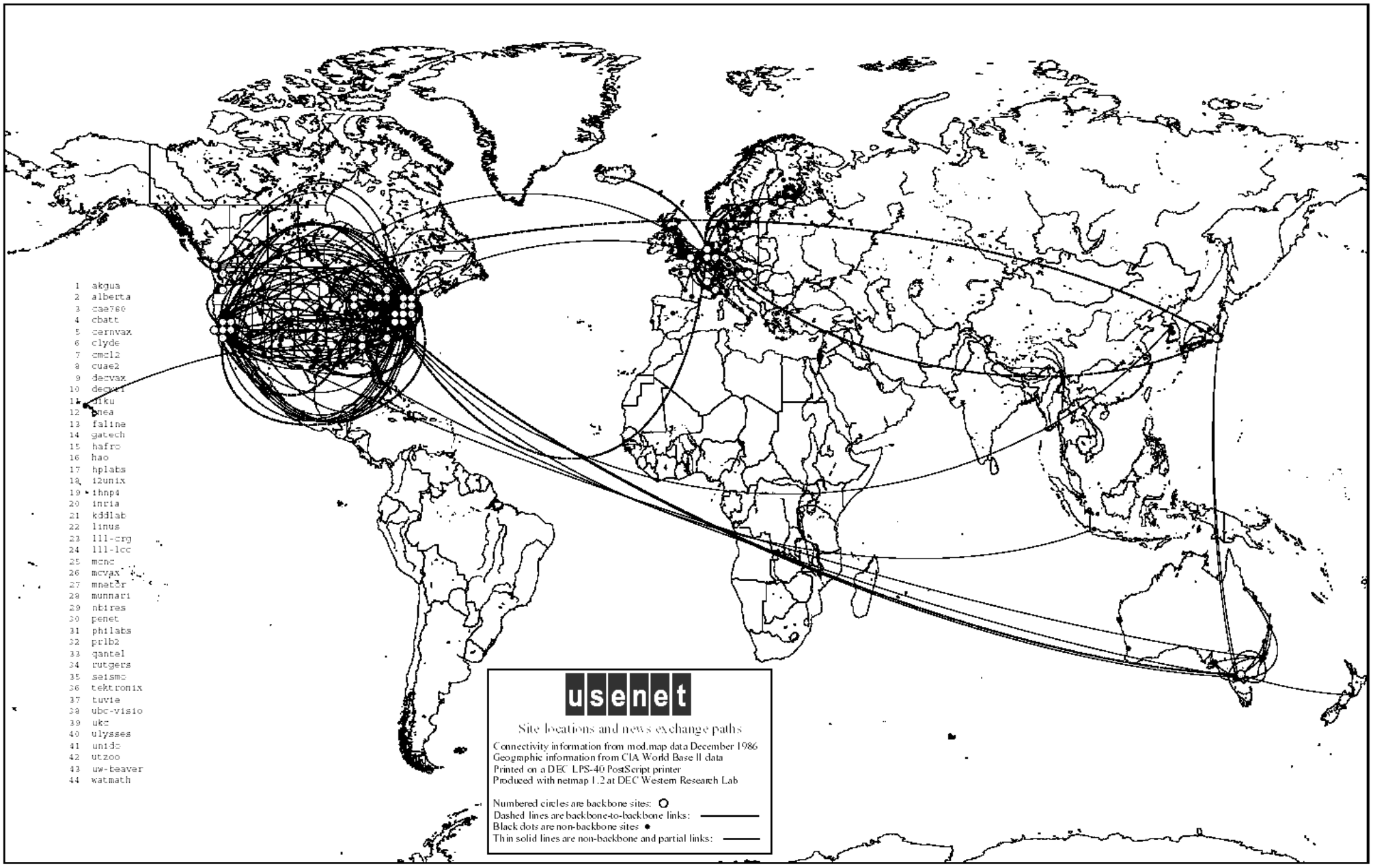
1981: Xây dựng mạng NSFNET

NSF: National Science Foundation

Phục vụ cho nghiên cứu khoa học, do sự quá tải của ARPANET

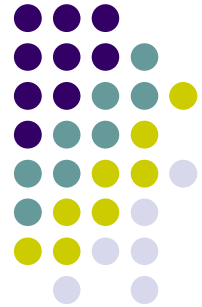


1986: Nối kết USENET& NSFNET



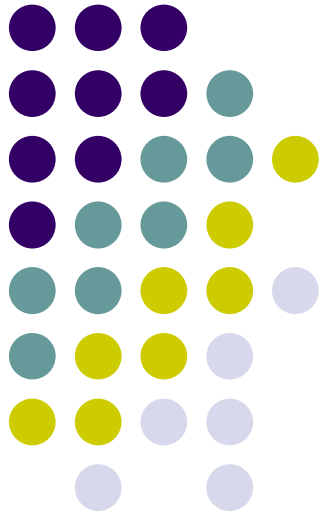
Source: http://www.cybergeography.org/atlas/historical.html

Thêm nhiều mạng và giao thức mới

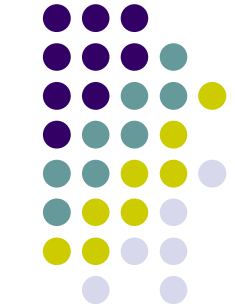


- Thêm nhiều mạng mới nối vào: MFENET, HEPNET (Dept. Energy), SPAN (NASA), BITnet, CSnet, NSFnet, Minitel …
- TCP/IP được chuẩn hóa và phổ biến vào 1980
- Berkeley tích hợp TCP/IP vào BSD Unix
- Dịch vụ: FTP, Mail, DNS …

Thập niên 90: Web và thương mại hóa Internet



Thập niên 90

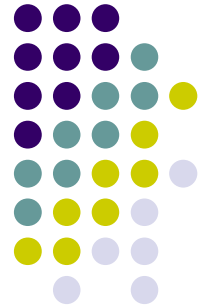


- Đầu 90: ARPAnet chỉ là một phần của Internet
- Đầu 90: Web
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, Netscape
- Cuối 90: Thương mại hóa Internet

Cuối 1990's – 2000's:

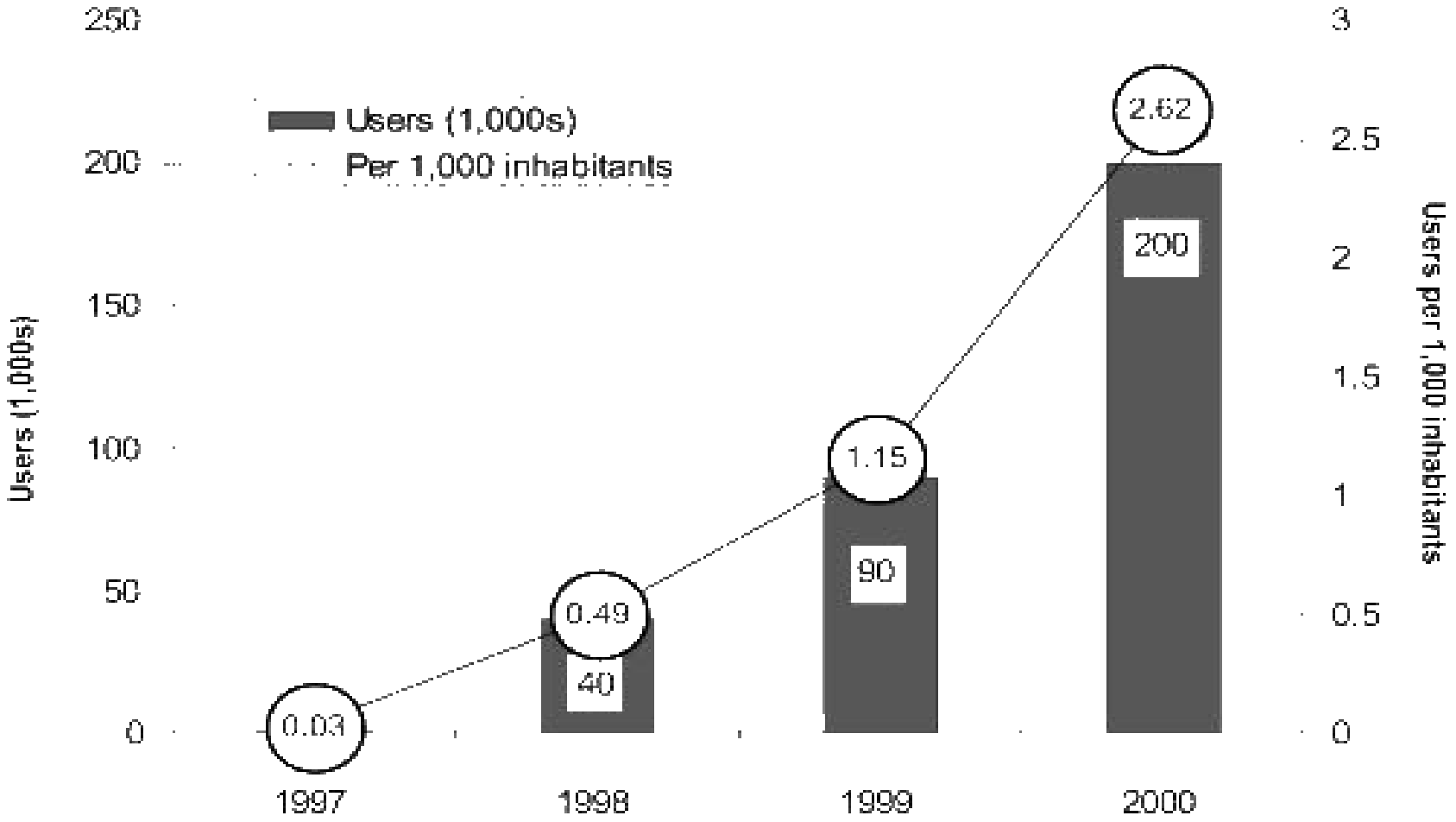
- Nhiều ứng dụng mới: chat, chia sẻ file P2P…
- E-commerce, Yahoo, Amazon, Google…
- > 50 triệu máy trạm, > 100 triệu NSD
- Vấn đề an toàn an ninh thông tin!
  - Internet dành cho tất cả mọi người
  - Tất cả các dịch vụ phải quan tâm tới vấn đề này

Lược sử Internet Việt Nam



- 1991: Nỗ lực kết nối Internet không thành. ☹ (Vì một lý do nào đó)
- 1996: Giải quyết các cản trở, chuẩn bị hạ tầng Internet
  - ISP: VNPT
  - 64kbps, 1 đường kết nối quốc tế, một số NSD
- 1997: Việt Nam chính thức kết nối Internet
  - 1 IXP: VNPT
  - 4 ISP: VNPT, Netnam (IOT), FPT, SPT
- 2007: "Mười năm Internet Việt Nam"
  - 20 ISPs, 4 IXPs
  - 19 triệu NSD, 22.04% dân số

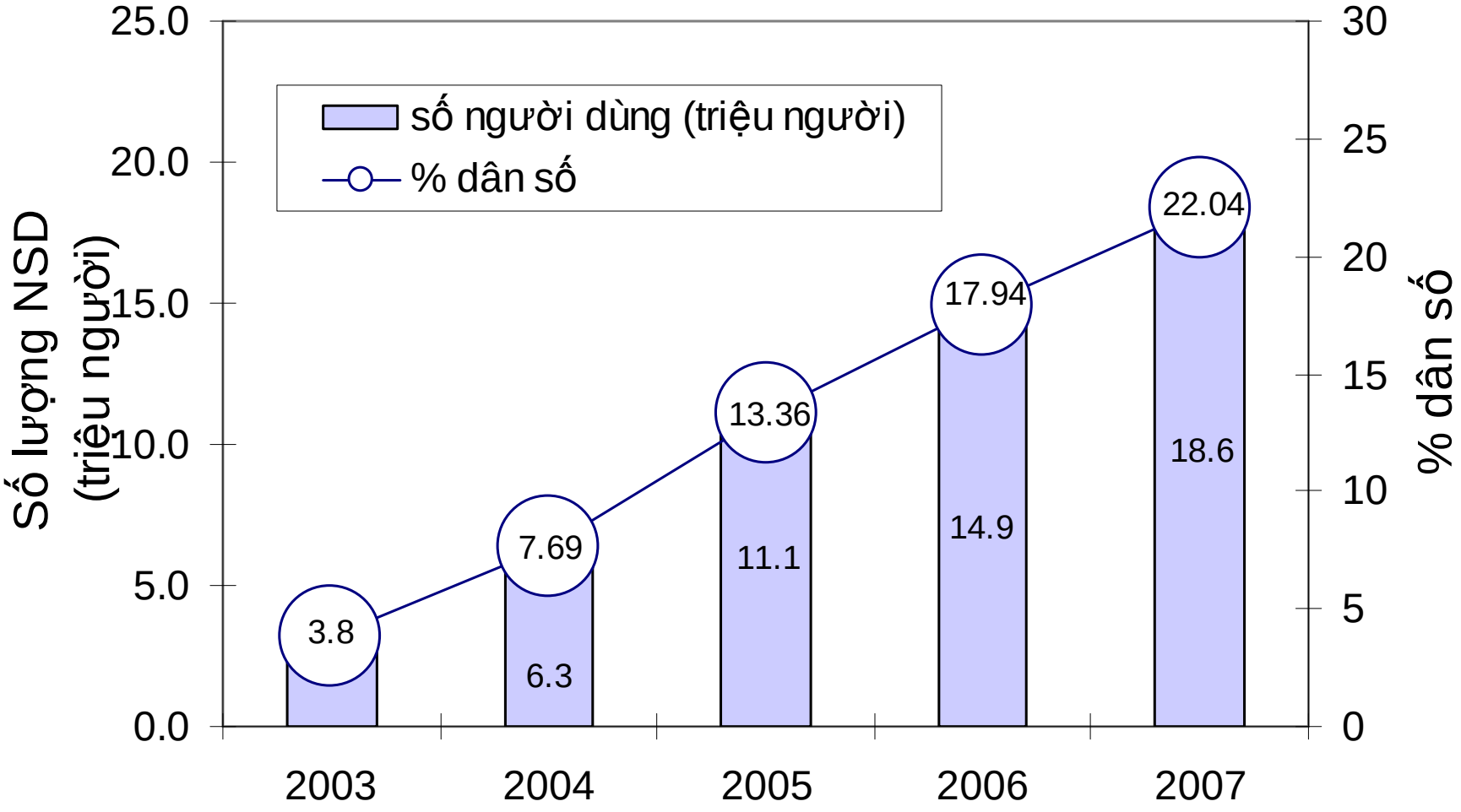
Phát triển Internet ở VN



Ước tính số người dùng bằng hai lần số thuê bao

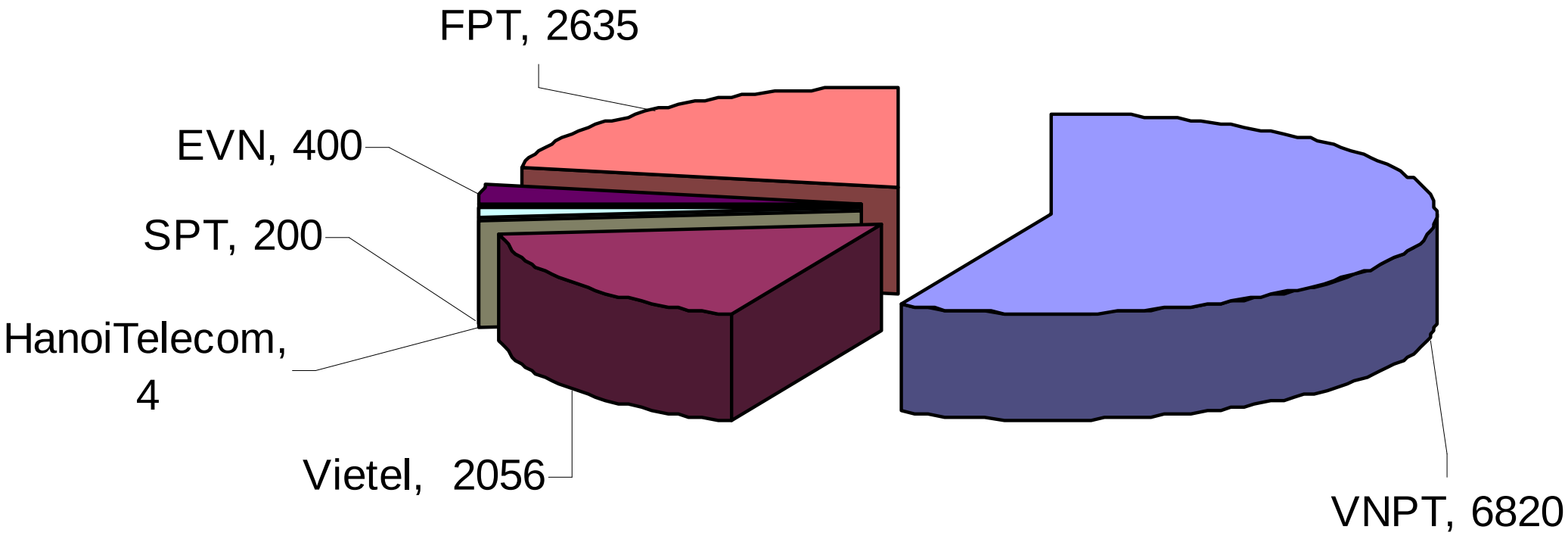
Source: Vietnam Internet Case Study, http://www.itu.int/asean2001/reports/material/VNM%20CS.pdf

Thống kê gần đây



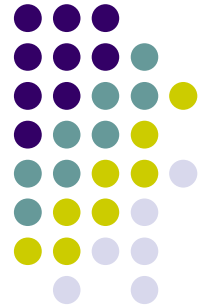
Source: *Vnnic*, http://www.thongkeinternet.vn

Băng thông kết nối đi quốc tế (Mbps), Q.3 2007



Tổng cộng: 12115.0 Mbps

Internet những năm 2000s: Tương lai là của các bạn



- Ứng dụng và công nghệ mới
  - Youtube, Skype, Bittorrent, Video & VoIP...
  - Mạng không dây, mạng quang học, thông tin di động
  - ….
- Internet sẽ tiếp tục cải tiến dịch vụ và biến đổi không ngừng
  - Mang lại sự thuận tiện cho mọi người
  - Các bạn (sinh viên CNTT) sẽ làm được điều đó!

Tóm tắt

- Giới thiệu môn học
- Lược sử Internet
- Khái niệm mạng máy tính
- Kiến trúc mạng
  - Topology
  - Protocol
- Mô hình truyền thông
  - Chuyển mạch kênh vs. chuyển mạch gói
  - Không liên kết vs. Hướng liên kết
- Các tham số cơ bản

Tuần tới…

- Kiến trúc phân tầng
- Mô hình tham chiếu OSI
- Địa chỉ IP, MAC, số hiệu cổng
- DNS và dịch vụ tên miền

TRƯỜNG ĐẠI HỌC BÁCH KHOA ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN

——— Â ———

GIÁO TRÌNH MÔN HỌC

MẠNG MÁY TÍNH

Ths. NGUYỄN TẤN KHÔI

(Lưu hành nội bộ)
Đà Nẵng – 2004

MỤC LỤC

Chương 1 MỞ ĐẦU 1

1.1 Giới thiệu 1

1.2 Phân loại mạng 2

1.2.1 Dựa theo khoảng cách địa lý 2

1.2.2 Dựa theo cấu trúc mạng 2

1.2.3 Theo phương pháp chuyển mạch 3

1.3 Kiến trúc phân tầng và chuẩn hoá mạng 5

1.3.1 Các tổ chức chuẩn hoá mạng 5

1.3.2 Kiến trúc phân tầng 6

1.4 Mô hình OSI 7

1.4.1 Kiến trúc của mô hình OSI 7

1.4.2 Sự ghép nối giữa các mức 8

1.4.3 Chức năng của mỗi tầng 9

1.4.4 Các giao thức chuẩn của OSI 11

1.5 Hệ điều hành mạng 12

1.6 Mạng Internet 13

1.6.1 Lịch sử ra đời và phát triển 13

1.6.2 Cấu trúc của mạng Internet 14

1.6.3 Các kiến trúc khác 15

Chương 2 TẦNG VẬT LÝ 16

2.1 Môi trường truyền tin 16

2.1.1 Phương tiện truyền 16

2.1.2 Các thông số cơ bản của môi trường truyền tin 19

2.2 Chuẩn giao diện 19

2.2.1 Modem 19

2.2.2 DTE và DCE 21

2.2.3 Chuẩn RS-232C 21

Chương 3 TẦNG LIÊN KẾT DỮ LIỆU 22

3.1 Chức năng 22

3.2 Các vấn đề của tầng liên kết dữ liệu 22

3.2.1 Cung cấp dịch vụ cho tầng mạng 22

3.2.2 Khung tin - Nhận biết gói tin 23

3.2.3 Kiểm tra lỗi 23

3.2.4 Điều khiển luồng dữ liệu 23
3.2.5 Quản lý liên kết 24
3.2.6 Nén dữ liệu khi truyền 24
3.3 Phát hiện và hiệu chỉnh lỗi 24
3.3.1 Phương pháp bit chẵn lẻ (Parity) 25
3.3.2 Tính theo đa thức chuẩn 25
3.3.3 Mã sửa sai 26
3.4 Thủ tục liên kết dữ liệu cơ bản 27
3.4.1 Giao thức đơn công với kênh có lỗi 28
3.5 Điều khiển dòng truyền 28
3.5.1 Cơ chế cửa sổ 29
3.5.2 Trao đổi bản tin với cửa sổ 1 bit 30
3.5.3 Vận chuyển liên tục 31
3.6 Các giao thức của tầng Liên kết dữ liệu 33
3.6.1 Giao thức BSC 33
3.6.2 Giao thức HDLC 34
Chương 4 MẠNG CỤC BỘ 37
4.1 Các cấu hình của mạng LAN 37
4.1.1 Mạng dạng hình sao (Star Topology) 37
4.1.2 Mạng hình tuyến (Bus Topology) 38
4.1.3 Mạng dạng vòng (Ring Topology) 38
4.1.4 Mạng dạng kết hợp 39
4.2 Các giao thức điều khiển truy nhập đường truyền 39
4.2.1 Phương pháp CSMA 40
4.2.2 Phương pháp CSMA/CD 41
4.2.3 Điều khiển truy nhập bus với thẻ bài 41
4.2.4 Điều khiển truy nhập vòng với thẻ bài 43
4.3 Chuẩn hóa mạng cục bộ 44
4.3.1 Chuẩn Ethernet 46
Chương 5 TẦNG MẠNG 47
5.1 Các vấn đề của tầng mạng 47
5.1.1 Định địa chỉ cho tầng mạng 47
5.1.2 Dịch vụ cung cấp cho tầng giao vận 48
5.1.3 Tổ chức các kênh truyền tin trong tầng mạng 49
5.1.4 Tìm đường đi trong mạng 50
5.1.5 Tắc nghẽn trong mạng 51

5.2 Kết nối liên mạng 51
5.2.1 Các thiết bị dùng để kết nối liên mạng 52
5.3 Giao thức liên mạng IP 58
5.3.1 Cấu trúc khung tin IP 59
5.3.2 Địa chỉ IP 64
5.4 Phân chia mạng con 66
5.5 Hoạt động của giao thức IP 67
5.6 Các giao thức liên quan đến IP 68
5.6.1 Giao thức phân giải địa chỉ ARP 68
5.6.2 Giao thức RARP (Reverse Address Resolution Protocol) 71
5.6.3 Giao thức ICMP 71
5.7 Phiên bản IPv6 76
5.7.1 Khung tin IPng v6 77
5.8 Định tuyến trên Internet 77
5.8.1 Bảng chọn đường 77
5.8.2 Xây dựng bảng chọn đường cho các Router/Gateway 78
5.9 Mạng X.25 80
5.9.1 Cơ sở kỹ thuật 80
5.10 Kỹ thuật FRAME RELAY 82
5.10.1 Khuôn dạng gói dữ liệu Frame-Relay 82
Chương 6 TẦNG GIAO VẬN 84
6.1 Các vấn đề của tầng giao vận 84
6.1.1 Cung cấp dịch vụ cho tầng phiên 84
6.1.2 Chất lượng dịch vụ QoS 86
6.1.3 Các lớp giao thức của tầng giao vận 87
6.1.4 Thủ tục giao vận trên X. 25 90
Chương 7 HỌ GIAO THỨC TCP/IP 91
7.1 Mô hình TCP/IP 91
7.2 Giao thức TCP 93
7.2.1 Khuôn dạng gói tin TCP 94
7.2.2 Quá trình nối-tách 96
7.2.3 Quá trình trao đổi dữ liệu 97
7.2.4 Thứ tự thực hiện ứng dụng TCP/IP 97
7.3 Giao thức UDP 100
7.4 Cổng và Socket 101

7.4.1 Số hiệu cổng 101
7.4.2 Socket 101
7.5 Mô hình giao tiếp Client/Server 103
7.5.1 Quá trình trao đổi dữ liệu dùng Stream Socket 103
7.5.2 Quá trình trao đổi dữ liệu dùng Datagram Socket 104
7.5.3 Ví dụ chương trình client/server 105
Chương 8 TẦNG PHIÊN 108
8.1 Dịch vụ OSI cho tầng Phiên 108
8.1.1 Cung cấp cho người sử dụng dịch vụ tầng phiên (SS-user) 108
8.1.2 Điều khiển trao đổi dữ liệu 109
8.1.3 Điều hành phiên làm việc 110
8.1.4 Liên kết phiên 111
8.2 Giao thức chuẩn tầng phiên 111
8.2.1 Các loại SPDU, các tham số và chức năng 112
Chương 9 TẦNG TRÌNH DIỄN 114
9.1 Vai trò và chức năng 114
9.1.1 Phiên dịch dữ liệu 116
9.2 Dịch vụ OSI cho tầng trình diễn 116
9.3 Giao thức chuẩn tầng trình diễn 117
9.3.1 Các chuẩn khác cho tầng trình diễn 118
Chương 10 TẦNG ỨNG DING 119
10.1 An toàn thông tin trên mạng 119
10.1.1 Các chiến lược an toàn hệ thống 119
10.1.2 An toàn thông tin bằng mã hóa 120
10.2 CáC phương pháp mã hóa dữ liệu 122
10.2.1 Phương pháp hoán vị 122
10.2.2 Phương pháp thay thế 123
10.2.3 Phương pháp mã hóa chuẩn DES 124
10.2.4 Phương pháp mã hoá khoá công khai 128
10.3 Cơ chế bảo vệ bằng firewall 132
10.3.1 Các loại firewall và cơ chế hoạt động 134
10.4 Hệ thống tên miền DNS (Domain Name System) 137
10.4.1 Không gian tên miền DNS 138
10.4.2 Máy chủ quản lý tên 140
10.4.3 Chương trình phân giải tên 140

10.5 Hệ quản trị mạng 140
10.5.1 Hệ bị quản trị 141
10.5.2 Cơ sở dữ liệu chứa thông tin quản trị mạng 141
10.6 Dịch vụ thư điện tử 142
10.6.1 Giao thức SMTP 143
10.6.2 MIME 147
10.6.3 Giao thức POP 151
10.7 Dịch vụ truy cập từ xa - TELNET 154
10.7.2 Dịch vụ truyền tập tin FTP 156
10.7.3 UserNEWS 162
10.7.4 WORLD-WIDE-WEB 163

Chương 1

MỞ ĐẦU

1.1 Giới thiệu

Mạng máy tính là tập hợp nhiều máy tính điện tử và các thiết bị đầu cuối được kết nối với nhau bằng các thiết bị liên lạc nhằm trao đổi thông tin, cùng chia sẻ pcứng, phần mềm và dữ liệu với nhau

Mạng máy tính bao gồm phần cứng, các giao thức và các phần mềm mạng.

Khi nghiên cứu về mạng máy tính, các vấn đề quan trọng được xem xét là giao thức mạng, cấu hình kết nối của mạng, và các dịch vụ trên mạng.

Mạng máy tính có những công dụng như sau :

1. *Tập trung tài nguyên tại một số máy và chia sẻ cho nhiều máy khác*

- Nhiều người có thể dùng chung một phần mềm tiện ích.
- Dữ liệu được quản lý tập trung nên an toàn hơn, trao đổi giữa những người sử dụng thuận lợi hơn, nhanh chóng hơn.
- Mạng máy tính cho phép người lập trình ở một trung tâm máy tính này có thể sử dụng các chương trình tiện ích của một trung tâm máy tính khác đang rỗi, sẽ làm tăng hiệu quả kinh tế của hệ thống.

2. *Khắc phục sự trở ngại về khoảng cách địa lý.*
3. *Tăng chất lượng và hiệu quả khai thác thông tin.*
4. *Cho phép thực hiện những ứng dụng tin học phân tán*
5. *Độ an toàn tin cậy của hệ thống tăng lên nhờ khả năng thay thế khi có sự cố với máy có sự cố :* An toàn cho dữ liệu và phần mềm vì phần mềm mạng sẽ khoá các tập tin khi có những người không đủ quyền hạn truy xuất các tập tin và thư mục đó.
6. *Phát triển các công nghệ trên mạng:* Người sử dụng có thể trao đổi thông tin với nhau dễ dàng và sử dụng hệ mạng như là một công cụ để phổ biến tin tức, thông báo về một chính sách mới, về nội dung buổi họp, về các thông tin kinh tế khác như giá cả thị trường, tin rao vặt (muốn bán hoặc muốn mua một cái gì đó), hoặc sắp xếp thời khoá biểu của mình chen lẫn với thời khoá biểu của những người khác , . . .

1.2 Phân loại mạng

1.2.1 Dựa theo khoảng cách địa lý

Mạng máy tính có thể phân bổ trên một khu vực nhất định hoặc có thể trong một quốc gia hay toàn cầu. Dựa vào phạm vi phân bố, người ta có thể phân ra các loại mạng như sau:

a. LAN (Local Area Network - Mạng cục bộ) : LAN thường được sử dụng trong nội bộ một cơ quan/tổ chức..., kết nối các máy tính trong một khu vực bán kính khoảng 100m-10km. Kết nối được thực hiện thông qua các môi trường truyền thông tốc độ cao, ví dụ cáp đồng trục hay cáp quang.

b. MAN (Metropolitan Area Network - Mạng đô thị) : Kết nối các máy tính trong phạm vi một thành phố. Kết nối này được thực hiện thông qua các môi trường truyền thông tốc độ cao (50-100 Mbit/s).

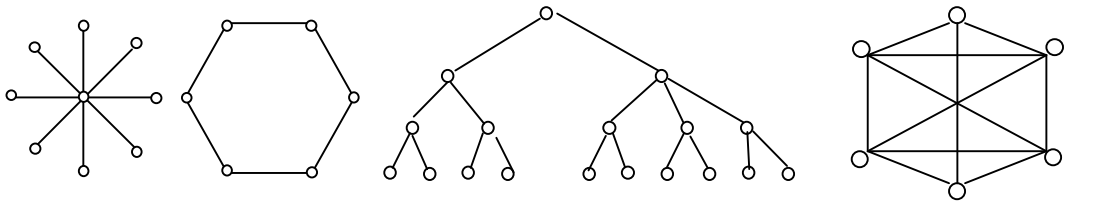
c. WAN (Wide Area Network) - Mạng diện rộng, kết nối máy tính trong nội bộ các quốc gia hay giữa các quốc gia trong cùng một châu lục. Thông thường kết nối này được thực hiện thông qua mạng viễn thông. Các WAN có thể được kết nối với nhau thành GAN hay tự nó đã là GAN.

d. GAN (Global Area Network) : Mạng toàn cầu, kết nối máy tính từ các châu lục khác nhau. Thông thường kết nối này được thực hiện thông qua mạng viễn thông và vệ tinh.

Trong các khái niệm nói trên, WAN và LAN là hai khái niệm hay được sử dụng nhất.

1.2.2 Dựa theo cấu trúc mạng

1.2.2.1 Kiểu điểm - điểm (point - to - point)



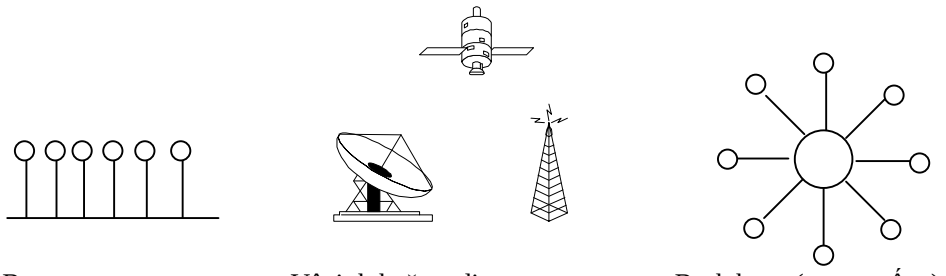
H×nh sao *Chu tr×nh* *H×nh c©y* *D¹ng ®Çy ®ñ*

Hình 1-1. Cấu trúc mạng kiểu điểm-điểm.

Đường truyền nối từng cặp nút mạng với nhau. Thông tin đi từ nút nguồn qua nút trung gian rồi gởi tiếp nếu đường truyền không bị bận. Do đó còn có tên là mạng lưu trữ và chuyển tiếp (*store and forward*).

1.2.2.2 Kiểu khuyếch tán

Bản tin được gởi đi từ một nút nào đó sẽ được tiếp nhận bởi các nút còn lại (còn gọi là broadcasting hay point to multipoint). Trong bản tin phải có vùng địa chỉ cho phép mỗi nút kiểm tra xem có phải tin của mình không và xử lý nếu đúng bản tin được gởi đến.



Hình 1-2. Sơ đồ kết nối theo kiểu khuyếch tán.

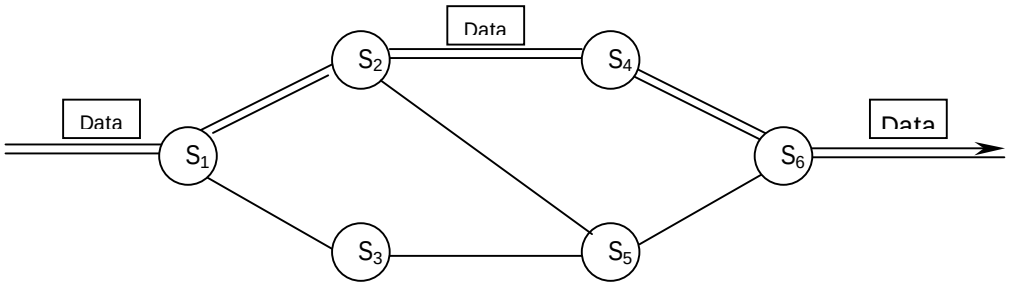
Trong cấu trúc dạng Bus và Vòng cần cơ chế "*trọng tài*" để giải quyết các xung đột (collision) xảy ra khi nhiều nút muốn truyền tin đồng thời. Trong cấu trúc vệ tinh hoặc radio, mỗi nút cần có ăng-ten thu và phát.

1.2.3 Theo phương pháp chuyển mạch

- Mạng chuyển mạch kênh (Line switching network), ví dụ như mạng điện thoại.
- Mạng chuyển mạch thông báo (Message switching network)
- Mạng chuyển mạch gói (Packet switching network)

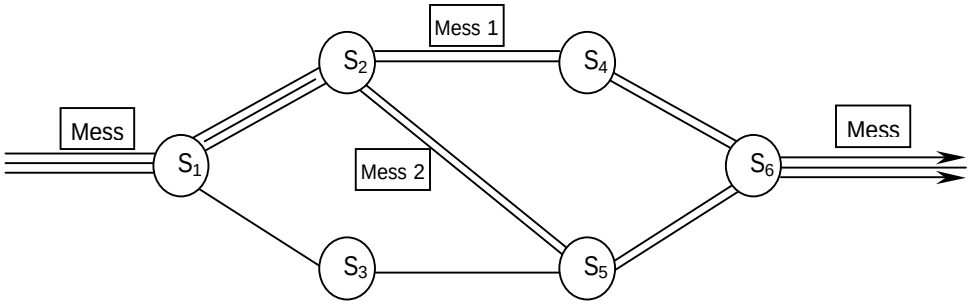
1.2.3.1 Chuyển mạch kênh

Chuyển mạch kênh (line switching) được dùng trong mạng điện thoại. Một kênh cố định được thiết lập giữa cặp thực thể cần liên lạc với nhau. Mạng này có hiệu suất không cao vì có lúc kênh bỏ không.



Hình 1-3. Mạng chuyển mạch kênh.

1.2.3.2 Mạng chuyển mạch bản tin



Hình 1-4. Phương pháp chuyển mạch thông báo.

Các nút của mạng căn cứ vào địa chỉ đích của "bản tin" để chọn nút kế tiếp. Như vậy các nút cần lưu trữ và đọc tin nhận được, quản lý việc truyền tin. Trong trường hợp bản tin quá dài và nếu sai phải truyền lại thì hiệu suất không cao. Phương pháp này giống như cách gởi thư thông thường.

- Ưu điểm so với phương pháp chuyển mạch kênh:

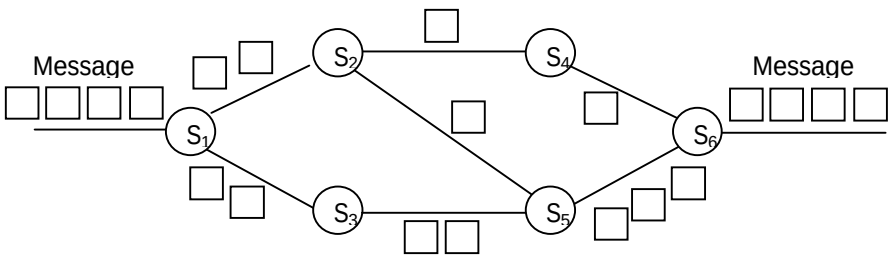
- Hiệu suất sử dụng đường truyền cao vì không bị chiếm dụng độc quyền mà được phân chia giữa nhiều thực thể.
- Mỗi nút mạng (hay nút chuyển mạch thông bao) có thể lưu trữ message cho tới khi kênh truyền rỗi mới gửi bản tin đi. Do đó giảm được tình trạng tắc nghẽn (congestion) trên mạng.
- Điều khiển việc truyền tin bằng cách sắp xếp độ ưu tiên cho các bản tin.
- Có thể tăng hiệu suất sử dụng giải thông của mạch bằng cách gán địa chỉ quảng bá (broadcast) để gởi bản tin đồng thời đến nhiều đích.

- Nhược điểm:

- Do không hạn chế kích thước của bản tin nên có thể dẫn đến phí tổn lưu trữ tạm thời cao và ảnh hưởng đến thời gian hồi đáp và chất lượng truyền đi.

Mạng chuyển mạch thông báo thích hợp với các dịch vụ thông tin kiểu thư điện tử (Email) hơn là đối với các ứng dụng có tính thời gian thực vì tồn tại độ trễ nhất định do lưu trữ và xử lý thông tin điều khiển tại mỗi nút.

1.2.3.3 Mạng chuyển mạch gói



Hình 1-5. Mạng chuyển mạch gói.

Bản tin được chia thành nhiều gói tin (packet) độ dài 512 bytes, phần đầu là địa chỉ đích, mã để tập hợp các gói. Các gói của các bản tin khác nhau có thể được truyền độc lập trên cùng một đường truyền. Vấn đề phức tạp ở đây là tạo lại bản tin ban đầu, đặc biệt khi được truyền trên các con đường khác nhau.

Chuyển mạch gói mềm dẻo, hiệu suất cao. Xu hướng phát triển hiện nay là sử dụng hai kỹ thuật chuyển mạch kênh và chuyển mạch gói trong cùng một mạng thống nhất gọi là mạng ISDN (*Integrated Services Digital Network* - Mạng thông tin số đa dịch vụ).

1.3 Kiến trúc phân tầng và chuẩn hoá mạng

Tình trạng không tương thích giữa các mạng đặc biệt là các mạng trên thị trường gây trở ngại cho những người sử dụng khác nhau. Do đó cần phải xây dựng mô hình chuẩn làm cơ sở cho các nhà nghiên cứu thiết kế mạng để tạo ra các sản phẩm mới về mạng, dễ phổ cập, sản xuất, sử dụng. Các chuẩn có vai trò quan trọng trong công tác thiết kế và xây dựng các hệ thống kỹ thuật và công nghệ.

Chuẩn hóa mạng máy tính là nêu ra các tiêu chuẩn cơ bản thống nhất về cấu trúc mạng giúp cho các mạng khác nhau có thể trao đổi thông tin được với nhau.

Để mạng hoạt động đạt khả năng tối đa, các tiêu chuẩn được chọn phải cho phép mở rộng mạng để có thể phục vụ những ứng dụng không dự kiến trước trong tương lai tại lúc lắp đặt hệ thống và điều đó cũng cho phép mạng làm việc với những thiết bị được sản xuất từ nhiều hãng khác nhau.

1.3.1 Các tổ chức chuẩn hoá mạng

Hai tổ chức chính thực hiện chuẩn hóa mạng là ISO và CCTTT.

1. ISO (*International Standards Organization*) - Tổ chức chuẩn hóa quốc tế. ISO hoạt động dưới sự bảo trợ của LHQ. Thành viên của ISO là các cơ quan tiêu chuẩn hóa của các quốc gia và các Ban chuyên môn. Ban TC97 được chia ra thành các tiểu ban và các nhóm công tác.
2. IEEE (*Institute of Electrical and Electronic Engineers*) - Viện nghiên cứu các vấn đề về kỹ thuật điện và điện tử của Mỹ. IEEE chịu trách nhiệm về tầng Data Link và Physical. Phân ban các chuẩn này là phân ban 802 (thành lập tháng Hai năm 1980).
3. CCITT (*Commité Consultatif Internatinal pour Télégraphe et Téléphone*) - Tổ chức tư vấn quốc tế về điện báo và điện thoại hoạt động dưới sự bảo trợ của LHQ, chuyên nghiên cứu nhằm công bố các khuyến nghị thống nhất về mạng

máy tính. Bao gồm các khuyến nghị liên quan đến việc truyền dữ liệu trên mạng, mạng ISDN.

4. ANSI (*American National Standards Institute*) :Viện nghiên cứu các chuẩn quốc gia của Mỹ.
5. ECMA (*European Computer Manufactures Association*) : Hiệp hội máy tính châu âu
6. ATM Forum (*Asynchronous Transfers Mode*) - Thực hiện các giải pháp cho mạng ISDN.
7. IETF (*Internet Enggineering Task Force*) : Sản xuất các chuẩn liên quan đến Internet (SNMP, TCP/IP ...)

1.3.2 Kiến trúc phân tầng

Để giảm độ phức tạp thiết kế, kiến trúc mạng được tổ chức thành một cấu trúc đa tầng, mỗi tầng được xây trên tầng trước nó, tầng dưới sẽ cung cấp dịch vụ cho tầng cao hơn. Tầng N trên một máy thực hiện việc giao tiếp với tầng N trên một máy khác. Các qui tắc, luật lệ được sử dụng cho việc giao tiếp này được gọi là các giao thức của tầng N.

Các thực thể (entity) nằm trên các tầng tương ứng trên những máy khác nhau gọi là các tiến trình đồng mức. Các tiến trình đồng mức giao tiếp với nhau bằng cách sử dụng các giao thức trong tầng của nó.

Giữa 2 tầng kề nhau tồn tại một giao diện (*interface*) xác định các hàm nguyên thủy và các dịch vụ tầng dưới cung cấp cho tầng trên.

Tập hợp các tầng và các giao thức được gọi là kiến trúc mạng (*Network Architecture*).

Cấu trúc phân tầng của mạng máy tính có ý nghĩa đặc biệt như sau :

- Thuận tiện trong công tác thiết kế, xây dựng và cài đặt các mạng máy tính, trong đó mỗi hệ thống thành phần được xem như là một cấu trúc đa tầng.
- Mỗi tầng được xây dựng dựa trên cơ sở tầng kề liền trước đó. Như vậy tầng dưới sẽ cung cấp dịch vụ cho tầng trên.
- Số lượng, tên gọi và chức năng của mỗi tầng sẽ được người thiết kế mạng máy tính cụ thể quy định.
- Tập hợp các giao thức, các vấn đề kỹ thuật và công nghệ cho mỗi tầng có thể được khảo sát, nghiên cứu triển khai độc lập với nhau.

- Giao thức : Mỗi khi trao đổi thông tin như điện thoại, telex, viết . . . người ta phải tuân theo một số quy luật. Các quy luật này được nhóm lại và gọi là giao thức (*protocol*).

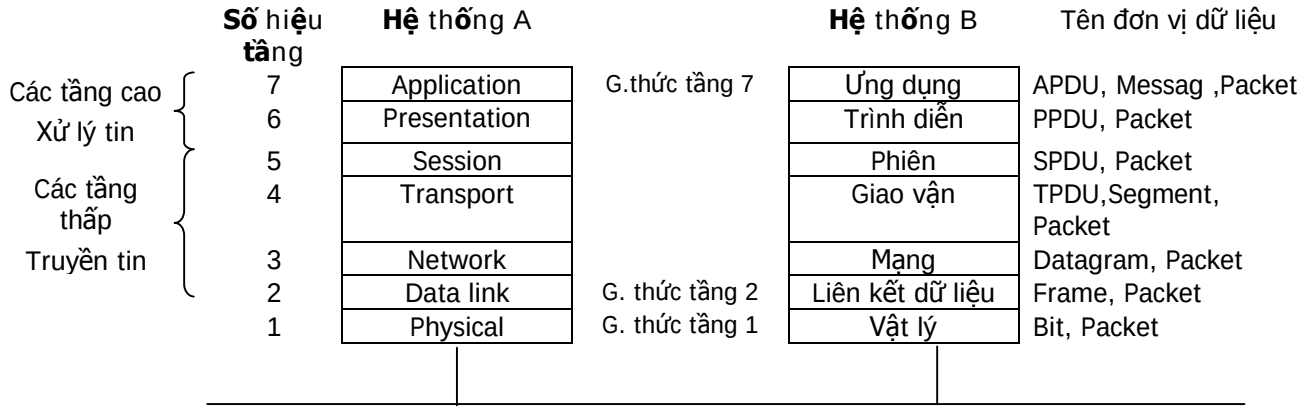
Giao thức có các chức năng chính như sau :

1. Định nghĩa cấu trúc khung một cách chính xác cho từng byte, các ký tự và bản tin.
2. Phát hiện và xử lý các lỗi, thông thường là gửi lại bản tin gốc sau khi phát hiện lần trước bị lỗi
3. Quản lý thứ tự các lệnh để đếm các bản tin, nhận dạng, tránh mất hoặc thu thừa bản tin.
4. Đảm bảo không nhầm lẫn giữa bản tin và lệnh
5. Chỉ ra các thuộc tính đường dây khi lập các đường nối đa điểm hoặc bán song công (cho biết ai đối thoại với ai).
6. Giải quyết vần đề xung đột thâm nhập (yêu cầu đồng thời), gửi khi chưa có số liệu, mất liên lạc, khởi động.

1.4 Mô hình OSI

1.4.1 Kiến trúc của mô hình OSI

Dựa trên kiến trúc phân tầng, ISO đã đưa ra mô hình 7 tầng (layer) cho mạng, gọi là mô hình kết nối hệ thống mở hoặc mô hình OSI (Open Systems Interconnection model), vào năm 1984.



Hình 1-6. Mô hình OSI 7 tầng.

Nhóm các tầng thấp (*physical, data link, network, transport*) liên quan đến các phương tiện cho phép truyền dữ liệu qua mạng. Các tầng thấp đảm nhiệm việc truyền dữ liệu, thực hiện quá trình đóng gói, dẫn đường, kiểm duyệt và truyền từng nhóm dữ liệu. Các tầng này không cần quan tâm đến loại dữ liệu mà nó nhận được từ hay gửi cho tầng ứng dụng, mà chỉ đơn thuần là gửi chúng đi.

Nhóm các tầng cao (*session, presentation, application*) liên quan chủ yếu đến việc đáp ứng các yêu cầu của người sử dụng để triển khai các ứng dụng của họ trên mạng thông qua các phương tiện truyền thông cung cấp bởi các nhóm tầng thấp.

Hệ thống kết nối mở OSI là hệ thống cho phép truyền thông tin với các hệ thống khác, trong đó các mạng khác nhau, sử dụng những giao thức khác nhau, có thể thông báo cho nhau thông qua chương trình để chuyển từ một giao thức này sang một giao thức khác.

Mô hình OSI đưa ra giải pháp cho vấn đề truyền thông giữa các máy tính không giống nhau. Hai hệ thống, dù khác nhau đều có thể truyền thông với nhau một cách hiệu quả nếu chúng đảm bảo những điều khiển chung sau đây :

1. Các hệ thống đều cài đặt cùng một tập hợp các chức năng truyền thông.
2. Các chức năng đó được tổ chức thành cũng một tập các tầng. Các tầng đồng mức phải cung cấp các chức năng như nhau, nhưng phương thức cung cấp không nhất thiết phải giống nhau.
3. Các tầng đồng mức phải sử dụng một giao thức chung.

Để đảm bảo những điều trên cần phải có các chuẩn xác định các chức năng và dịch vụ được cung cấp bởi một tầng (nhưng không cần chỉ ra chúng phải cài đặt như thế nào). Các chuẩn cũng phải xác định các giao thức giữa các tầng đồng mức. Mô hình OSI chính là cơ sở để xây dựng các chuẩn đó.

1.4.2 Sự ghép nối giữa các mức

Trong thực tế dữ liệu không truyền trực tiếp từ tầng i máy này sang tầng i máy kia (trừ tầng thấp nhất). tầng thấp nhất có đường truyền thông vật lý tới tầng thấp nhất của máy tương ứng từ đó dữ liệu và thông tin điều khiển lại được chuyển ngược lên tầng trên. Tầng trên chỉ xác định đường truyền thông lôgic (truyền thông ảo).

- Các Header của giao thức : Thông thường, thông tin điều khiển giao thức được gói thành một khối và được đặt trước dữ liệu nó đi kèm và được gọi là *Header* hay *Protocol Header*, được dùng để truyền thông tin giữa các tầng và giữa các máy tính với nhau. Các header của giao thức được phát triển theo các luật được cho trong tập tài liệu ASN.1 của ISẶ.
- Khi máy A gửi tin đi, các đơn vị dữ liệu đi từ tầng trên xuống dưới. Qua mỗi tầng nó được bổ sung thông tin điều khiển của tầng đó.
- Khi nhận tin, thông tin đi từ dưới lên. Qua mỗi tầng thông tin điều khiển được khử bỏ dần và cuối cùng máy B nhận được bản tin của A.

1.4.3 Chức năng của mỗi tầng

1. Tầng Vật lý

Cung cấp phương tiện truyền tin, thủ tục khởi động, duy trì huỷ bỏ các liên kết vật lý. Giữ nhiệm vụ chuyển tải các bit thông tin trên kênh truyền thông. Tầng Vật lý làm việc với các giao diện cơ, điện và giao diện thủ tục (chức năng) trên môi trường vật lý, không quan tâm đến nội dung biểu diễn của các bit.

Thực chất tầng này thực hiện nối liền các phần tử của mạng thành một hệ thống bằng các phương pháp vật lý, ở mức này sẽ có các thủ tục đảm bảo cho các yêu cầu về chuyển mạch hoạt động nhằm tạo ra các đường truyền thực cho các chuỗi bit thông tin.

2. Tầng liên kết dữ liệu

Thiết lập, duy trì, huỷ bỏ các liên kết dữ liệu kiểm soát luồng dữ liệu, phát hiện và khắc phục sai sót truyền tin

Tiến hành chuyển đổi thông tin dưới dạng chuỗi các bit ở mức mạng thành từng đoạn gọi là khung tin (frame). Sau đó đảm bảo truyền liên tiếp các khung tin tới tầng vật lý, đồng thời xử lý các thông báo từ trạm thu gửi trả lại. Bit thông tin trong khung tin đều mang những ý nghĩa riêng, bao gồm các *trường địa chỉ, trường kiểm tra, dữ liệu và kiểm tra lỗi* dùng cho các mục đích riêng.

Nhiệm vụ chính của mức 2 này là khởi tạo, tổ chức các khung tin và xử lý các thông tin liên quan tới khung tin.

3. Tầng mạng

Tầng mạng được xây dựng dựa trên kiểu nối kết *điểm - điểm* do tầng LKDL cung cấp, bảo đảm trao đổi thông tin giữa các mạng con trong một mạng lớn, mức này còn được gọi là mức thông tin giữa các mạng con với nhau.

Có nhiệm vụ gán địa chỉ cho các bản tin và chuyển đổi địa chỉ logic hay các tên thành các địa chỉ vật lý.

Thực hiện chọn đường truyền tin, cung cấp dịch vụ định tuyến (chọn đường) cho các gói dữ liệu trên mạng. Tầng này chỉ ra dữ liệu từ nguồn tới đích sẽ đi theo tuyến nào trên cơ sở các điều kiện của mạng, độ ưu tiên dịch vụ và các nhân tố khác.

Kiểm soát luồng dữ liệu, khắc phục sai sót, cắt/hợp dữ liệu, giúp loại trừ sự tắc nghẽn cũng như điều khiển luồng thông tin.

4. Tầng Giao vận

Tầng giao vận giúp đảm bảo độ tin cậy khi chuyển giao dữ liệu và tính toàn vẹn dữ liệu từ nơi gửi đến nơi nhận. Điều này được thực hiện dựa trên cơ chế kiểm tra lỗi do các tầng bên dưới cung cấp. Tầng giao vận còn chịu trách nhiệm tạo ra nhiều kết nối cục bộ trên cùng một kết nối mạng gọi là ghép kênh (multiplexing), phân chia thời gian xử lý (time sharing), cắt hợp dữ liệu.

Nhiệm vụ của mức này là xử lý các thông tin để chuyển tiếp các chức năng từ tầng phiên đến tầng mạng và ngược lại. Thực chất mức truyền này là để đảm bảo thông tin giữa các máy chủ với nhau. Mức này nhận các thông tin từ tầng phiên, phân chia thành các đơn vị dữ liệu nhỏ hơn và chuyển chúng tới mức mạng.

5. Tầng phiên

Thiết lập, duy trì, đồng bộ hoá và huỷ bỏ các phiên truyền thông. Liên kết phiên phải được thiết lập thông qua đối thoại và trao đổi các thông số điều khiển.

Dùng tầng giao vận để cung cấp các dịch vụ nâng cao cho phiên làm việc như: kiểm soát các cuộc hội thoại, quản lý thẻ bài (*token*), quản lý hoạt động (*activity management*).

Nhận dạng tên và thủ tục cần thiết cũng như là các công việc bảo mật, để hai ứng dụng có thể giao tiếp với nhau trên mạng. Nhờ tầng phiên, những người sử dụng lập được các đường nối với nhau, khi cuộc hội thoại được thành lập thì mức này có thể quản lý cuộc hội thoại đó theo yêu cầu của người sử dụng. Một kết nối giữa hai máy cho phép người sử dụng được đăng ký vào một hệ thống phân chia thời gian từ xa hoặc chuyển tập tin giữa 2 máy.

6. Tầng trình diễn

Quản lý cách thức biểu diễn thông tin theo cú pháp dữ liệu của người sử dụng, loại mã sử dụng (ASCII, QBCDIC, ...) và thực hiện các vấn đề nén dữ liệu.

Nhiệm vụ của mức này là lựa chọn cách tiếp nhận dữ liệu, biến đổi các ký tự, chữ số của mã ASCII hay các mã khác và các ký tự điều khiển thành một kiểu mã nhị phân thống nhất để các loại máy khác nhau đều có thể thâm nhập vào hệ thống mạng.

7. Tầng ứng dụng

Tầng này là giao diện giữa người sử dụng và môi trường hệ thống mở.

Tầng này có nhiệm vụ phục vụ trực tiếp cho người sử dụng, cung cấp tất cả các yêu cầu phối ghép cần thiết cho người sử dụng, yêu cầu phục vụ chung như chuyển các File, sử dụng các Terminal của hệ thống,.... Mức sử dụng bảo đảm tự động hoá quá trình thông tin, giúp cho người sử dụng khai thác mạng tốt nhất.

1.4.4 Các giao thức chuẩn của OSI

1.4.4.1 Các hàm nguyên thủy

Mỗi thực thể truyền thông với các thực thể ở tầng trên và dưới nó qua một *giao diện* (interface). Giao diện này gồm một hoặc nhiều điểm truy cập dịch vụ (SAP - Service Access Point). Thực thể tầng N-1 cung cấp dịch vụ cho thực thể tầng N thông qua việc gọi các hàm dịch vụ nguyên thủy (primitive).

Hàm nguyên thuỷ chỉ rõ chức năng cần thực hiện và được dùng để chuyển dữ liệu và thông tin điều khiển. Bốn hàm nguyên thuỷ được sử dụng để định nghĩa tương tác giữa các tầng kề nhau như sau :

request	*Ịêu cầu*
indication	*Chỉ báo*
response	*Trả lời*
confirm	*Xác nhận*

request được gởi bởi người sử dụng dịch vụ ở tầng N+1 trong hệ thống A để gọi thủ tục của giao thức ở tầng N. Ịêu cầu này được cấu tạo dưới dạng một hoặc nhiều đơn vị dữ liệu giao thức (PDU - *Protocol Data Unit*) để gởi tới B.

Khi nhận được PDU, một thủ tục của giao thức ở tầng N của B sẽ thông báo yêu cầu đó lên tầng N+1 bằng hàm nguyên thủy *indication*. Sau đó *response* được gởi từ N + 1 của B xuống N gọi thủ tục giao thức tầng N để trả lời tới A.

Khi nhận được trả lời này một thủ tục giao thức tầng N sẽ gởi hàm *confirm* lên N+1 để hoàn tất chu trình yêu cầu thiết lập liên kết của người sử dụng ở tầng N+1 của A.

Các chu trình của người sử dụng khác nhau được phân biệt nhờ khái niệm điểm thâm nhập dịch vụ (SAP - *Service Access Point*) ở ranh giới của 2 tầng N + 1 và N.

1.4.4.2 Các phương thức truyền thông

Tại mỗi tầng trong mô hình OSI có 2 phương thức hoạt động chính được sử dụng : phương thức có liên kết (*connection oriented*) và phương thức không liên kết (*connectionless*).

Với các phương thức truyền không liên kết thì chỉ có một giai đoạn truyền dữ liệu. Các gói tin dữ liệu (còn được gọi là datagram) được truyền độc lập với nhau theo một con đường xác định dần bằng địa chỉ đích được đặt trong mỗi datagram. Có 3 giai đoạn phân biệt :

- *Thiết lập liên kết* : hai thực thể cùng tầng ở hai đầu của liên kết sẽ thương lượng với nhau về tập các tham số sử dụng trong giai đoạn truyền dữ liệu.
- *Truyền dữ liệu* : các cơ chế kiểm soát sai sót, luồng dữ liệu, ghép kênh, cắt hợp dữ liệu được thực hiện để tăng cường độ tin cậy và hiệu suất của việc truyền dữ liệu.
- *Kết thúc truyền* : giải phóng các tài nguyên hệ thống đã được cấp phát cho liên kết để dùng vào mục đích khác.

Tương ứng với 3 giai đoạn trao đổi trên, có 3 loại thủ tục cơ bản được sử dụng : CONNECT, DATA, DISCONNECT.

Ví dụ đối với giao thức tầng N ta có các thủ tục :

N_CONNECT	Thiết lập liên kết
N_DATA	Truyền dữ liệu
N_DISCONNECT	Huỷ bỏ liên kết

Ngoài ra có một số các thủ tục phụ được sử dụng tuỳ theo chức năng của mỗi tầng.

Ví dụ:	Thủ tục N_RESTART	Dùng để khởi động lại hệ thống ở tầng 3
	Thủ tục T_EXPEDITED_DATA	Dùng cho việc truyền dữ liệu nhanh tầng 4
	Thủ tục S_TOKEN_GIVE	Dùng để chuyển điều khiển ở tầng 5

Mỗi thủ tục trên sẽ dùng các hàm nguyên thuỷ (*request, indication, response, confirm*) để tạo thành các hàm cơ bản của mô hình OSI.

1.5 Hệ điều hành mạng

Việc lựa chọn hệ điều hành mạng (NOS - Network Operating System) làm nền tảng cho mạng tuỳ thuộc vào kích cỡ của mạng hiện tại và sự phát triển trong tương lai, ngoài ra còn tuỳ thuộc vào những ưu điểm và nhược điểm của từng hệ điều hành.

Một số hệ điều hành mạng phổ biến hiện nay:

- Hệ điều hành mạng UNIX: Đây là hệ điều hành do các nhà khoa học xây dựng và được dùng rất phổ biến trong giới khoa học, giáo dục. Hệ điều hành mạng UNIX là hệ điều hành đa nhiệm, đa người sử dụng, phục vụ cho truyền thông tốt. Nhược điểm của nó là hiện nay có nhiều Version khác nhau, không thống nhất gây khó khăn cho người sử dụng và là hệ điều hành này phức tạp.
- Hệ điều hành mạng Windows 2000: Đây là hệ điều hành của hãng Microsoft, cũng là hệ điều hành đa nhiệm, đa người sử dụng. Được xây dựng dựa trên công nghệ của hệ điều hành Windows NT. Đặc điểm của nó là tương đối dễ sử dụng, hỗ trợ mạnh cho các phần mềm WINDOWS. Windows 2000 có thể

liên kết tốt với máy chủ Novell Netware, Unix. Tuy nhiên, để chạy có hiệu quả, Windows 2000 Server đòi hỏi cấu hình máy tương đối mạnh.

- Hệ điều hành mạng NetWare của Novell: Đây là hệ điều hành phổ biến trên thế giới trong thời gian cuối, nó có thể dùng cho các mạng nhỏ (khoảng từ 5-25 máy tính) và cũng có thể dùng cho các mạng lớn gồm hàng trăm máy tính. Netware là một hệ điều hành LAN dùng cho các máy tính theo chuẩn của IBM hay các máy tính Apple Macintosh, chạy trên hệ điều hành MS-DOS hoặc OS/2.

1.6 Mạng Internet

1.6.1 Lịch sử ra đời và phát triển

Vào những năm 60, Bộ Quốc phòng Mỹ cho triển khai khẩn trương một mạng lưới thông tin với yêu cầu: Nếu như một trạm trung chuyển nào đó trong mạng bị phá huỷ, toàn bộ hệ thống thông tin vẫn phải làm việc bình thường... Cơ quan Nghiên cứu Dự án Cao cấp (ARPA - Advanced Research Projects Agency) thuộc Bộ Quốc phòng Mỹ được giao trách nhiệm thực hiện việc nghiên cứu kỹ thuật liên mạng (internet) nhằm đáp ứng yêu cầu trên. Đây là mạng chuyển mạch gói (packet switching) đầu tiên trên thế giới, lấy tên là ARPAnet. Ban đầu, ARPAnet chỉ gồm một vài mạng nhỏ được chọn lựa của các trung tâm nghiên cứu và phát triển khoa học. Giao thức truyền thông lúc bấy giờ là kiểu điểm - điểm, rất chậm và thường xuyên gây tắc nghẽn trên mạng. Để giải quyết vấn đề này, vào năm 1974 Vinton G. Cerf và Robert O. Kahn đưa ra ý tưởng thiết kế một bộ giao thức mạng mới thuận tiện hơn, đó chính là tiền thân của giao thức TCP/IP.

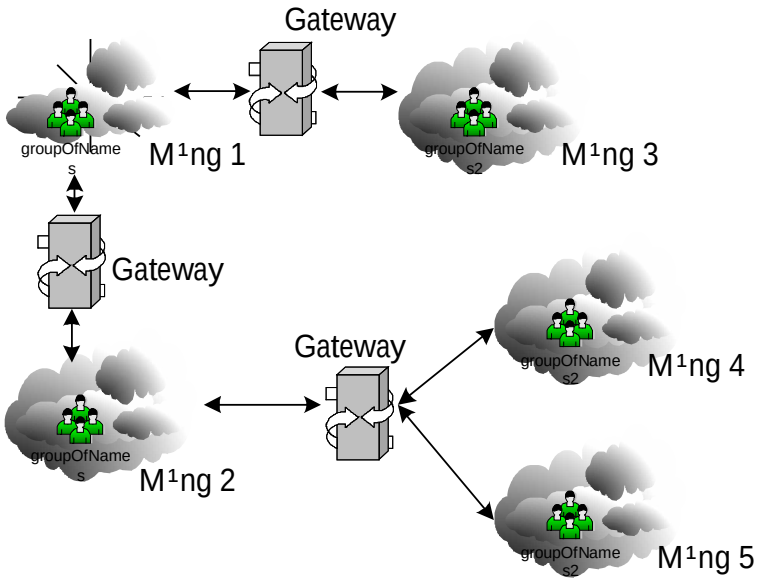
Tháng 09/1983, dưới sự tài trợ của Bộ Quốc phòng Mỹ, Berkeley Software Distribution đưa ra bản Berkeley UNIX 4.2BSD có kết hợp giao thức TCP/IP, biến TCP/IP thành phương tiện kết nối các hệ thống UNIX. Trên cơ sở đó, mạng ARPANỌT nhanh chóng lan rộng và chuyển từ mạng thực nghiệm sang hoạt động chính thức: nhiều trường đại học, viện nghiên cứu ghi tên gia nhập để trao đổi thông tin. Đến năm 1984, mạng ARPANỌT được chia thành hai nhóm mạng nhỏ hơn là MILNET, dành cho quốc phòng, và nhóm mạng thứ hai vẫn gọi là ARPANET, dành cho nghiên cứu và phát triển. Hai nhóm này vẫn có mối liên hệ trao đổi dữ liệu với nhau qua giao thức TCP/IP và được gọi chung là Enternet.

Mạng Internet đã và đang trở thành phương tiện trao đổi thông tin toàn cầu, là phương thức thông tin nhanh với lưu lượng truyền tải dữ liệu rất lớn. Thông qua Internet mà các nhà nghiên cứu khoa học kỹ thuật, các cơ quan giáo dục đào tạo, các nhà doanh nghiệp... có thể trao đổi thông tin với nhau, hoặc truy cập thông tin

của nhau về các công trình, các lĩnh vực nghiên cứu mới nhất; về các phương pháp, hình thức giáo dục và đào tạo, về các thông tin kinh tế, thị trường giá cả... một cách nhanh chóng, thuận tiện và dễ dàng.

1.6.2 Cấu trúc của mạng Internet

Mạng Internet không phải một mạng đơn mà là bao gồm nhiều mạng con (sub-network) được kết nối với nhau thông qua các cổng (gateway) như trên hình. Thuật ngữ mạng con ở đây mang nghĩa một *đơn vị mạng hoàn chỉnh* trong hệ thống mạng lớn. Mạng con hoàn toàn có thể là một mạng WAN với quy mô quốc gia, và có khả năng hoạt động độc lập với Internet. Do giao thức TCP/IP không phụ thuộc lớp vật lý, các mạng con có thể sử dụng những công nghệ ghép nối khác nhau (như Ọthernet, X.25,...) mà vẫn giao tiếp được với nhau.

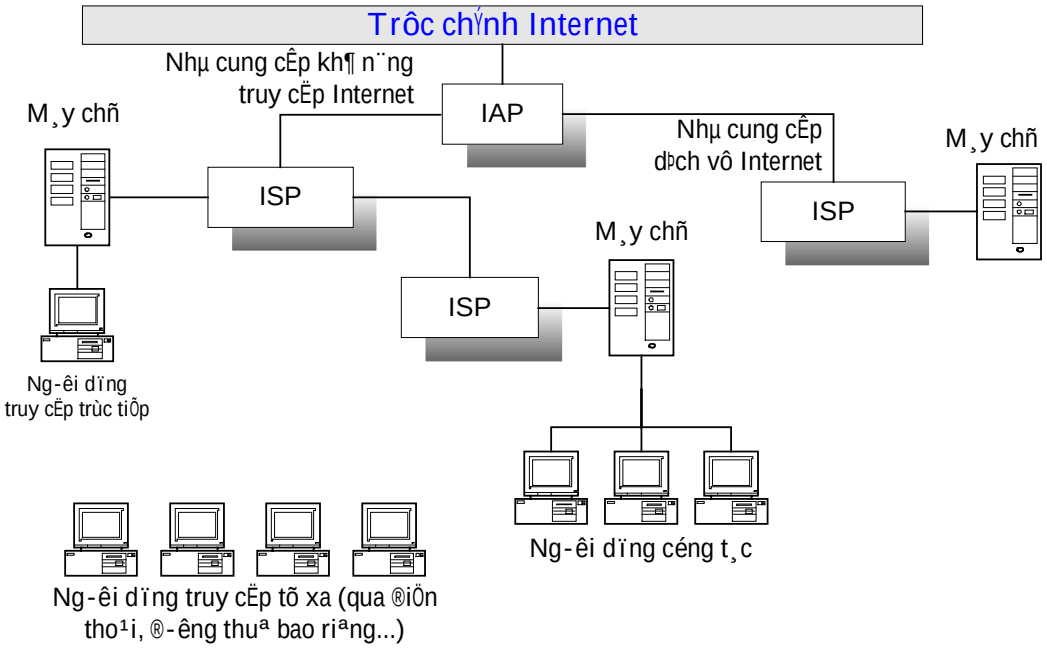


Hình 1-7. Cấu trúc của mạng Internet.

Các cổng được dùng để nối các mạng con tạo thành một mạng lớn.

Có 2 cách kết nối với Internet như sau :

- Máy con nối trong mạng LAN (hay WAN) và mạng này nối với Internet
- Máy con nối đến một trạm cung cấp dịch vụ Internet (Internet Service Provider), thông qua đó kết nối với Internet. Trong hình trên, ta có thể thấy các trạm ISP lại kết nối với Internet thông qua IAP (Internet Access Provider). Một IAP có thể làm luôn chức năng của ISP nhưng ngược lại thì không.



Hình 1-8. Sơ đồ kết nối của các trung tâm cung cấp dịch vụ (ISP)

1.6.3 Các kiến trúc khác

Level	ISO	ARPANET	SNA	DECNET
7	Application	User	End User	Application
6	Presentation	Telnet, FTP	NAU Services	
5	Session	(none)	Data Flow Control	(none)
			Transmision Control	
4	Transport	Host - Host		Network Services
		SRC to DESI - IMP	Path Control	
3	Network			Transport
		IMP - IMP		
2	Datalink		Data Link Control	Data Link Control
1	Physical	Physical	Physical	Physical

ARPANET: Advanced Research Projects Agency
FTP: File Transfer Protocol
SNA: System Network Architecture cña IBM
IMP: Interface Message Procesor
NAU: Network Addressable Unit

Chương 2

TẦNG VẬT LÝ

Nhiệm vụ của tầng vật lý là chuyển các bit tin từ máy này đến máy kia. Tốc độ truyền tin phụ thuộc vào môi trường truyền tin. Tín hiệu truyền có thể ở dạng tương tự (*analog*) hoặc ở dạng số (*digital*). Hướng phát triển hiện nay :

- Truyền tin bằng cáp quang, bằng vệ tinh.
- Hệ thống nối nhanh (Fast - Connect), hệ thống chuyển mạch gói
- Mạng thông tin số đa dịch vụ (Integrated Services Digital Network)

2.1 Môi trường truyền tin

2.1.1 Phương tiện truyền

Mục đích lắp đặt cáp là đảm bảo dung lượng (tốc độ) cần thiết cho các nhu cầu truyền thông trong mạng. Hệ thống cáp cần phải ổn định. Để đạt được mục tiêu này, người quản trị mạng phải cân đối bốn yếu tố sau:

- Tốc độ truyền lớn nhất của hệ thống cáp hiện hành, khả năng nâng cấp.
- Nhu cầu về tốc độ truyền thông trong vòng 5-10 năm tới là bao nhiêu.
- Chọn trong số những loại cáp đang có trên thị trường.
- Chi phí để lắp đặt thêm cáp dự phòng.

Việc kết nối vật lý một máy tính vào mạng được thực hiện bằng cách cắm một card giao tiếp mạng NIC (Network Interface Card) vào khe cắm của máy tính và nối với cáp mạng. Sau khi kết nối vật lý đã hoàn tất, quản lý việc truyền tin giữa các trạm trên mạng tuỳ thuộc vào phần mềm mạng.

NIC sẽ chuyển gói tín hiệu vào mạng LAN, gói tín hiệu được truyền đi như một dòng các bit dữ liệu thể hiện bằng các biến thiên tín hiệu điện. Khi nó chạy trong cáp dùng chung, mọi trạm gắn với cáp đều nhận được tín hiệu này, NIC ở mỗi trạm sẽ kiểm tra địa chỉ đích trong tín hiệu đầu của gói để xác định đúng địa chỉ đến, khi gói tín hiệu đi tới trạm có địa chỉ cần đến, đích ở trạm đó sẽ sao gói tín hiệu rồi lấy dữ liệu ra khỏi khung tin và đưa vào máy tính.

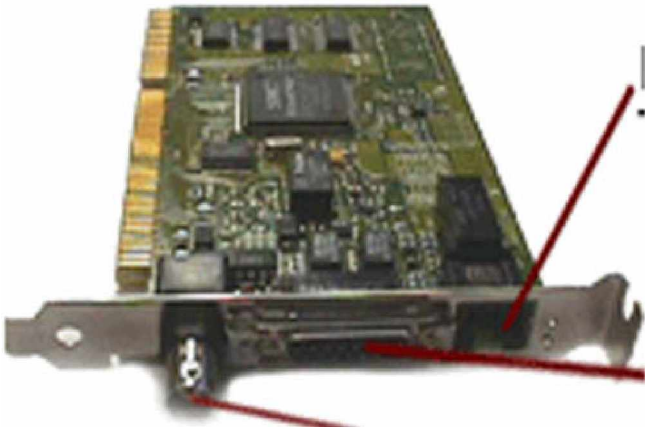
Có hai kỹ thuật truyền tín hiệu đã mã hóa lên mạng : Truyền ở dải tần gốc (baseband) và truyền ở dải tần rộng (broadband).

Đặc tính của cáp bao gồm sự nhạy cảm với nhiễu của điện, độ mềm dẻo, khả năng uốn nắn để lắp đặt, cự ly truyền dữ liệu, tốc độ truyền (Mbit/s). Hiện nay, tốc độ truyền dữ liệu trên các loại cáp biến động từ 10Mbit/s đến 100Mbit/s và hơn nữa.

Có 3 nhóm cáp chính được dùng để nối hầu hết các mạng :

- Cáp đồng trục (Coaxial)
- Cáp xoắn đôi (Twisted-Pair) : gồm có cáp xoắn đôi trần (Unshielded Twisted-Pair) và cáp xoắn đôi có bọc (Shielded Twisted-Pair).
- Cáp sợi quang (Fiber-Optic)

2.1.1.1 Card mạng



Hình 2-1. Card mạng (NIC)

Card mạng còn được gọi là card giao tiếp mạng NIC (Network Interface Card) được lắp đặt trong mỗi máy tính trong mạng cục bộ, Card này có nhiệm vụ chuyển dữ liệu từ máy tính vào cáp mạng và ngược lại. Quá trình này chính là sự chuyển đổi từ tín hiệu số của máy tính thành các tín hiệu điện hay quang được truyền dẫn trên cáp mạng. Đồng thời nó cũng thực hiện chức năng tổ hợp dữ liệu thành các gói và xác định nguồn và đích của gói.

- Các loại đầu nối cho card mạng :

Một vài loại card mạng có nhiều đầu nối để nối với cáp mạng, để xác định đầu nào dùng ta có thể thay đổi các jump hay công tắc chuyển DIP ngay trên card mạng hoặc sử dụng phần mềm.

- Mạng thin Ethernet sử dụng các đầu nối cáp đồng trục BNC (British Naval Connector)
- Mạng thicknet dùng giắc nối AUI 15 chân để cắm vào đầu DB15 của card mạng.
- Mạng Ethernet twisted-pair (10 Base T) sử dụng đầu nối RJ45.

2.1.1.2 Cáp đồng trục

Cáp đồng trục được chế tạo gồm một dây đồng ở giữa chất cách điện, chung quanh chất cách điện được quán bằng dây bện kim loại dùng làm dây đất. Giữa dây đồng dẫn điện và dây đất có một lớp cách ly, ngoài cùng là một vỏ bọc bảo vệ.

Cáp đồng trục có hai loại : loại nhỏ (Thin) và loại to (Thick). Dây cáp đồng trục loại nhỏ được thiết kế để truyền tin cho băng tần cơ bản (Base Band) hoặc băng tần rộng (broadband). Dây cáp loại to dùng cho đường xa, dây cáp nhỏ dùng cho đường gần, tốc độ truyền tin qua cáp đồng trục có thể đạt tới 35 Mbit/s.

2.1.1.3 Cáp dây xoắn (Twisted Pair)

Cáp xoắn gồm hai sợi dây đồng được xoắn cách điện với nhau. Nhiều đôi dây cáp xoắn gộp với nhau và được bọc chung bởi vỏ cáp hình thành cáp nhiều sợi. Cáp này có đặc tính dễ bị ảnh hưởng của nhiễu điện nên chỉ truyền dữ liệu ở cự ly khoảng 100m (khoảng 328 feet). Cáp xoắn đôi có hai loại: cáp xoắn đôi không bọc (UTP) và cáp xoắn đôi có bọc (STP).

Cáp xoắn thường được dùng trong hệ thống điện thoại để truyền tín hiệu tương tự (analog) cũng như tín hiệu số (digital). Trong khoảng cách vài km thì không cần bộ khuyếch đại và có tốc độ ở mức megabit/giây.

2.1.1.4 Cáp quang (Fiber Optics)

Khi các tín hiệu số được điều chế thành các tín hiệu xung ánh sánh thì được truyền tải qua cáp quang. Cáp sợi quang bao gồm một sợi thuỷ tinh cực mảnh gọi là lõi (core), được bao bọc bởi một lớp thuỷ tinh đồng tâm gọi là lớp vỏ bọc hay còn gọi là lớp phủ (cladding). Đôi khi các sợi được làm bằng chất dẻo. Chất dẻo dễ lắp đặt hơn nhưng không thể mang xung ánh sáng đi xa như thuỷ tinh.

Mỗi sợi thuỷ tinh chỉ truyền tín hiệu theo một hướng nhất định, do đó cáp có 2 sợi nằm trong vỏ bọc riêng biệt : một sợi truyền và một sợi nhận. Cáp sợi quang có thể truyền tín hiệu đi xa hơn với tốc độ cực nhanh (theo lý thuyết cáp quang có thể truyền tín hiệu với tốc độ tối đa 200.000Mbit/s).

Cáp quang có dải thông lớn hơn cáp đồng, ưu điểm mạnh của cáp quang là khoảng cách truyền dẫn lớn, giá rẻ, dung lượng truyền cao.

2.1.1.5 Vệ tinh thông tin

Vệ tinh truyền thông (communication satellites) nhận thông tin mặt đất, khuyếch đại tín hiệu thu được và phát lại xuống mặt đất ở tần số khác để tránh giao thoa (interference) với tín hiệu thu được. Các vệ tinh có vai trò như những trạm lặp tin giữa các trạm mặt đất với nhau. Một vệ tinh đều phủ sóng rất rộng và có thể có nhiều trạm mặt đất, thường hoạt động ở tần số 12 - 14Ghz. Truyền tin qua vệ tinh có dãi truyền rất rộng, do đó những khoảng cách xa (hàng trăm km) được bảo đảm chất lượng tin. Ngoài ra giá của truyền vệ tinh đang giảm nhanh.

Uỷ ban kỹ thuật điện tử (IEEE) đề nghị dùng các tên sau đây để chỉ 3 loại dây cáp dùng với mạng Ethernet chuẩn 802.3 :

1. Dây cáp đồng trục sợi to (thick coax) gọi là 10BASE5, có tốc độ 10 Mbps, tần số cơ sở, ≤ 500m.

2. Dây cáp đồng trục sợi nhỏ (thin coax) gọi là 10BASE2, có tốc độ 10 Mbps, tần số cơ sở, ≤ 200m.
3. Dây cáp đôi xoắn không vỏ bọc (twisted pair) gọi là 10BASET, có tốc độ 10 Mbps, tần số cơ sở, sử dụng cáp sợi xoắn.
4. Dây cáp quang (Fiber Optic Inter-Repeater Link) gọi là FOIRL .

2.1.2 Các thông số cơ bản của môi trường truyền tin

2.1.2.1 Độ suy giảm

Tín hiệu trên đường dây bị suy giảm trong quá trình truyền tin. Để khắc phục ta dùng các bộ khuyếch đại (amplifiers). Độ suy giảm được tính bằng đơn vị decibel. Nếu điện thế ban đầu là V_1 và sau đó giảm xuống V_2 thì số decibel của độ suy giảm được định nghĩa như sau:

$$\mathrm{S\,(decibel)} = 20\log_{10}\frac{V_1}{V_2}$$

2.1.2.2 Độ nhiễu

Điện từ trường trong môi trường truyền tin gây nhiễu cho các tín hiệu mang thông tin. Để khắc phục ta dùng các bộ lọc nhiễu (*filters*). Để đặc trưng độ nhiễu trên đường dây, ta dùng tỉ số tần số tín hiệu/tạp âm (Signal/Noise - S/N) :

$$SN(decibel) = 10\log_{10}\frac{S}{N} \quad (S: Signal;\ \ N: Noise)$$

2.1.2.3 Tốc độ truyền

$$C = B\log_2\left(1+\frac{S}{N}\right) bit/s$$

Trong đó B là độ rộng dải tần tính bằng Hz. C là tốc độ tính bằng bit/giây (b/s). Nếu mạng điện thoại có dải tần 3000Hz, tỉ số S/N = 20dB thì tốc độ truyền cực đại là :

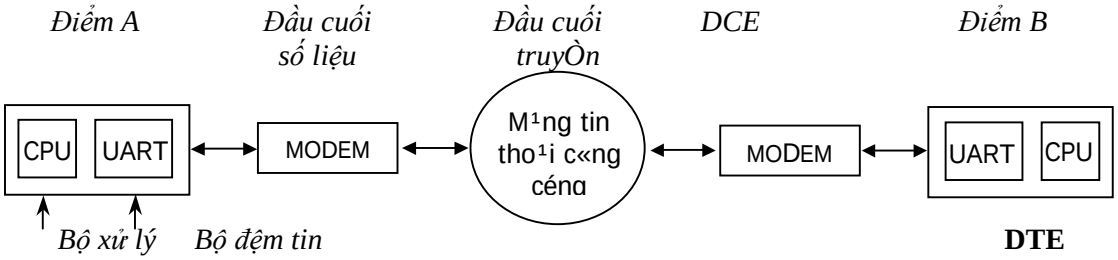
$$\frac{S}{N} = 10\log_{10}\frac{S}{N} = 20 \rightarrow \frac{S}{N} = 100 \qquad C = B\log_2\left(1+\frac{S}{N}\right) = 3000\times\log_2(1+100) = 19963b/s$$

Các tín hiệu trên kênh truyền có thể là tín hiệu tương tự hoặc tín hiệu số và tương ứng sẽ tạo thành kênh tương tự hoặc kênh số.

2.2 Chuẩn giao diện

2.2.1 Modem

Modem là bộ điều chế và giải điều chế biến đổi các tín hiệu số thành các tín hiệu tương tự và ngược lại trên mạng điện thoại.



Hình 2-2. Sơ đồ truyền tin giữa hai điểm A và B.

Tín hiệu số từ máy tính đến modem, được modem biến đổi thành tín hiệu tương tự để có thể đi qua mạng điện thoại. Tín hiệu này đến modem ở điểm B được biến đổi ngược lại thành tín hiệu số đưa vào máy tính ở B.

Các kỹ thuật điều chế cơ bản là điều chế biên độ AM, điều chế tần số FM, điều chế pha PM .

- Điều chế biên độ : Các tín hiệu 1 và 0 được phân biệt bởi biên độ, còn tần số của tín hiệu là giống nhau. Điều chế biên độ để thực hiện nhưng dễ bị nhiễu.
- Điều chế tần số : Các tín hiệu 1 và 0 được phân biệt bởi tần số, còn biên độ các tín hiệu giống nhau.

  Kỹ thuật điều tần phức tạp hơn nhưng tính chống nhiễu cao.
- Điều chế theo pha : Các tín hiệu 1 và 0 được phân biệt bởi các pha của dao động, còn biên độ và tần số của các tín hiệu giống nhau. Điều pha cũng phức tạp nhưng ít bị nhiễu.

Để tăng tốc độ truyền tin người ta kết hợp điều pha với điều biên gọi là điều pha biên.

Hiện nay có rất nhiều loại modem hiện đại từ loại thấp: 300, 600, 1200, 2400 bit/s, đến loại 9600 bit/s. Với tốc độ truyền tương đối cao trên đường truyền băng hẹp (băng thoại) nên đòi hỏi những phương pháp điều biên phức tạp.

Các *phương thức truyền dữ liệu* giữa hai điểm có thể là:

- Một chiều đơn (simplex)
- Hai chiều luân phiên (half - duplex)
- Hai chiều đầy đủ (duplex)

Truyền một chiều đơn chỉ cho phép truyền một hướng. Truyền hai chiều luân phiên cho phép truyền hai hướng, nhưng mỗi thời điểm chỉ có một hướng được truyền, sau đó phải thực hiện chuyển mạch để truyền ngược lại. Truyền hai chiều đầy đủ có thể nhận hoặc phát cùng một lúc. Các modem hiện nay đều có thể hoạt động ở hai chế độ bán song công và song công.

2.2.2 DTE và DCE

Trước khi nghiên cứu các chuẩn cho giao diện tầng Vật lý, chúng ta có hai khái niệm mới : đó là DTE và DCE.

- DTE (Data Terminal Equipment - Đầu cuối số liệu) : là khái niệm được sử dụng để chỉ các máy mà người sử dụng bình thường thao tác trực tiếp lên đó. Các máy này có thể là máy tính hay trạm cuối.
- DCE (Data Communication Equipment - Đầu cuối truyền) : là khái niệm chỉ các thiết bị cuối kênh dữ liệu có chức năng nối các DTE với các đường truyền vật lý và chuyển đổi dữ liệu. DCE có thể là các Modem, Tranducer, Multiplexer...

ISO qui định các chuẩn quy ước phương thức ghép nối giữa đầu cuối số liệu DTE và đầu cuối truyền DCE.

2.2.3 Chuẩn RS-232C

Đầu những năm 50, chuẩn RS-232(Recommended Standard 232C, của EIA) được phát triển để truyền tin giữa các thiết bị đầu cuối dữ liệu. Chuẩn này hiện nay đang được sử dụng, nó chính là các cổng COM1, COM2 trên các máy PC.

- *Phần cơ học :* là một bộ có 25 chân độ rộng tính ở giữa là 47,05mm ± 13 hàng trên đánh số 1 ÷ 13 (trái qua phải) hàng dưới 14 ÷ 25 (trái qua phải).
- *Phần điện :* gồm qui ước logic 1 <-3V và logic 0 >+ 3V.

Tốc độ truyền cho phép 20 *kbps* qua dây cáp 15m (thường là 9,6 *kbps*)

Từ năm 1987, RS-232-C đã được sửa đổi và đặt tên lại là EIA-232-D. Ngoài ra còn có một số chuẩn mở rộng khác như RS-422-A, RS-423-A RS-449, các khuyến nghị loại X của CCITT như X21. . .. Mặc dầu RS-232-C vẫn là chuẩn thông dụng nhất cho giao diện DTE/DCE nhưng các chuẩn mới nói trên được áp dụng phổ biến hiện nay.

Đối với các máy tính, thông thường người ta sử dụng hai cổng COM1, COM2 để *kết nối trực tiếp*. Cổng COM1 có địa chỉ vào/ra là 3F8_3FF hex và ngắt là IRQ4, cổng COM2 có địa chỉ vào/ra là 2F8_2FF hex và ngắt là IRQ3. Các chân cắm của hai cổng cũng được chuẩn hóa để tiện lợi hơn cho việc sử dụng.

Chương 3

TẦNG LIÊN KẾT DỮ LIỆU

3.1 Chức năng

Tầng liên kết dữ liệu thực hiện các công việc chính như sau :

- Định danh các thiết bị trên mạng, cấu hình logic của mạng.
- Điều khiển luồng dữ liệu và việc truy nhập ở tầng vật lý.
- Phát hiện và chỉnh sửa các lỗi xuất hiện trong quá trình truyền dữ liệu.

Chức năng chính của tầng LKDL là tách rời các khung thành các bít để truyền đi và kiến tạo các khung (frames) từ các dòng bít nhận được.

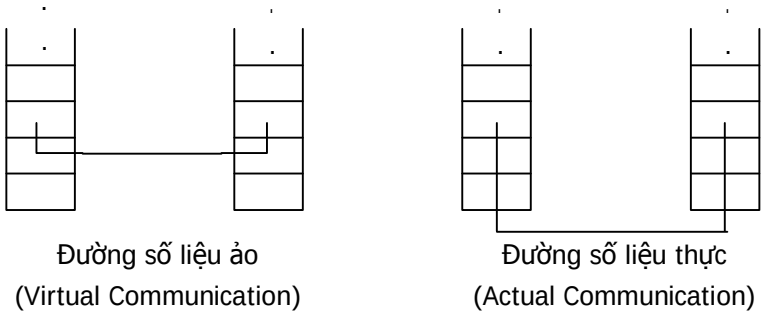
Tầng LKDL nghiên cứu các thuật toán thực hiện thông tin hiệu suất, tin cậy giữa hai máy cạnh nhau ở tầng 2. Đưa ra các thủ tục truyền tin có lưu ý đến lỗi có thể xẩy ra do nhiễu trên đường dây, sự trễ do lan truyền.

Thông thường, tầng LKDL có liên quan đến nhiễu của tín hiệu của phương tiện truyền vật lý, cho dù là truyền qua dây đồng, cáp quang hay truyền thông qua sóng ngắn. Nhiễu là một vấn đề rất thông thường và có thể do rất nhiều nguồn khác nhau, trong đó có cả nhiễu của các tia vũ trụ, nhiễu do tạp âm của khí quyển và từ các nguồn khác nhau.

3.2 Các vấn đề của tầng liên kết dữ liệu

3.2.1 Cung cấp dịch vụ cho tầng mạng

Tầng 2 chuyển dữ liệu từ mức 3 ở máy nguồn tới mức 3 ở máy nhận.



Hình 3-1. đường truyền dữ liệu trong tầng LKDL.

Các dịch vụ tầng 2 có thể là:

1. Dịch vụ không kết nối, không biên nhận (*Unacknowledged Connectionless Service*)
2. Dịch vụ không kết nối, có biên nhận (*Acknowledged Connectionless Service*)
3. Dịch vụ có kết nối (*Connection Oriented Service*)

Dịch vụ kết nối có hướng có 3 giai đoạn: *kết nối, truyền số liệu, tách bỏ liên kết* (kết thúc) : CONNECT, DATA, DISCONNECT. Truyền tin giữa 2 tầng kề nhau dùng các hàm dịch vụ nguyên thủy (request, indication, response và confirm).

Dịch vụ không kết nối được thể hiện bằng một bước duy nhất là truyền tin, không cần thiết lập liên kết logic. Các đơn vị dữ liệu truyền độc lập với nhau.

3.2.2 Khung tin - Nhận biết gói tin

Để cung cấp dịch vụ cho tầng mạng, tầng LKDL phải dùng dịch vụ được cung cấp từ tầng Vật lý. Tầng Vật lý tiếp nhận dòng bít và giao cho nơi nhận. Dòng bit này có thể có lỗi. Tầng LKDL sẽ kiểm tra và nếu cần sẽ sửa lỗi.

Tầng LKDL tách dòng bit thành các khung tin (frame) và tính thông số kiểm tra tổng (checksum) cho mỗi khung tin này, nếu kết quả tính được khác với checksum chứa trong khung tin, nghĩa là có lỗi và khi đó lỗi sẽ được thông báo cho nơi gửi.

Muốn tách các khung tin, có thể chèn các đoạn phân cách (timegaps) vào giữa các khung tin, giống như khoảng trống (*space*) giữa các từ trong văn bản. Nhưng điều này khó thực hiện nên người ta thường dùng các phương pháp sau :

- Đếm số ký tự : Hiện nay ít được dùng, vì từ đếm cũng bị lỗi khi truyền.
- Dùng ký tự bắt đầu (STX) và kết thúc (ETX) với ký tự đệm (DLE).
- Dùng các cờ (*flags*) đánh dấu bắt đầu và kết thúc với các bit đệm.

3.2.3 Kiểm tra lỗi

Các cách để kiểm tra lỗi trong quá trình truyền :

- Dùng thông số trả lời có biên nhận (ACK) hoặc không biên nhận (NAK) để biết đã nhận đúng bản tin hay phải phát lại.
- Dùng bộ định thời gian, nếu quá thời gian quy định không có trả lời nghĩa là bản tin chưa nhận được.
- Dùng phương pháp đánh số thứ tự các khung tin (frame) được gửi đi.

Quá trình kiểm tra lỗi đồng thời với quản lý thời gian và số thứ tự của các khung tin nhằm bảo đảm mỗi khung tin chỉ nhận được một lần duy nhất. Đây là chức năng quan trọng của tầng LKDL.

3.2.4 Điều khiển luồng dữ liệu

Trong quá trình truyền dữ liệu, nếu tốc độ bên phát nhanh hơn bên thu thì xảy ra hiện tượng mất tin do không nhận kịp. Vì vậy cần phải điều khiển luồng truyền

(*flow control*) để quá trình thu phát được phối hợp nhịp nhàng và đồng bộ với nhau. Chức năng có tại một vài cấp giao thức, kể cả tầng con LLC.

Các giao thức phải chứa các quy tắc xác định rõ khi nào nơi gởi có thể phát các khung tin kế tiếp.

3.2.5 Quản lý liên kết

Một chức năng khác của tầng LKDL là quản lý các kết nối như tách, nối, đánh số khung tin, bắt đầu lại khi lỗi, quản lý các thiết bị đầu cuối thứ cấp hoặc sơ cấp bằng khung tin thăm dò (poll).

3.2.6 Nén dữ liệu khi truyền

Nén dữ liệu là một vấn đề quan trọng đơn vị việc truyền dữ liệu trên mạng. Về cơ bản, nén dữ liệu là ép chúng lại để đỡ tốn chỗ khi lưu trữ trên đĩa và đỡ tốn thời gian khi truyền trên đường dây. Thực tế, các dữ liệu số chứa nhiều đoạn lặp đi lặp lại, nén dữ liệu sẽ thay thế các thông tin lặp lại bằng một ký hiệu hoặc một đoạn mã để rút ngắn độ dài của tập tin. Các kỹ thuật nén dữ liệu cơ sở bao gồm :

- *Null compression* : Thay thế một dãy các dấu cách bằng một mã nén và một giá trị số lượng các dấu cách.
- *Run-length compression* : Mở rộng kỹ thuật trên bằng cách nén bất kỳ một dãy nào có từ 4 ký tự lặp. Các ký tự này được thay thế bằng một mã nén, là một trong các ký tự này, và một giá trị bằng đúng số lần lặp.
- *Keyword encoding* : Tạo ra một bảng mã cho các từ hoặc các cặp ký tự thường xuyên xuất hiện và thay thế.
- *Phương pháp thống kê Huffman* : Kỹ thuật nén này giả thiết rằng sự phân bố của các ký tự trong dữ liệu là không đồng nhất. Tức là một số ký tự xuất hiện nhiều hơn các ký tự khác. Ký tự nào càng xuất hiện nhiều thì càng ít tốn bit để mã hóa nó. Một bảng được tạo ra để ghi lại lược đồ mã hóa và bảng này có thể chuyển cho modem nhận để nó biến đổi trở lại các ký tự đã mã hóa.
- Ngoài ra còn một thuật toán nén nữa được gọi là nén ngẫu nhiên. Thuật toán này được sử dụng trong một chuẩn nén dữ liệu V.24bits

3.3 Phát hiện và hiệu chỉnh lỗi

Trong khi truyền đi một byte trong hệ thống máy tính thì khả năng xảy ra một lỗi do hỏng hóc ở phần nào đó hoặc do nhiễu gây nên là khá lớn. Các kênh vào-ra thường xẩy ra nhiều lỗi, đặc biệt là khi truyền số liệu. Phần lớn các hệ thống đều có các phương pháp phát hiện và sau đó sửa lỗi. Quá trình sửa lỗi thường khó hơn rất nhiều so với phát hiện lỗi. Có thể chia phương pháp xử lý lỗi ra làm hai nhóm:

- Phát hiện lỗi và thông báo cho bên phát biết để phát lại tin.
- Phát hiện lỗi và tự sửa.

3.3.1 Phương pháp bit chẵn lẻ (Parity)

Đây là phương pháp thường dùng nhất để phát hiện lỗi. Bằng cách thêm 1 bit (được gọi là bit chẵn lẻ) vào từ nhị phân phụ thuộc vào tổng số các bit 1 trong một từ là một số chẵn hay lẻ, và nhờ vào phép toán logic XOR, ta sẽ biết được bit thêm vào đó là bit chẵn hay bit lẻ.

Mạch kiểm tra sẽ xác định các số bit 1 có đúng tính chẵn lẻ hay không. Phương pháp tương đối đơn giản và có hai cách như sau :

- Kiểm tra ngang (VRC - Vertical Redundancy Checking) : Thêm một bit chẵn lẻ vào mỗi byte để phát hiện lỗi. Cách này làm mất đi khoảng 12,5% dung lượng bản tin. Để khắc phục ta có thể dùng phép kiểm tra tổng các byte.
- Kiểm tra dọc (LRC - Longitudinal Redundancy Checking) : lỗi được phát hiện trong các khối byte thay cho việc tìm lỗi trong từng byte. Trong phương pháp này người ta thêm mỗi khối *1 byte ở cuối*, byte này mang các thông tin về tính chất đặc thù của khối (Characteristic Redundancy Checking - CRC). Byte này đơn giản có thể tính bằng phép logic XOR của tất cả các byte trong khối hoặc tính theo đa thức chuẩn để được FCS.

  Ví dụ :

Vị trí bit trong ký tự	Khối ký tự truyền đi					LRC
	A	S	C	I	I	
0	1	1	1	1	1	1
1	0	0	0	0	0	0
2	0	1	0	0	0	1
3	0	0	0	1	1	0
4	0	0	0	0	0	0
5	0	1	1	0	0	0
6	1	1	1	1	1	1
VRC	0	0	1	1	1	1

Kiểm soát lỗi 2 chiều : VRC-LRC.

Bên nhận sẽ kiểm tra parity theo cả hai chiều để phát hiện và định vị lỗi cho từng ký tự. ($1 \oplus 1 = 0$ $0 \oplus 0 = 0$ $1 \oplus 0 = 0$ $0 \oplus 1 = 1$)

3.3.2 Tính theo đa thức chuẩn

Cách tính check sum như sau :

- Giả sử ta nhận được bản tin M(x).

- Nếu đa thức chuẩn G(x) có bậc là r, ta bổ sung thêm r bit 0 vào cuối bản tin và được m+r bit tương ứng đa thức xrM(x).
- Chia xrM(x) theo module 2 cho G(x). Kết quả ta được số dư T(x) là checksum được phát đi.

Các đa thức chuẩn thường được dùng để tính biến kiểm tra tổng là :

CRC-12 = $x^{12} + x^{11} + x^3 + x^2 + x + 1$ (dùng cho ký tự 6 bit)

CRC-16 = $x^{16} + x^{15} + x^2 + 1$ (dùng cho ký tự 8 bit)

CRC-CCTTT= $x^{16} + x^{12} x^5 + 1$ (dùng cho ký tự 8 bit)

Ví dụ Khung tin ban đầu 1101011011, G(x) = $x^4 + x + 1$, vậy r = 4, chuỗi bit thêm : 10011. Ta có xrM(x) = 1101011011 0000. Chia xrM(x) theo module 2 cho G(x), ta được thông số kiểm tra tổng T(x) = 1110

```
   11010'1'1011 0'00'0'
⊕  10011
   01001 1
    10011
     0000010110
           10011
           0010100
              10011
               001110 → Số dư là 1110
```

Khung tin được truyền đi: 1101011011 1110

3.3.3 Mã sửa sai

Để sửa sai một bit, ta dùng tập mã Hamming dựa trên các "bit chẵn lẻ" được rải vào các bit số liệu trong từng byte theo nguyên lý cân bằng chẵn lẻ để chỉ ra các bit lỗi.

Nếu trong bản tin có k bit và số "bit chẵn lẻ" là r, thì số bit tin và "bit chẵn lẻ" phát đi sẽ là n=k+r. r bit kiểm tra luôn các vị trí 1, 2, 4, 8,..., 2r-1 và được tạo bởi cộng module 2 giá trị nhị phân của các vị trí có bit '1' của từ mã. Vì các bit kiểm tra chiếm vị trí 2^i với i = 0, 1, 2,..., r-1 nên độ dài cực đại của các từ mã Hamming là n $\leq 2^r - 1$ và từ dãy số cực đại của các bit tin được bảo vệ là : k $\leq (2^r - 1 - r)$. Từ đây ta xác định được r.

Ví dụ: Bản tin 11 bit (10101011001) được bảo vệ bởi mã Hamming.

Từ điều kiện 11 ≤ 2' - 1-r, ta cần 4 bit kiểm tra (r =4) để tạo mã Hamming (n=11+4=15)

1	0	1	0	1	0	1	C	1	0	0	C	1	C	C
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Các bít kiểm tra C được tính như sau:

VÞ trÝ bit 1 Gi¸ trÞ nhÞ ph©n

15	1111
13	1101
11	1011
9	1001
7	0111
	1101

-> TËp m· c¸c bit kiÓm tra Hamming: 0100

Tõ m· Hamming: 10101010010011100

NÕu b¶n tin nhËn ®-îc lµ:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
1	0	1	0	0	0	1	0	1	0	0	1	1	0	0

↑ bit error

Vị trí bit 1	Giá trị nhị phân
15	1111
13	1101
9	1001
7	0111
4	0100
3	0011
	1011 (11)

→ Vb trÝ sai ë bit 11

3.4 Thủ tục liên kết dữ liệu cơ bản

Để truyền tin có độ tin cậy cao ta dùng dịch vụ liên kết (Connection Oriented Service).

Ví dụ máy A gửi số liệu cho máy B, khi tầng 2 đã được nối, số liệu từ tầng 3 máy A chuyển xuống tầng 2 nhờ chương trình con “FromNetworkLayer”. Tầng 2 bổ sung phần đầu thông tin điều khiển và tính cờ kiểm tra tổng (FCS).

Khung tin (Frame) =	Đầu tin (Header)	Thông tin (Information)	FCS

Khung tin được phát sang tầng 2 máy B nhờ chương trình con ToPhysicalLayer.

Máy B đợi tin bằng chương trình con Procedure CallWait(Event). Khi khung tin tới bên nhận, máy B tính cờ kiểm tra tổng, nếu không đúng cờ sẽ báo event = CKsumErr, nếu khung tin đúng nó báo event=FrameArrival và thu nhận khung tin từ tầng Vật lý nhờ chương trình con FromPhysicalLayer.

Sau đó đầu tin chứa các thông tin điều khiển (*header*) sẽ được kiểm tra và nếu tất cả đều đúng cả, phần số liệu được chuyển lên tầng 3 nhờ chương trình con ToNetworkLayer.

- Giao thức đơn công với kênh không lỗi và không chờ : Trong giao thức này do tin chỉ truyền theo một hướng, đường kênh không có lỗi nên số liệu luôn sẵn sàng không phải chờ.
- Giao thức đơn công với kênh không lỗi và phải đợi : Bên thu bộ nhớ hạn chế và tốc độ vật lý hữu hạn, do đó bên phát phải chờ.

3.4.1 Giao thức đơn công với kênh có lỗi

- *Bên nhận*

Khi nào đường kênh có lỗi, bên nhận sẽ chỉ gửi tín hiệu biên nhận nếu gói tin nhận được là đúng, nếu gói tin nhận được là sai thì sẽ bị bỏ đi. Quá thời hạn qui định, bên phát sẽ gửi lại gói tin. Quá trình này lặp lại cho đến khi nhận được gói tin đúng. Trong trường hợp này, tầng 3 ở máy B không biết được gói tin bị mất hay nhận hai lần, tầng 2 phải nhận biết được điều này.

Có thể xảy ra các trường hợp :

- Tầng 3 ở máy A gửi gói tin X xuống tầng 2 của nó và phát đi.
- Máy B nhận được và trả lời bằng tín hiệu biên nhận ACK.
- Tín hiệu biên nhận bị mất trên đường đi.
- Quá thời gian qui định mà máy A không nhận được tín hiệu biên nhận, nó sẽ phát lại gói tin X. Dẫn đến máy B nhận được hai gói tin X

Để giải quyết vấn đề này người ta đánh dấu gói tin gửi đi và bên nhận gửi tín hiệu cho biết đã nhận gói tin số mấy.

- *Bên phát*

Bên phát sau khi phát gói tin, có 3 khả năng xảy ra: nhận được tín hiệu biên nhận đúng, tín hiệu biên nhận bị mất hoặc quá thời gian mà chưa nhận được trả lời. Nếu tín hiệu biên nhận đúng, máy A nhận tiếp gói tin từ tầng mạng đặt vào vùng đệm (*buffer*), xoá gói tin trước, tăng số thứ tự gói tin phát. Nếu tín hiệu biên nhận bị mất hoặc đã quá thời gian mà chưa nhận được thì phát lại gói tin với số thứ tự gói tin không thay đổi.

Bên nhận nếu nhận đúng gói tin thì tiếp nhận và chuyển đến tầng mạng và phát tín hiệu biên nhận. Nếu gói tin sai hoặc nhận 2 lần thì không được chuyển lên tầng mạng.

3.5 Điều khiển dòng truyền

Để tận dụng đường dây, các tín hiệu biên nhận (ACK) được ghép cùng với gói tin. Khi gói tin đến, thay cho việc trả lời ngay tín hiệu biên nhận, bên thu nhận tiếp gói tin từ tầng mạng để ghép cùng cùng tín hiệu biên nhận và gửi trả lời. Kỹ thuật này được gọi là Piggybacking (ghép thêm).

Ưu điểm của phương pháp này là tận dụng đường kênh. Nếu quá thời gian (vài μs) mà không có gói tin mới thì bên thu cũng phải trả lời tín hiệu biên nhận để bên phát không phải phát lại gói tin cũ.

Để tận dụng đường kênh, bên phát và bên thu phải đồng bộ để bên thu kịp nhận các gói tin và bên phát cũng không lãng phí đường truyền, người ta dùng cơ chế cửa sổ trượt (sliding windows). Cửa sổ mở to thì số gói tin đưa lên đường kênh nhiều hơn(tốc độ nhanh), cửa sổ mở bé thì số gói tin đưa lên đường kênh ít lại (tốc độ chậm lại). Tương tự như cửa chắn đập nước.

3.5.1 Cơ chế cửa sổ

Người ta dùng số bit để đặc trưng cho độ rộng cực đại của cửa sổ. Trong thủ tục này, mỗi gói tin đi sẽ được đánh số từ 0 đến Max (Max là 2^n - 1) thông qua một dãy gồm các số 0, 1. Chẳng hạn cửa sổ 3 bit sẽ quản lý các gói tin có số từ 0 $\rightarrow$7. Ta có thể dùng *n* tuỳ ý.

Danh sách các gói tin gửi đi giữ trong cửa sổ phát. Danh sách các gói tin nhận được giữ trong cửa sổ nhận. Cửa sổ phát và nhận không bắt buộc phải có kích thước, giới hạn trên và dưới giống nhau.

Mặc dầu thủ tục này cho phép tầng liên kết dữ liệu linh hoạt hơn về thứ tự gởi, nhận gói tin nhưng nó yêu cầu phải đảm bảo tầng mạng đích ở bên nhận có cùng thứ tự với tầng mạng nguồn ở bên gửi.

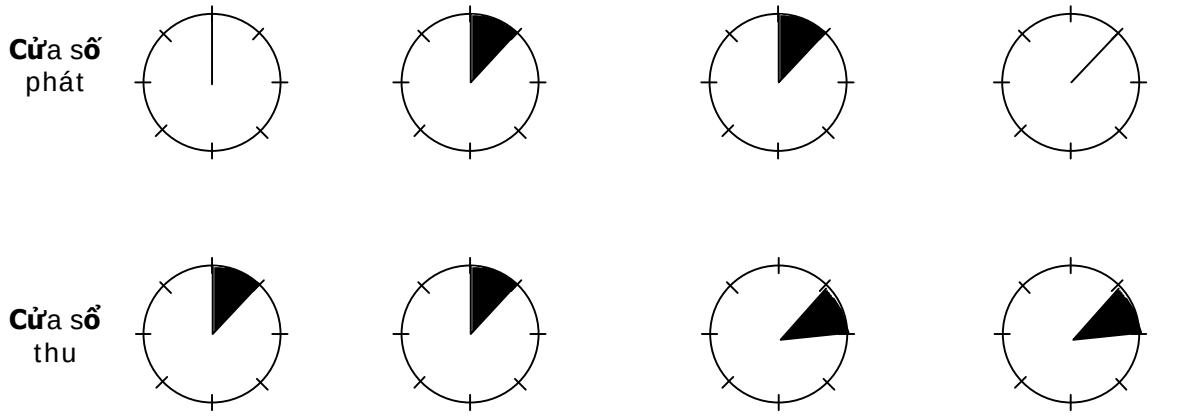
- *Cửa sổ bên phát*

Trong cửa sổ bên phát đặt các gói tin gửi đi nhưng chưa nhận được tín hiệu biên nhận. Khi nhận được gói tin mới đến từ tầng mạng để phát đi, biên trên cửa sổ tăng 1, và khi có tín hiệu biên nhận, biên dưới của cửa sổ tăng 1. Bên phát luôn giữ trong bộ nhớ các gói tin đã phát đi nhưng chưa nhận được tín hiệu biên nhận vì có thể phát lại. Như vậy nếu Max bằng n thì bên phát cần n vùng đệm để giữ các gói tin đã phát đi nhưng chưa nhận được trả lời. Nếu cửa sổ đã tới Max thì tầng liên kết dữ liệu bên phát ngừng nhận tin từ tầng 3 cho đến khi có bộ đệm tự do.

- *Cửa sổ bên nhận*

Cửa sổ bên nhận chứa các gói tin được chuyển đến. Khi gói tin có số thứ tự trùng với biên dưới của cửa sổ được nhận, cửa sổ chuyển tin lên tầng ba, phát tín hiệu biên nhận và quay một đơn vị. Không như cửa bên phát, cửa sổ bên nhận luôn duy trì cùng một kích thước. Khi kích thước cửa sổ = 1, tầng 2 nhận gói tin theo thứ tự. Nhưng nếu kích thước cửa sổ lớn hơn thì không phải như vậy.

Hoạt động của cửa sổ có kich thước là 3 bit với độ trượt 1 bit như sau :

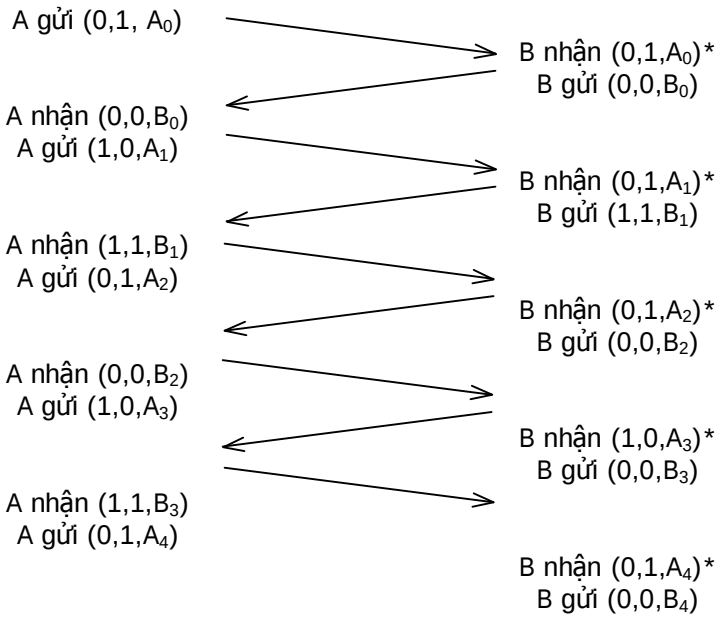


Hình 3-2. điều khiển dòng truyền theo cơ chế cửa sổ.

3.5.2 Trao đổi bản tin với cửa sổ 1 bit

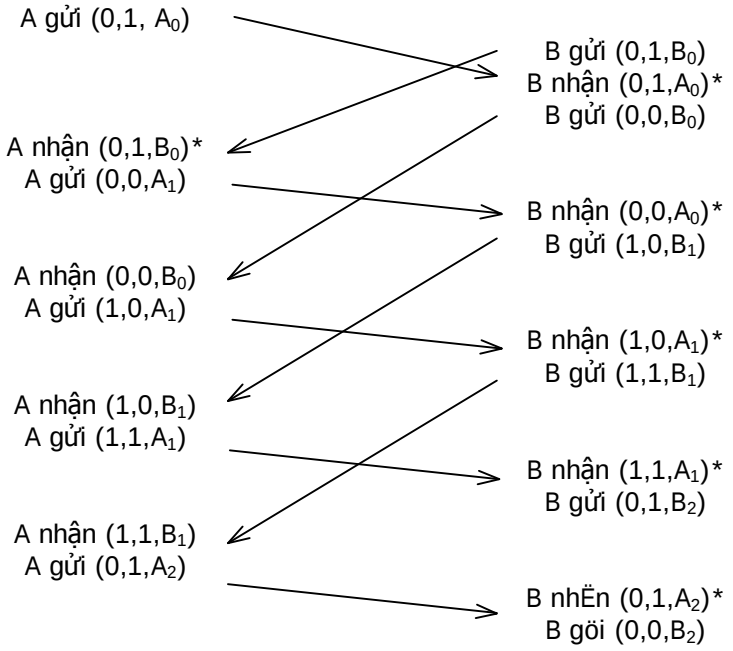
Bản tin gồn có gói tin với phần điều khiển (Header). Phần điều khiển gồm có số gói tin, số thứ tự phát *seq*, số gói tin, số thứ tự nhận là *ack*.

Trong trường hợp bình thường máy A gửi trước như sau :



Hình 3-3. Trao đổi bản tin với cửa sổ 1 bit bình thường.

Trong trường hợp bất thường máy A và B cùng gửi như sau :



Hình 3-4. Trao đổi bản tin với cửa sổ 1 bit bất thường.

Máy A ở tầng 2 nhận gói tin ở tầng 3, tạo bản tin và gửi đi. Khi bản tin này đến tầng 2 máy B, nó sẽ được kiểm tra xem có bị lặp lại không. Nếu đúng là bản tin đang mong đợi thì nó được chuyển lên tầng 3 và cửa sổ nhận dịch đi 1 nấc.

Vùng tín hiệu biên nhận chứa số bản tin cuối cùng đã được nhận mà không có lỗi. Nếu số này trùng với số bản tin vừa gửi. Bên phát sẽ lấy bản tin tiếp theo từ tầng mạng. Nếu số không đúng nó phải gửi lại bản tin cũ.

3.5.3 Vận chuyển liên tục

Thực tế cho ta thấy thời gian từ lúc phát gói tin đến lúc nhận trả lời biên nhận ACK là không đáng kể. Khi đó, nếu đường kênh vệ tinh có tốc độ 50Kbp/s với trễ lan truyền 500 ms, ta dùng thủ tục điều khiển dòng truyền gửi gói tin là 1000 bit qua vệ tinh. Thời gian phát gói tin là 20ms, vậy sau 520ms mới nhận được tín hiệu biên nhận trả lời. Như vậy bên phát phải chờ đến 96% thời gian (500/520), chỉ có 4% độ rộng băng được dùng đến.

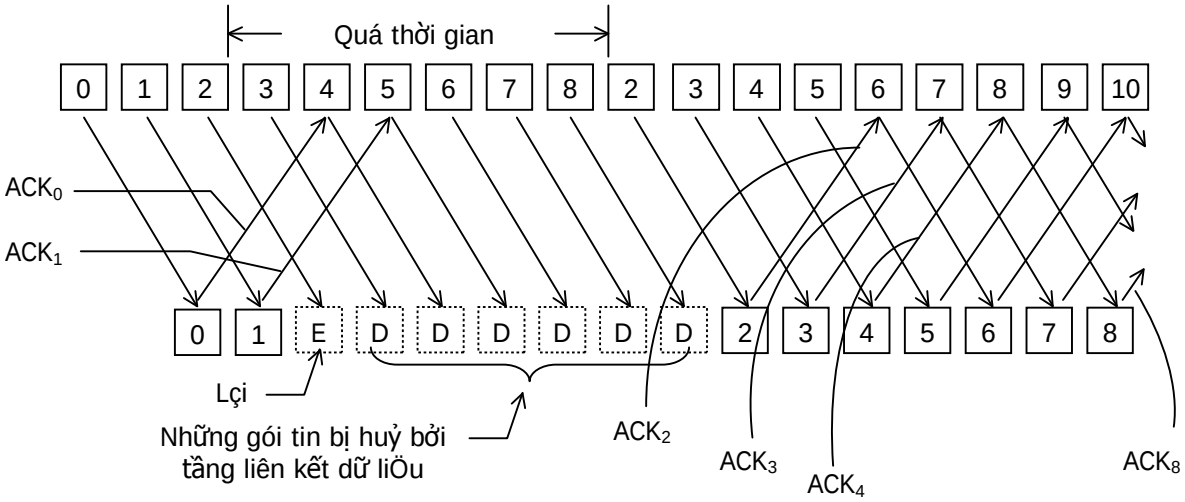
Để nâng cao hiệu suất đường truyền ta không chờ tín hiệu biên nhận mà cứ phát tiếp. Ví dụ, với thời gian phát 20ms cho một gói tin, ta sẽ gửi liên tục 26 gói tin. Như thế khi gửi hết 26 gói tin thì mất khoảng thời gian là 520 ms, đúng lúc tín hiệu biên nhận cho gói tin 0 cũng vừa đến. Kỹ thuật này gọi là Pipe-Lining (vận chuyển liên tục).

Khi có gói tin ở đoạn giữa bị hỏng thì làm thế nào ?, có bỏ những gói tin đúng đi tiếp sau nó không?. Có hai phương pháp như sau :

- Phát lại tất cả các gói tin kể từ gói tin hỏng (*go back n*)
- Phát lại chỉ riêng gói tin bị hỏng, còn gọi là phát có chọn lọc .
- Phát lại từ gói tin hỏng

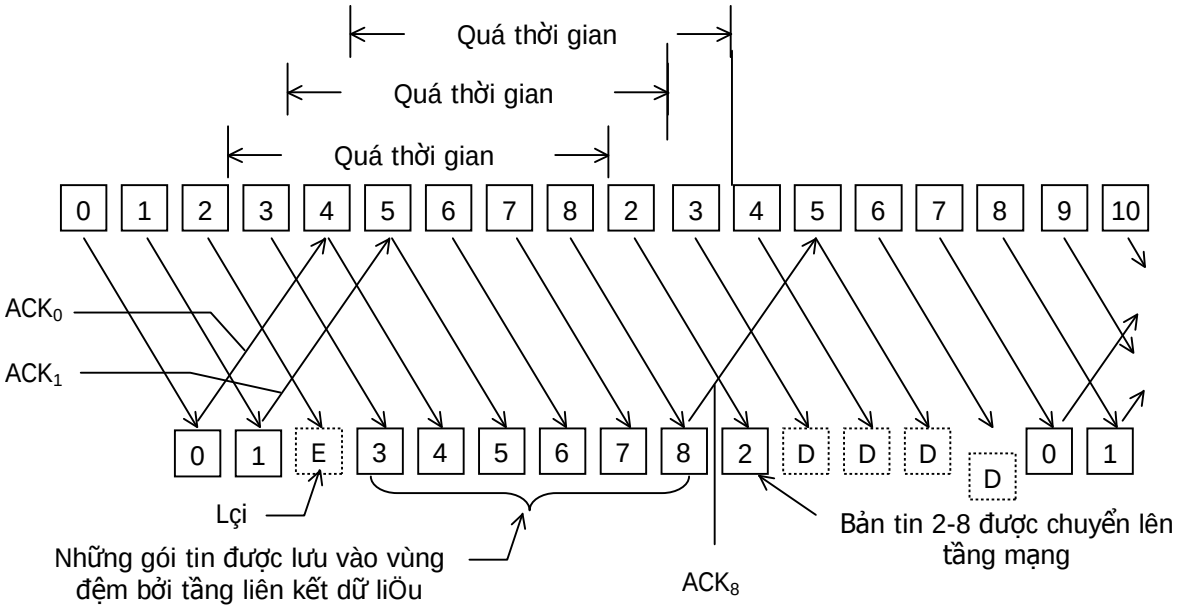
Trong trường hợp này, bên thu huỷ bỏ các gói tin tiếp theo gói tin bị hỏng. Bên phát phát lại tất cả các gói tin chưa được biên nhận bắt đầu từ gói tin bị hỏng.

Phương pháp này lãng phí đường truyền vì phải phát lại nhiều gói tin.



Hình 3-5. Cơ chế vận chuyển liên tục.

3.5.3.1 Phát lại có chọn läc



Hình 3-6. Cơ chế phát bản tin có chọn lọc.

Trong phương pháp này, các gói tin nhận được có thể không theo thứ tự nhưng sẽ được sắp xếp lại để chuyển lên tầng mạng theo đúng thứ tự. Khi có gói tin bị lỗi, bên thu tiếp tục thu các gói tin đúng sau gói tin hỏng ở tầng 2. Bên phát chỉ phát lại

gói tin hỏng. Phương pháp này ứng với cửa sổ bên thu lớn hơn 1 và đòi hỏi bộ nhớ lớn để giữ các gói tin sau gói tin hỏng.

3.6 Các giao thức của tầng Liên kết dữ liệu

Tầng LKDL cung cấp các phương tiện để truyền thông tin qua liên kết vật lý đảm bảo tin cậy thông qua các cơ chế đồng bộ hóa, kiểm soát lỗi và kiểm soát luồng dữ liệu. Các giao thức được xây dựng cho tầng LKDL (DLP - Data Link Protocol) được phân thành hai loại :

1. Giao thức dị bộ (asynchronous DLP) : Cho phép một ký tự dữ liệu được truyền đi bất kỳ lúc nào mà không cần quan tâm đến các tính hiệu đồng bộ trước đó.
2. Giao thức đồng bộ (synchronous DLP) : Chèn các ký tự điều khiển hoặc các cờ giữa các dữ liệu của người sử dụng để báo cho bên nhận. Có hai nhóm giao thức đồng bộ :
    a. Đồng bộ hướng ký tự (character -oriented)
    b. Đồng bộ hướng bit (bit - oriented)

Các hệ thống truyền thông đòi hỏi hai mức đồng bộ hóa :

- Mức vật lý : để giữ đồng bộ giữa các đồng hồ người gửi và người nhận
- Mức LKDL : để phân biệt dữ liệu của người sử dụng với các 'cờ'và các vùng thông tin điều khiển khác

Sau đây ta xét hai loại giao thức đồng bộ là giao thức truyền tin đồng bộ nhị phân BSC (Binary Synchronous Control) và giao thức điều khiển liên kết dữ liệu mức cao HDLC (Highlevel Data Link Control).

3.6.1 Giao thức BSC

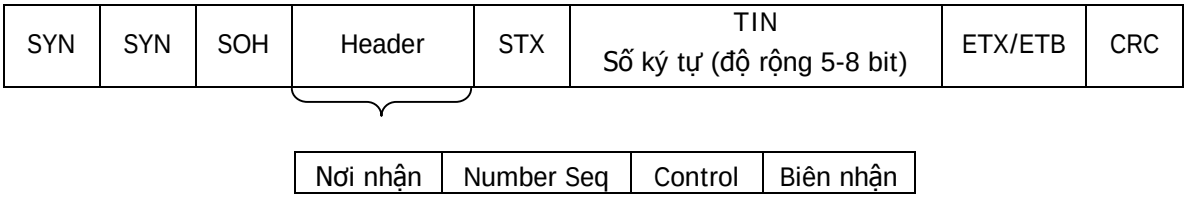
Đây là giao thức *hướng ký tự* (COP - Character Oriented Protocol) được xây dựng dựa trên các ký tự đặc biệt của một bộ mã chuẩn nào đó (như ASCII hoặc EBCDIC) hoạt động theo phương thức hai chiều luân phiên.

3.6.1.1 Tập ký tự điều khiển

ENQ (05): Enquire - Yêu cầu trả lời từ một trạm xa
ACK (06): Acknowledgement - Thông báo tiếp nhận tốt thông tin
NAK (15): Negative ACK - Thông báo tiếp nhận không tốt thông tin
STX (02): Start of text - Kết thúc phần Header và bắt đầu phần dữ liệu
ETX (03): End of text - Kết thúc phần dữ liệu
ETB (17): End of transmission block - Kết thúc đoạn tin (khối dữ liệu)

SOH (01): Start of heading - Bắt đầu phần header của bản tin
EOT (04): End of transmission - Kết thúc quá trình truyền tin và giải phóng liên kết
DLE (10): Data Link Escape - Để thay đổi ý nghĩa của các ký tự điều khiển truyền tin khác
SYN (16): Synchronous - Ký tự đồng bộ bản tin dùng để duy trì đồng bộ giữa 2 bên

3.6.1.2 Khuôn dạng tổng quát bản tin của giao thức BSC

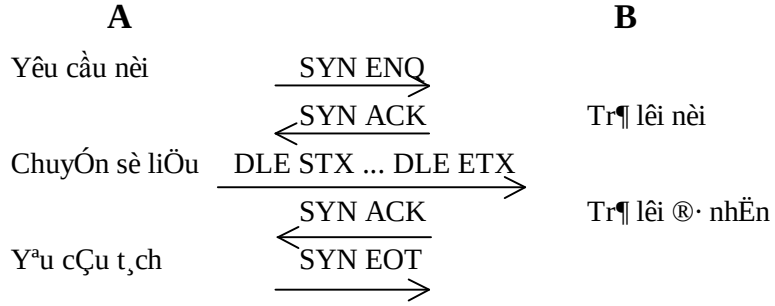


Để thông suốt bản tin, có thể dùng thêm các byte đệm :

SYN	SYN		DLE	DLE		DLE	ETX/ETB	CRC

Khi phát nếu ký tự phát trùng với DLE thì ta chèn thêm DLE. Khi thu, DLE chèn thêm sẽ được khử bỏ.

Ví dụ về thủ tục BCS



3.6.2 Giao thức HDLC

HDLC là giao thức hướng bit (Bit Oriented Protocol - BOP) có các phần tử của giao thức (đơn vị dữ liệu, thủ tục) được xây dựng từ các cấu trúc nhị phân (xâu bit) và khi nhận dữ liệu sẽ được tiếp nhận lần lượt từng bit một.

Đây là giao thức có vị trí quan trọng nhất, được ISO phát triển để sử dụng trong cả hai trường hợp : điểm - điểm và nhiều điểm, cho phép truyền thông hai chiều đồng thời.

3.6.2.1 Khuôn dạng tổng quát bản tin của giao thức HDLC

<--- *Hướng truyền*

	8 bits	8 bits	128-1024 bytes		
Flag	HEADER		INFORMATION	FCS	Flag
01111110	Address	Control	Số các bit	(16 bit)	01111110

Trong đó :

- *Flag* (01111110): là cờ dùng để nhận biết điểm bắt đầu và kết thúc bản tin.

Để tránh sự xuất hiện của mã cờ trong nội dung của bản tin, người ta cài đặt cơ chế '*cứng*' có các chức năng sau :

- Khi truyền tin cứ sau năm bit 1 liên tiếp thì thêm một bit 0 để không nhầm với Flag : 01101111111110010

  01101111011110010

  ↑ *bit chèn thêm (khi thu thì bit này sẽ được khử bỏ)*
- Khi nhận tin, nếu phát hiện có bit 0 sau 5 bit 1 liên tiếp thì tự động loại bỏ bít 0 đó đi.

- *Address* : vùng chứa địa chỉ trạm đích của khung tin.
- *Information* : vùng ghi thông tin truyền đi, có kích thước không xác định.
- *FCS (Frame Check Sequence)* : vùng để ghi mã kiểm soát lỗi (checksum) cho nội dung khung tin, dùng phương pháp CRC với đa thức sinh là CRC-CCITT = $x^{16} + x^{12} + x^5 + 1$
- *Control* : vùng định danh cho các loại khung tin khác nhau của HDLC, có ba dạng như sau :

  Dạng I : hiệu lực truyền tin tức - Information
  Dạng S : hiệu lực điều hành sự nối - Supervisor
  Dạng N : chức năng phụ của điều hành nối – Unnumbered

3.6.2.2 Phương thức trao đổi thông tin

Giao thức HDLC có 3 phương thức trao đổi thông tin chính, ứng với mỗi phướng thức có các giao thức khung tin tương ứng là SNRM, SARM hoặc SABM :

- *Phương thức trả lời chuẩn SNRM (Set Normal Response Mode):* Được sử dụng trong trường hợp cấu hình không cân bằng, có một trạm điều khiển chung (master), các trạm còn lại (slave) chỉ có thể truyền tin khi trạm chủ cho phép.
- *Phương thức trả lời dị bộ SARM (Set Asynchronous Response Mode):* Cũng được sử dụng trong trường hợp cấu hình không cân bằng như trường hợp trên, nhưng các trạm slave được phép truyền tin mà không cần sự cho phép của trạm master. Phương thức này được sử dụng trong trường hợp điểm-điểm với liên kết 2 chiều, cho phép trạm slave gửi các gói tin (frame) không đồng bộ với trạm master.

- *Phương thức trả lời dị bộ cân bằng SABM* (Set Asynchronous Balanced Mode) : Sử dụng trong trường hợp điểm-điểm, liên kết 2 chiều. Trong đó các trạm đều có vai trò tương đương.

3.6.2.3 Các giao thức dẫn xuất của HDLC

- LAP (Link Access Procedure) : tương ứng với phương thức trả lời dị bộ (ARM).
- LAPB (Link Access Protocol-Balanced) : tương ứng với phương thức trả lời dị bộ cân bằng (ABM), được dùng hầu hết trong các mạng truyền dữ liệu công cộng X25.
- LAP-D (Link Access Procedure, D Channel) : Được xây dựng từ LAP-B và được dùng như giao thức liên kết dữ liệu cho các mạng ISDN
- SDLC, ADCCP

3.6.2.4 So sánh BOP và COP

- BOP nhận lần lượt từng bit một, do đó mềm dẻo, dễ dàng tương thích với các hệ khác nhau.
- BOP có overhead (phụ trội) ngắn, số bít bổ sung và số tín hiệu điều khiển ít do đó có tốc độ cao.
- Thủ tục điều khiển trên bit nhị phân đảm bảo không phụ thuộc mã dùng. Cách giải quyết này mềm dẻo và cho phép giải quyết vô số yêu cầu khác.
- Thủ tục HDLC được coi là chuẩn quốc tế và sẽ thông trị trong thời gian tới, nó thích ứng với các hệ thống phức tạp. Đối với các thiết bị ít phức tạp có thể dùng HDLC đơn giản hoá để đảm bảo sự tương thích với HDLC và sự phát triển mở rộng hệ thống sau này.

BÀI TẬP

1. Tìm hiểu thêm về chuẩn giao tiếp RC232 và các chuẩn khác được phát tiển từ chuẩn này.
2. Tìm hiểu các chuẩn mở rộng của giao thức HDLC.

-

Chương 4

MẠNG CỤC BỘ

Mạng cục bộ (LAN) là hệ truyền thông tốc độ cao được thiết kế để kết nối các máy tính và các thiết bị xử lý dữ liệu khác cùng hoạt động với nhau trong một khu vực địa lý nhỏ như ở một tầng của toà nhà, hoặc trong một toà nhà.... (100m đến vài km), có tốc độ truyền dữ liệu cao (có thể tới 100Mbps), tỷ lệ sai số dữ liệu nhỏ (10^{-8} ->10^{-11}). Một số mạng LAN có thể kết nối lại với nhau trong một khu làm việc.

Mạng LAN thường bao gồm một hoặc một số máy chủ (file server, host), còn gọi là máy phục vụ) và một số máy tính khác gọi là trạm làm việc (Workstations, Client) hoặc còn gọi là nút mạng (Network Node) - một hoặc một số máy tính cùng nối vào một thiết bị nút.

4.1 Các cấu hình của mạng LAN

Cấu hình (topology) của mạng là cấu trúc hình học không gian mà thực chất là cách bố trí phần tử của mạng cũng như cách nối giữa chúng với nhau. Thông thường mạng có 3 dạng cấu trúc là: Mạng dạng hình sao (Star Topology), mạng dạng vòng (Ring Topology) và mạng dạng tuyến (Linear Bus Topology). Ngoài 3 dạng cấu hình kể trên còn có một số dạng khác biến tướng từ 3 dạng này như mạng dạng cây, mạng dạng hình sao - vòng, mạng hỗn hợp,v.v....

4.1.1 Mạng dạng hình sao (Star Topology)

Mạng dạng hình sao bao gồm một trung tâm và các nút thông tin. Các nút thông tin là các trạm đầu cuối, các máy tính và các thiết bị khác của mạng. Trung tâm của mạng điều phối mọi hoạt động trong mạng với các chức năng cơ bản là:

- Xác định cặp địa chỉ gửi và nhận được phép chiếm tuyến thông tin và liên lạc với nhau.
- Cho phép theo dõi và xử lý sai trong quá trình trao đổi thông tin.
- Thông báo các trạng thái của mạng...

Ưu điểm :

- Hoạt động theo nguyên lý nối song song nên nếu có một thiết bị nào đó ở một nút thông tin bị hỏng thì mạng vẫn hoạt động bình thường.
- Cấu trúc mạng đơn giản và các thuật toán điều khiển ổn định.
- Mạng có thể mở rộng hoặc thu hẹp tuỳ theo yêu cầu của người sử dụng.

Nhược điểm:

- Khả năng mở rộng mạng hoàn toàn phụ thuộc vào khả năng của trung tâm. Khi trung tâm có sự cố thì toàn mạng ngừng hoạt động.

- Mạng yêu cầu nối độc lập riêng rẽ từng thiết bị ở các nút thông tin đến trung tâm. Khoảng cách từ máy đến trung tâm rất hạn chế (100 m).

Nhìn chung, mạng dạng hình sao cho phép nối các máy tính vào một bộ tập trung (HUB) bằng cáp xoắn, giải pháp này cho phép nối trực tiếp máy tính với HUB không cần thông qua trục BUS, tránh được các yếu tố gây tắc nghẽn mạng. Gần đây, cùng với sự phát triển switching hub, mô hình này ngày càng trở nên phổ biến và chiếm đa số các mạng mới lắp.

4.1.2 Mạng hình tuyến (Bus Topology)

Theo cách bố trí hành lang các đường như hình vẽ thì máy chủ (host) cũng như tất cả các máy tính khác (workstation) hoặc các nút (node) đều được nối về với nhau trên một trục đường dây cáp chính để chuyển tải tín hiệu.

Tất cả các nút đều sử dụng chung đường dây cáp chính này. Phía hai đầu dây cáp được bịt bởi một thiết bị gọi là terminator. Các tín hiệu và gói dữ liệu (packet) khi di chuyển lên hoặc xuống trong dây cáp đều mang theo địa chỉ của nơi đến.

Loại hình mạng này dùng dây cáp ít nhất, dễ lắp đặt. Tuy vậy cũng có những bất lợi đó là sẽ có sự ùn tắc giao thông khi di chuyển dữ liệu với lưu lượng lớn và khi có sự hỏng hóc ở đoạn nào đó thì rất khó phát hiện, một sự ngừng trên đường dây để sửa chữa sẽ ngừng toàn bộ hệ thống.

4.1.3 Mạng dạng vòng (Ring Topology)

Mạng được bố trí theo dạng vòng tròn, đường dây cáp được thiết kế làm thành một vòng khép kín, tín hiệu chạy quanh theo một chiều nào đó. Các nút truyền tín hiệu cho nhau mỗi thời điểm chỉ được một nút mà thôi. Dữ liệu truyền đi phải có kèm theo địa chỉ cụ thể của mỗi trạm tiếp nhận.

Mạng Token Ring có thể chạy ở tốc độ 4Mbps hoặc 16Mbps. Phương pháp truy cập dùng trong mạng Token Ring gọi là Token passing. Token passing là phương pháp truy nhập xác định, trong đó các xung đột được ngăn ngừa bằng cách ở mỗi thời điểm chỉ một trạm có thể được truyền tín hiệu. Điều này được thực hiện bằng việc truyền một bó tín hiệu đặc biệt gọi là Token (mã thông báo) xoay vòng từ trạm này qua trạm khác. Một trạm chỉ có thể gửi đi bó dữ liệu khi nó nhận được Token, khi đó nó sẽ chiếm được quyền ưu tiên hoạt động trên mạng.

Mạng dạng vòng có thuận lợi là có thể nới rộng ra xa, tổng đường dây cần thiết ít hơn so với hai kiểu trên. Nhược điểm là đường dây phải khép kín, nếu bị ngắt ở một nơi nào đó thì toàn bộ hệ thống cũng bị ngừng.

4.1.4 Mạng dạng kết hợp

4.1.4.1 Kết hợp hình sao và tuyến (star/Bus Topology)

Cấu hình mạng dạng này có bộ phận tách tín hiệu (spitter) giữ vai trò thiết bị trung tâm, hệ thống dây cáp mạng có thể chọn hoặc Ring Topology hoặc Linear Bus Topology.

Ưu điểm của cấu hình này là mạng có thể gồm nhiều nhóm làm việc ở cách xa nhau, ARCNET là mạng dạng kết hợp Star/Bus Topology. Cấu hình dạng này đưa lại sự uyển chuyển trong việc bố trí đường dây tương thích dễ dàng đối với bất cứ toà nhà nào.

4.1.4.2 Kết hợp hình Sao và Vòng (Star/Ring Topology)

Cấu hình dạng kết hợp Star/Ring Topology, có một "thẻ bài" (token) được chuyển vòng quanh một cái HUB trung tâm. Mỗi trạm làm việc được nối với HUB - là cầu nối giữa các trạm làm việc và để tăng khoảng cách cần thiết.

Một hub thông thường có nhiều cổng nối với người sử dụng để gắn máy tính và các thiết bị ngoại vi. Mỗi cổng hỗ trợ một bộ kết nối dùng cặp dây xoắn 10BASET từ mỗi trạm của mạng. Khi bó tín hiệu Ethernet được truyền từ một trạm tới hub, nó được lặp lại trên khắp các cổng khác của hub. Các hub thông minh có thể định dạng, kiểm tra, cho phép hoặc không cho phép bởi người điều hành mạng từ trung tâm quản lý hub. Có ba loại hub:

- Hub đơn (stand alone hub)
- Hub modun (modular hub) : Modular hub rất phổ biến cho các hệ thống mạng vì nó có thể dễ dàng mở rộng và luôn có chức năng quản lý, modular có từ 4 đến 14 khe cắm, có thể lắp thêm các modun Ethernet 10BASET.
- Hub phân tầng (stackable hub) : thuận tiện cho những cơ quan muốn đầu tư tối thiểu ban đầu nhưng lại có kế hoạch phát triển LAN sau này.

4.2 Các giao thức điều khiển truy nhập đường truyền

Giao thức dùng để đánh giá khả năng của một mạng được phân chia bởi các trạm như thế nào. Hệ số này được quyết định chủ yếu bởi hiệu quả sử dụng môi trường truy xuất (medium access) của giao thức.

Mọi kênh phương tiện chỉ có thể hỗ trợ một lần tín hiệu. Nếu hai máy tính truyền trên kênh cùng một lúc, các tín hiệu của chúng sẽ gây nhiễu cho nhau (ví dụ như hai người cùng nói một lúc). Có hai phương pháp điều khiển việc truy nhập phương tiện để không xảy ra sự cố gây nhiễu : truy nhập ngẫu nhiên và truy nhập có điều khiển.

- *Loại truy nhập ngẫu nhiên*

Trạm có thể truy nhập phương tiện truyền tuỳ theo ý muốn, bất kỳ ở thời điểm ngẫu nhiên nào.

a. Kỹ thuật truy cập ngẫu nhiên đối với dạng bus

- Phương pháp đa truy nhập sử dụng sóng mang (CSMA - Carrier Sense Multiple Access).
- Phương pháp đa truy nhập sử dụng sóng mang với phát hiện xung đột (CSMA/CD - with Collision Detection)

b. Kỹ thuật truy cập ngẫu nhiên đối với dạng vòng

- Phương pháp chèn thanh ghi (Register insertion)
- Phương pháp vòng có ngăn (Slotted-ring)

- *Loại truy nhập có điều khiển*

Phương pháp điều khiển tranh chấp thường thích hợp với các mạng có sự trao đổi dữ liệu không liên tục và tương đối ít máy tính. Đây là dạng thông dụng trong cấu trúc mạng cục bộ.

- Kỹ thuật bus với thẻ bài (Token Bus) : dùng cho các mạng LAN
- Kỹ thuật vòng với thẻ bài (Token Ring) : dùng cho các mạng LAN
- Kỹ thuật tránh xung đột : dùng cho các mạng cục bộ tốc độ cao.

4.2.1 Phương pháp CSMA

Còn được gọi là phương pháp LBT (Listen Before Talk - Nghe trước khi nói). Một trạm có dữ liệu cần truyền trước hết phải 'nghe' xem phương tiện truyền rỗi hay bận. Nếu rỗi thì bắt đầu truyền tin, còn nếu bận thì thực hiện một trong ba giải thuật sau :

- Giải thuật '*non-persistent*' : Tram rút lui (không kiên trì) chờ đợi một thời gian ngẫu nhiên nào đó rồi lại bắt đầu 'nghe' đường truyền. Giải thuật này có hiểu quả tránh xung đột nhưng có thời gian chết.
- Giải thuật '*1-persistent*' : Trạm tiếp tục nghe đến khi phương tiện truyền rỗi thì tiến hành truyền dữ liệu đi (với xác xuất 1). Giải thuật này giảm thời gian chết, xong nếu có nhiều trạm cùng chờ và tiến hành phát dữ liệu cùng một lần thì sẽ xảy ra xung đột.
- Giải thuật '*p-persistent*' : trạm tiếp tục nghe, đến khi phương tiện truyền rỗi thì tiến hành phát tin với một xác xuất nhất định nào đó (mỗi trạm có gắn một hệ số ưu tiên). Ngược lại trạm 'rút lui' trong một thời gian cố định rồi

truyền với xác suất p hoặc tiếp tục chờ đợi với xác xuất 1-p. Giải thuật này phức tạp nhưng giảm được tối đa xung đột và thời gian chết.

Phương pháp CSMA chỉ 'nghe trước khi nói', không có khả năng phát hiện xung đột trong quá trình truyền, dẫn đến lãng phí đường truyền.

4.2.2 Phương pháp CSMA/CD

Phương pháp CSMA/CD có nguồn gốc từ hệ thống radio đã phát triển ở trường đại học Hawai vào khoảng năm 1970, gọi là ALOHANET, còn được gọi là phương pháp LWT (Listen While Talk - Nghe cả trong khi nói). Các va chạm luôn xảy ra tại một cấp nào đó trên các mạng, với số lượng gia tăng theo tỉ lệ thuận khi các phiên truyền gia tăng.

Phương pháp CSMA/CD ngoài các chức năng của CSMA còn bổ sung các quy tắc sau :

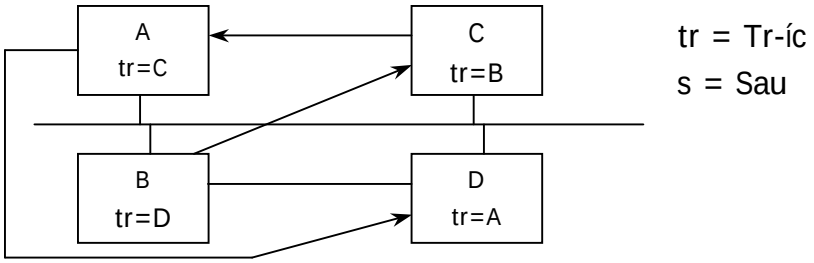
1. Khi đang truyền vẫn tiếp tục nghe đường dây.
2. Nếu phát hiện có xung đột thì *ngừng truyền* và *tiếp tục gởi sóng mang* thêm một thời gian nữa để bảo đảm các trạm đều có thể nghe được sự kiện xung đột.
3. Sau khi chờ đợi một thời gian ngẫu nhiên thì trạm thử truyền lại bằng cách sử dụng các phương pháp của CSMA.

Với phương pháp CSMA/CD thời gian chiếm dụng vô ích đường truyền giảm xuống bằng thời gian dùng để phát hiện một đụng độ. CSMA/CD sử dụng ba giải thuật 'persistent' ở trên. Trong đó giải thuật '*1-persistent*' được sử dụng trong mạng Ethernet, Mitrenet và được chọn cả trong chuẩn IEEE.802. Ngoài ra mỗi chuẩn LAN còn có thêm các cơ chế bổ sung.

4.2.3 Điều khiển truy nhập bus với thẻ bài

Các trạm trên bus tạo nên một vòng logic, được xác định vị trí theo một dãy thứ tự, trong đó trạm cuối sẽ tiếp liền ngay sau trạm đầu tiên. Mỗi trạm được biết địa chỉ của các trạm kề sau và kề trước nó.

Thẻ bài dùng cấp phát quyền truy nhập, được lưu chuyển trong vòng logic. Khi trạm nhận được thẻ bài thì được trao quyền sử dụng phương tiện trong một thời gian xác định để truyền dữ liệu. Khi truyền xong hoặc hết thời hạn, trạm sẽ chuyển thẻ bài đến trạm kế tiếp trong vòng logic. Các trạm không sử dụng thẻ bài vẫn có mặt trên bus nhưng chúng chỉ có thể trả lời cho yêu cầu xác nhận (nếu chúng là đích của gói tin nào đó). Thứ tự vật lý của trạm trên bus là không quan trọng, độc lập với thứ tự logic.



Hình 4-1. Điều khiển truy nhập bus với thẻ bài.

Các chức năng :

- Khởi tạo vòng logic : khi thiết lập mạng hoặc khi vòng logic bị gãy.
- Bổ sung trạm vào vòng logic (xem xét định kỳ) bằng cách mời nút đứng sau nhập vòng. Loại bỏ một trạm ra khỏi vòng logic bằng cách nối trạm trước và sau nó với nhau
- Quản lý sai sót : trùng địa chỉ, gãy vòng (các trạm bị treo, rơi vào trạng thái chờ lẫn nhau), bởi nút giữ Token.
- Khi đang giữ thẻ mà có trạm khác nhận được gói tin thì chứng tỏ nút khác đã có thẻ, lúc đó nó sẽ bỏ thẻ bằng cách chuyển sang trạng thái 'nghe'.
- Khi nút đã hoàn thành công việc, nó gửi thẻ đến nút đứng sau, nếu nút tiếp sau hoạt động thì nó gởi thẻ chuyển sang trạng thái bị động. Nếu ngược lại, nó gửi thẻ cho nút kế tiếp lần nữa. Nếu hai lần gửi không được thì xem như nút kế tiếp hỏng và gửi đi gói tin "tìm nút kế tiếp" để tìm nút tiếp theo.
- Nếu không thành công thì nút bị xem là có sự cố. Nút ngừng hoạt động và 'nghe' trên bus.

- Dạng bản tin của mạng Token bus

Bắt đầu tin	Điều khiển gói tin	Địa chỉ nguồn	Địa chỉ đích	TIN	FSC	Kết thúc tin
1 byte	1 byte	2-6 bytes	2 - 6 bytes		4 bytes	1 byte
	Khung tin cực đại 8191 bytes				Tốc độ có thể là 1; 5; 10Mbps	

- So sánh CSMA/CD và Token Bus

- Token bus quản lý phức tạp hơn so với CSMA/CD. Trong trường hợp tải nhẹ thì không hiệu quả bằng CSMA/CD (do phải qua nhiều trạm)
- Tuy nhiên Token Bus có hiệu quả trong trường hợp tải nặng, dễ điều hoà lưu thông trên mạng Token Bus. Không quy định độ dài tối thiểu của gói tin, không cần nghe trước khi nói.

4.2.4 Điều khiển truy nhập vòng với thẻ bài

Đây là giao thức thông dụng được dùng trong các LAN có cấu trúc vòng (Ring). Phương pháp này sử dụng một khối tín hiệu đặc biệt gọi là Token di chuyển vòng quanh mạng theo một chiều xác định. Một trạm muốn truyền phải đợi cho đến khi nhận được thẻ bài. Khi một trạm đang chiếm Token thì nó có thể phát đi một gói dữ liệu. Khi đã phát hết gói dữ liệu cho phép hoặc không còn gì để phát nữa thì trạm đó chuyển khung thẻ bài đến cho trạm kế tiếp trên mạng. Trong token có chứa một địa chỉ đích và được luân chuyển tới các trạm theo một trật tự đã định trước. Đối với cấu hình mạng dạng xoay vòng thì trật tự của sự truyền token tương đương với trật tự vật lý của các trạm xung quanh vòng.

Các chuẩn mạng sử dụng phương pháp điều khiển truy nhập thẻ bài :

- Chuẩn IEEE 802.5, còn gọi là chuẩn Token Ring.
- FDDI là chuẩn sợi quang 100 Mps sử dụng phương pháp chuyển thẻ bài và vòng tròn.

Phương pháp chuyển thẻ bài thích hợp trong các điều kiện như sau :

- Khi mạng đang tải dữ liệu quan trọng về thời gian do phương pháp này cung cấp khả năng bàn giao.
- Khi mạng được sử dụng nhiều, do tránh được xung đột.
- Khi một vài trạm có mức ưu tiên cao hơn so với các trạm khác. Phương pháp chuyển thẻ bài có thể áp dụng các mức ưu tiên cho trạm để ngăn cấm một trạm bất kỳ không được độc quyền về mạng.
- Do thẻ bài luân chuyển quanh mạng nên mỗi trạm có thể truyền theo quãng thời gian tối thiểu.

Phương pháp chuyển thẻ bài đòi hỏi cơ chế điều khiển phức tạp và chi phí đầu tư phần cứng cao, nhưng được thiết kế với độ tin cậy cao. Tuy vậy hiện nay Ethernet vẫn là chuẩn LAN thông dụng, chứng tỏ được ưu điểm của phương pháp tranh chấp khi sử dụng trên các mạng LAN.

Giao thức truyền token có trật tự hơn nhưng cũng phức tạp hơn CSMA/CD, có ưu điểm là vẫn hoạt động tốt khi lưu lượng truyền thông lớn. Giao thức truyền token tuân thủ đúng sự phân chia của môi trường mạng, hoạt động dựa vào sự xoay vòng tới các trạm. Việc truyền token sẽ không thực hiện được nếu việc xoay vòng bị đứt đoạn. Giao thức phải chứa các thủ tục kiểm tra token để cho phép khôi phục lại token bị mất hoặc thay thế trạng thái của token và cung cấp các phương tiện để sửa đổi logic (thêm vào, bớt đi hoặc định lại trật tự của các trạm).

Khung tin cực đại là 16KB ở chế độ truyền 16Mbps và 4KB ở chế độ truyền 4Mps.

Dạng bản tin với mạng Token Ring :

Bắt đầu tin	Điều khiển thâm nhập	Điều khiển gói tin	Địa chỉ nguồn	Địa chỉ đích	TIN	FSC	Kết thúc gói tin	Trạng thái gói tin
1 byte	1 byte	2-6 bytes	2 - 6 bytes	2 - 6 bytes		4 bytes	1 byte	1 byte

4.2.4.1 Phương pháp điều khiển truy nhập dò báo

Dò báo (*polling*) là một phương pháp điều khiển truy cập sử dụng một thiết bị trung tâm để điều khiển toàn bộ việc truy cập mạng. Đây là phương pháp được sử dụng phổ dụng nhất trên các mạng máy tính lớn.

Thiết bị trung tâm có tên là thiết bị chính sẽ yêu cầu dữ liệu từ các thiết bị khác trên mạng có tên là thiết bị thứ cập (*secondaries*). Sau khi được dò báo, thiết bị thứ cấp có thể truyền một lượng dữ liệu được xác định bởi các giao thức dùng trên mạng. Một thiết bị thứ cấp không thể truyền trừ phi nó được thiết bị chính dò báo.

Phương pháp dò báo có nhiều ưu điểm của phương pháp chuyển thẻ bài như :

- Dự đoán được các lần truy cập định sẵn.
- Gán được các mức ưu tiên, tránh được va chạm.

So sánh phương pháp dò báo và phương pháp chuyển thẻ bài : kỹ thuật dò báo tập trung hóa quyền điều khiển. Nhìn dưới góc độ quản lý thì đây là một ưu điểm, nhưng nếu cơ chế điều khiển trung tâm bị hỏng, mạng sẽ ngừng hoạt động. Phương pháp chuyển thẻ bài sử dụng các chức năng điều khiển phân phối hơn do đó ít bị hỏng tập trung tại một điểm. Bên cạnh đó, phương pháp dò báo đôi khi lãng phí các lượng băng thông lớn do phải dò báo từng thiết bị thứ cấp, cho dù các thiết bị không có gì để truyền.

4.3 Chuẩn hóa mạng cục bộ

Các chuẩn LAN là các tiêu chuẩn công nghệ cho Lan được phê chuẩn bởi các tổ chức chẩun hóa quốc tế, nhằm hướng dẫn các nhà sản xuất thiết bị mạng đi đến sự thống chung khả năng sử dụng chung các sản phẩm của họ, vì lợi ích của người sử dụng và tạo điều kiện thuận lợi cho các nghiên cứu phát triển.

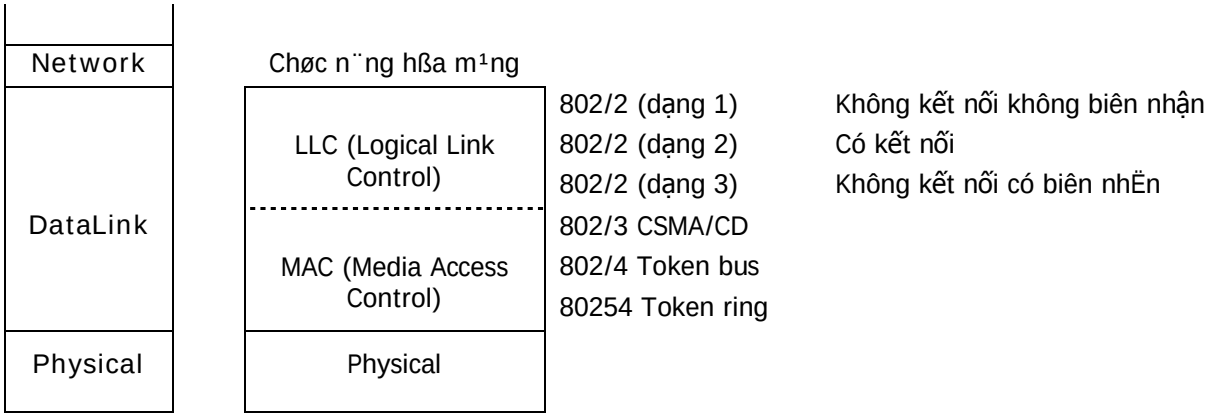
Các chuẩn này quy định môi trường truyền dẫn cũng như cách thức sử dụng chúng trong kết nối LAN; Các giao thức truyền thông ở các tầng vật lý và tầng liên kết dữ liệu của mạng theo mô hình OSI.

Các giao thức truyền thông ở các tầng trên của mô hình OSIhiện tại được xác định qua một số giao thức phổ biến như TCP/IP, IPX/SPX, NetBIOS,. . .

Uỷ ban IEEE phát triển tiêu chuẩn IEEE LAN và đề xuất phân chia hai tầng thấp nhất của mô hình OSI như dưới đây.

Theo chuẩn 802 thì tầng LKDL được chia thành 2 tầng con:

- Tầng con điều khiển logic LLC (Logical Link Control Sublayer) : giữ vai trò tổ chức dữ liệu, tổ chức thông tin để truyền và nhận. Thủ tục tầng LLC không bị ảnh hưởng khi sử dụng các đường truyền dẫn khác nhau, nhờ vậy mà linh hoạt hơn trong khai thác.
- Tầng con điều khiển xâm nhập mạng MAC (Media Access Control Sublayer). làm nhiệm vụ điều khiển việc xâm nhập mạng.



Hình 4-2. Các tầng con LLC và MAC.

Chuẩn 802.2 ở mức con LLC tương đương với chuẩn HDLC của ISO hoặc X.25 của CCITT.

Chuẩn 802.3 xác định phương pháp thâm nhập mạng tức thời có khả năng phát hiện lỗi chồng chéo thông tin CSMA/CD. Phương pháp CSMA/CD được đưa ra từ năm 1993 nhằm mục đích nâng cao hiệu quả mạng. Theo chuẩn này các mức được ghép nối với nhau thông qua các bộ ghép nối.

Chuẩn IEEE 802.3 dùng cho mạng Ethernet (sử dụng giao thức truy nhập CSMA/CD) bao gồm cả 2 phiên bản băng tần cơ bản và băng tần mở rộng.

Chuẩn IEEE 802.4 liên quan tới sự sắp xếp tuyến token, thực chất là phương pháp thâm nhập mạng theo kiểu phát tín hiệu thăm dò token qua các trạm và đường truyền bus.

Chuẩn IEEE 802.5 dùng cho mạng dạng vòng và trên cơ sở dùng tín hiệu thăm dò token. Mỗi trạm khi nhận được tín hiệu thăm dò token thì tiếp nhận token và bắt đầu quá trình truyền thông tin dưới dạng các frame. Các frame có cấu trúc tương tự như của chuẩn 802.4. Phương pháp xâm nhập mạng này quy định nhiều mức ưu tiên khác nhau cho toàn mạng và cho mỗi trạm, việc quy định này vừa cho người thiết kế vừa do người sử dụng tự quy định.

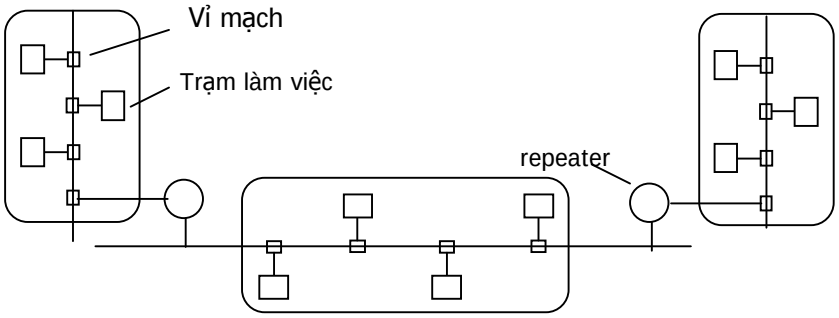
Chuẩn IEEE 802.11 dùng cho mạng không dây (Wireless).

4.3.1 Chuẩn Ethernet

Chuẩn Ethernet được sử dụng phổ biến nhất, đến mức đôi khi được hiểu đồng nghĩa với LAN. Tuy nhiên nó đã được xây dựng và phát triển qua các giai đoạn với các tên gọi là DIX standard Ethernet và IEE 802.3 standard. Chuẩn Ethernet do các công ty Xerox, Intel và Digital equipment xây dựng và phát triển. Ethernet LAN được xây dựng theo chuẩn 7 lớp trong cấu trúc mạng của ISO, mạng truyền số liệu Ethernet cho phép đưa vào mạng các loại máy tính khác nhau kể cả máy tính mini. Ethernet có các đặc tính kỹ thuật chủ yếu sau đây:

- Có cấu trúc dạng tuyến phân đoạn, đường truyền dùng cáp đồng trục, tín hiệu truyền trên mạng được mã hoá theo kiểu đồng bộ (Manchester), tốc độ truyền dữ liệu là 10 Mb/s.
- Chiều dài tối đa của một đoạn cáp tuyến là 500m, các đoạn tuyến này có thể được kết nối lại bằng cách dùng các bộ chuyển tiếp và khoảng cách lớn nhất cho phép giữa 2 nút là 2,8 km.
- Sử dụng tín hiệu băng tần cơ bản, truy xuất tuyến (bus access) hoặc tuyến token (token bus), giao thức là CSMA/CD, dữ liệu chuyển đi trong các gói. Gói tin dùng trong mạng có độ dài từ 64 đến 1518 byte.

- Cấu trúc của mạng Ethernet : Mạng Ethernet có cấu trúc dạng bus như sau :



Hình 4-3. Cấu trúc của mạng Ethernet.

Số trạm cực đại trong mạng là 1024, số lượng segment của mạng giới hạn nhỏ hơn 5 segment, khoảng cách tối đa giữa hai trạm là 2,5km. Mạng sử dụng cáp đồng trục tốc độ 10Mps. Cấu trúc khung tin Ethernet có khuôn dạng như sau :

Cờ	Địa chỉ đích	Địa chỉ nguồn	Loại tin	TIN	CRC	Cờ
	6 bytes	6 bytes	2 bytes	46 – 1500 bytes	4 bytes	

Chương 5

TẦNG MẠNG

Tầng mạng đảm bảo truyền tin thông suốt giữa hai nút đầu cuối trong mạng. Trên cơ sở cấu hình của mạng, tầng mạng sẽ kiểm tra sơ đồ kết nối (*topology*) của toàn mạng để quyết định đường đi tối ưu truyền gói dữ liệu, tránh quá tải trên một đường truyền trong khi một số đường truyền rỗi. Thực hiện cắt/ hợp dữ liệu khi qua mạng và liên kết mạng khi có nhiều mạng nối với nhau.

5.1 Các vấn đề của tầng mạng

5.1.1 Định địa chỉ cho tầng mạng

Tầng mạng sử dụng các kiểu địa chỉ bổ sung sau :

1. Địa chỉ mạng logic (Logical network addresses), định tuyến các gói tin theo các mạng cụ thể trên liên mạng. Dùng để định danh một mạng cụ thể trên liên mạng dưới dạng một nguồn hay đích của một gói tin.
2. Địa chỉ dịch vụ (Service addresses), định tuyến các gói tin theo các tiến trình cụ thể đang chạy trên thiết bị đích, dùng định danh một giao thức hay tiến trình trên máy tính là nguồn hay đích của một gói tin.
3. Địa chỉ mạng vật lý (MAC) định danh một thiết bị cụ thể dưới dạng một nguồn hay đích của một khung.

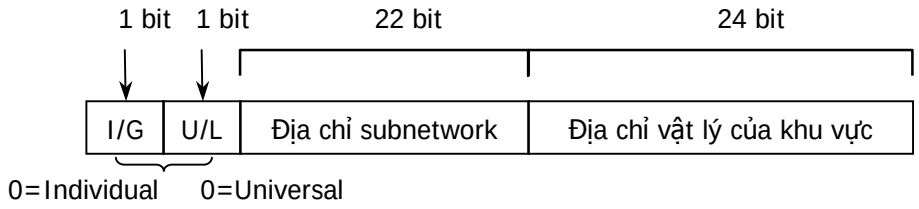
Địa chỉ vật lý của máy trạm :

Mỗi thiết bị trên một mạng có một địa chỉ vật lý duy nhất để giao tiếp với các thiết bị khác, còn gọi là địa chỉ phần cứng. Trên tất cả các mạng hiện nay, mỗi địa chỉ xuất hiện một lần duy nhất (nghĩa là mỗi thiết bị chỉ có một địa chỉ duy nhất). Đối với phần cứng, địa chỉ thường được mã hoá trong thiết bị card mạng (Network Interface Card), có thể được đặt bằng chuyển mạch hoặc bằng phần mềm. Trong mô hình OSI thì địa chỉ này được đặt ở lớp vật lý.

Độ dài của địa chỉ vật lý phụ thuộc vào từng mạng, chẳng hạn với mạng Ethernet và một số mạng khác thì dùng địa chỉ vật lý dài 48 bit. Để trao đổi thông tin thì cần có địa chỉ của nơi gởi, và địa chỉ của nơi nhận.

Hiện nay IEEE đang đảm nhiệm việc ấn định địa chỉ vật lý tổng thể (*universal physical address*) cho các subnetwork. Đối với mỗi subnetwork, IEEE ấn định một phần địa chỉ đồng nhất đối với tất cả các subnetwork gọi là OUI (Organization Unique Identifier) phần này có độ dài là 24 bit, cho phép IEEE ấn định phần địa chỉ 24 bit còn lại theo yêu cầu. (Trên thực tế, hai trong 24 bit địa chỉ OUI là các bit điều khiển, do đó 22 bit là để xác định subnetwork đó. Dó đó chỉ có khoảng 2^{22} địa chỉ

được dùng, nếu với tốc độ phát triển như hiện nay có thể sẽ thiếu địa chỉ trong tương lai). Sau đây là cấu tạo của địa chỉ OUI :



Hình 5-1. Cấu tạo của địa chỉ vật lý ĂUI.

5.1.2 Dịch vụ cung cấp cho tầng giao vận

- Các dịch vụ phải độc lập với công nghệ được dùng trong mạng.
- Tầng giao vận phải độc lập với công nghệ được dùng trong mạng.
- Các địa chỉ mạng phải thống nhất để tầng giao vận có thể dùng cả mạng LAN và WAN.

Có 2 loại dịch vụ :

- Dịch vụ truyền tin có liên kết (*Connection Ăriennted Service*)
- Dịch vụ truyền tin không liên kết (*Connectionless Service*)

Sự khác nhau giữa hai dịch vụ

Vấn đề	**Dịch vụ có liên kết**	**Dịch vụ không liên kết**
Khởi động kênh	Cần thiết	Không
Địa chỉ đích	Chỉ cần lúc khởi động	Cần ở mọi gói tin
Thự tự gói tin	Được đảm bảo	Không đảm bảo
Kiểm soát lỗi	ở tầng mạng	ở tầng giao vận
Điều khiển thông lượng	ở tầng mạng	ở tầng giao vận
Thảo thuận tham số	Có	Không
Nhận dạng liên kết	Có	Không

Các hàm cơ bản của dịch vụ liên kết tầng mạng :

N-CONNECT. Request (callce, caller, acks wanted, exp wanted, qos, user data)
N-CONNECT. Indication (callce, caller, acks wanted, exp wanted, qos, user data)
N-CONNECT. Response (response acks wanted, exp wanted, qos, user data)
N-CONNECT. Confirmation (response acks wanted, exp wanted, qos, user data)
N-DISCONNNECT. Request (originator, reason, user data, responding address)
N-DISCONNNECT. Indication (originator, reason, user data, responding address)
N-DATA. Request (user data)
N-DATA. Indication (user data)
N-DATA-ACKNOWLEDGED. Request ()
N-DATA-ACKNOWLEDGED. Indication ()
N-EXPEDITED-DATA. Request (user data)
N-EXPEDITED-DATA. Indication (user data)
N-RESET. Request (originator, reason)

N-RESET. Indication (originator, reason)
N-RESET. Response()
N-RESET. Confim()

Các hàm cơ bản của dịch vụ không liên kết tầng mạng

N-UNITDATA. Request (source address, destination address, qos, user_data)
N-UNITDATA. Indication (source address, destination address, qos, user_data)
N-FACILITY. Request (qos)
N-FACILITY. Indication (destination address, qos, reason)
N-FACILITY. Indication (destination address, qos, reason)

Hàm N_FACILITY.request cho phép NSD dịch vụ mạng biết tỷ lệ phần trăm gói tin đang được giao vận.

Hàm N_REPORT.indication cho phép tầng mạng thông báo lại cho NSD dịch vụ mạng.

5.1.3 Tổ chức các kênh truyền tin trong tầng mạng

Có hai loại kênh truyền tin hoạt động trong mạng :

5.1.3.1 Kênh ảo (virtual circuit)

Tương đương kênh điện thoại trong tầng vật lý sử dụng trong mạng có liên kết. Kênh ảo được thiết lập cho mỗi liên kết. Một khi đã được thiết lập thì các gói tin được chuyển đi tương tự trong mạng điện thoại cho đến khi liên kết bị hủy bỏ.

- Mỗi nút mạng chứa một kênh ảo, với cửa vào cho một kênh ảo
- Khi một liên kết được khởi động, một kênh ảo chưa dùng sẽ được chọn
- Nút chọn kênh ảo chứa đường dẫn đến trạm tiếp theo và có số thấp nhất

Khi gói tin khởi động đến nút đích, nút chọn kênh ảo có số thấp nhất thay thế số trong gói tin và chuyển vào trạm đích. Số kênh ảo nối với trạm đích có thể khác số kênh ảo mà trạm nguồn sử dụng.

5.1.3.2 Mạng Datagram

Tương đương với điện báo sử dụng trong mạng không liên kết. Trong mạng này, không có tuyến đường nào được thiết lập. Các gói tin có thể đi theo nhiều đường khác nhau mà không nhất thiết theo một trình tự xác định. Thông tin vào là địa chỉ đích, thông tin ra là nút mạng phải tới.

Mạng Datagram phức tạp về điều khiển nhưng nếu kênh hỏng thì dễ dàng đi theo kênh khác. Do đó có thể giải quyết được vấn đề tắc nghẽn dữ liệu.

- Các đặc trưng của mạng Datagram và mạng kênh ảo

Vấn đề	Mạng datagram	Mạng kênh ảo
Khởi động kênh	Không	Cần thiết
Địa chỉ (đ/c) hoá	Gói tin phải có đ/c nguồn và đ/c đích	Gói tin chỉ cần số của kênh ảo
Thông tin tìm đường	Không cần bất cứ thông tin nào.	Mỗi kênh ảo cần một vùng trong bảng
Tìm đường	Mỗi gói tin tìm đường độc lập. Phải tìm đường mỗi khi có gói tin tới nút mạng.	Được thiết lập khi khởi động kênh ảo mới. Liên kết sẽ được duy trì cho cả phiên.
Điều khiển	Chỉ mất gói tin ở trong nút hỏng	Kênh ảo đi qua nút hỏng sẽ bị huỷ
Hỏng nút	Khó khắc phục	Dễ khắc phục hơn
Độ phức tạp	Trong tầng giao vận	Trong tầng mạng
Thích hợp	Các dịch vụ liên kết và không liên kết	Các dịch vụ liên kết

5.1.4 Tìm đường đi trong mạng

Chức năng quan trọng nhất của tầng mạng là dẫn đường cho các gói tin từ trạm nguồn tới trạm đích. Thuật toán tìm đường là qui trình để quyết định chọn đường ra khỏi nút mạng nhằm gởi gói tin đi tiếp tới nút khác.

- Yêu cầu của thuật toán tìm đường

- Chính xác, ổn định, đơn giản và tối ưu.
- Thuật toán tìm đường phải có khả năng cập nhật lại cấu hình và đường vận chuyển để không phải khởi động lại mạng khi có một nút hỏng hoặc phải ngừng hoạt động của các máy ở trạm.

- Các thuật toán chia làm hai nhóm chính:

- Nhóm không thích nghi (*non adaptive*) : việc chọn đường không dựa vào việc đánh giá tình trạng giao thông và cấu hình trong thời gian thực.
- Nhóm thích nghi (*adaptive*) : việc tìm đường phải thích nghi với tình trạng giao thông hiện tại.

Sơ đồ mạng được biểu diễn dưới dạng đồ thị, mỗi nút của đồ thị là một nút mạng, cung của đồ thị biểu diễn đường truyền nối giữa hai nút. Việc chọn đường giữa hai nút mạng là tìm đường ngắn nhất giữa chúng.

Mỗi cung được gán một nhãn cho biết thời gian trung bình phải đợi và thời gian truyền một gói tin chuẩn. Thời gian này được thử mỗi giờ hay mỗi ngày một lần. Đường ngắn nhất là đường có ít bước chuyển tiếp qua nút nhất và có số đo độ dài nhỏ nhất, mất ít thời gian.

Có nhiều thuật toán để tìm đường ngắn nhất giữa 2 điểm, ví dụ như thuật toán Dijkstra (1959). Ta xây dựng đồ thị cho các nút mạng và tìm khoảng cách giữa các nút mạng.

5.1.5 Tắc nghẽn trong mạng

Khi có quá nhiều gói tin trong mạng hay một phần của mạng làm cho hiệu suất của mạng giảm đi vì các nút mạng không còn đủ khả năng lưu trữ, xử lý, gửi đi và chúng bắt đầu bị mất các gói tin. Hiện tượng này được gọi là sự tắc nghẽn (*congestion*) trong mạng.

Hàng đợi sẽ bị đầy (phải lưu tập tin, tạo các bảng chọn đường ...) nếu khả năng xử lý của nút yếu hoặc khi thông tin vào nhiều hơn khả năng của đường ra

Điều khiển dòng dữ liệu là xử lý giao thông giữa điểm với điểm, giữa trạm thu và phát. Trong khi đó điều khiển tránh tắc nghẽn là một vấn đề tổng quát hơn bao gồm việc tạo ra hoạt động hợp lý của các máy tính của các nút mạng, quá trình lưu trữ bên trong nút, điều khiển tất cả các yếu tố làm giảm khả năng vận chuyển của toàn mạng.

- Các biện pháp ngăn ngừa

- Bố trí khả năng vận chuyển, lưu trữ, xử lý của mạng dư so với yêu cầu.
- Huỷ bỏ các gói tin bị tắc nghẽn quá thời hạn.
- Hạn chế số gói tin vào mạng nhờ cơ chế cửa sổ (*flow control*).
- Chặn đường vào khi của các gói tin khi mạng quá tải.

5.2 Kết nối liên mạng

Nhu cầu trao đổi thông tin và phân chia các tài nguyên dùng chung đòi hỏi hoạt động truyền thông không chỉ ở phạm vi cục bộ mà ở cả khuôn khổ quốc gia và quốc tế. Từ đó dẫn đến sự nối kết các mạng viễn thông tin học được đặt ở các vị trí địa lý khác nhau và chịu sự quản lý của các tổ chức hoặc quốc gia khác nhau.

Sự nối kết mạng (*Networks Interconnection*) giống như ghép nối mạng đơn lẻ nhưng phức tạp hơn nhiều do tính chất không thuần nhất của các mạng con được kết nối. Chúng có thể có kiến trúc khác nhau bao gồm các máy tính nút mạng. Đường truyền khác nhau, chiến lược quản lý khác nhau.

Người ta thường xem xét các vấn đề sau để kết nối các mạng con lại với nhau :

- Xem mỗi nút của mạng con như là một hệ thống mở : mỗi nút mạng con có thể truyền thông trực tiếp với một nút của mạng con khác bất kỳ. Như thế yêu cầu phải xây dựng một chuẩn chung cho các mạng.
- Xem mỗi mạng như là một hệ thống mở : Hai nút thuộc hai mạng con không bắt tay trực tiếp với nhau mà phải thông qua một phần tử trung gian gọi là *giao diện kết nối (interconnection interface)* dat giữa hai mạng con đó.

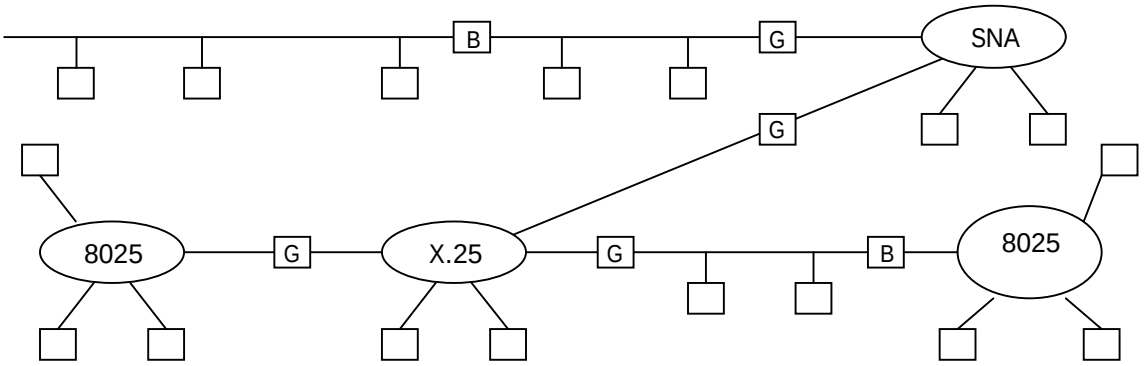
Chức năng của giao diện kết nối phụ thuộc vào sự khác biệt kiến trúc của mạng con : sự khác biệt càng lớn thì chức năng của giao diện càng phức tạp.

Có thể có các kết nối mạng như sau :

LAN-LAN	: Nối các mạng cục bộ.
LAN-WAN	: Nối các mạng cục bộ với mạng đường dài.
WAN-WAN	: Nối các mạng đường dài
LAN-WAN- LAN	: Nối mạng đường dài với mạng cục bộ.

Nếu máy nguồn và máy đích không ở cùng một mạng phải tìm đường từ mạng này sang mạng khác. Nếu trạm nguồn và đích không ở hai mạng liền kề thì giải quyết tìm đường qua nhiều trạm.

Các mạng khác nhau có các giao thức khác nhau, dẫn đến khác nhau về dạng khuôn của gói tin, đầu gói tin, điều khiển dòng dữ liệu và qui tắc xác nhận.



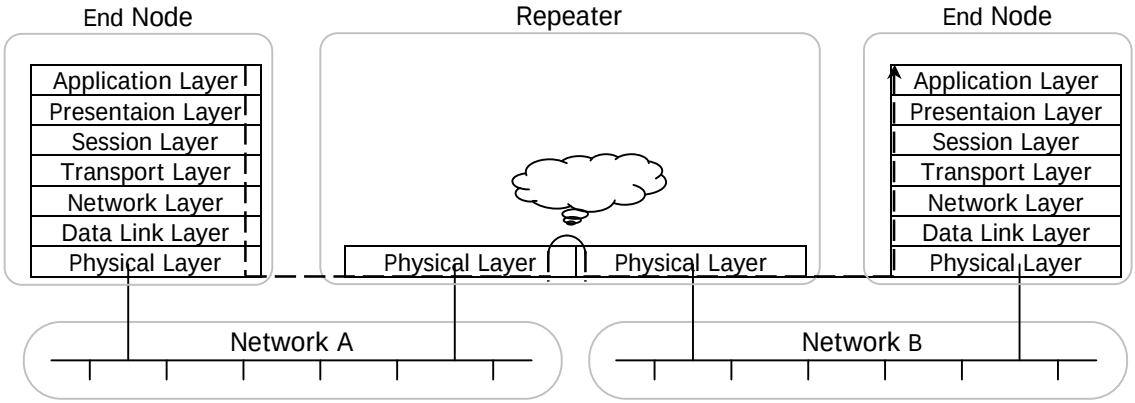
Hình 5-2. Kết nối liên mạng.

5.2.1 Các thiết bị dùng để kết nối liên mạng

Việc kết nối các LAN riêng lẻ thành một liên mạng chung gọi là Internetworking, sử dụng các thiết bị kết nối thông dụng như sau :

5.2.1.1 Bộ lặp

Bộ lặp (repeater) thực hiện chức năng ở tầng vật lý để khuyếch đại tín hiệu khi tín hiệu truyền đi xa. Bộ lặp được sử dụng để kết nối các đoạn mạng lại với nhau. Bộ lặp nhận tín hiệu từ một đoạn mạng, tái tạo và truyền tín hiệu này đến đoạn mạng khác. Nhờ có bộ lặp mà tín hiệu bị suy yếu do phải truyền qua một đoạn cáp dài có thể trở lại dạng ban đầu và truyền đi được xa hơn.



Hình 5-3. Sơ đồ kiến trúc của Repeater trong mô hình OSI.

Bộ lọc không có khả năng xử lý lưu lượng. Tất các tín hiệu điện, bao gồm cả nhiễu điện từ và các lỗi khác cũng được lặp và khuếch đại. Để bộ lặp hoạt động, cả hai đoạn mạng nối tới bộ lặp phải sử dụng cùng một phương thức truy nhập đường truyền. Ví dụ: bộ lặp không thể nối một đoạn mạng sử dụng phương thức CSMA/CD và một đoạn mạng sử dụng phương thức chuyển thẻ bài.

Bộ lặp có thể di chuyển gói dữ liệu từ phương tiện truyền dẫn này sang phương tiện truyền dẫn khác. Ví dụ có thể nhận gói dữ liệu từ một đoạn mạng dùng cáp đồng trục và chuyển gói đó sang đoạn mạng sử dụng cáp quang.

5.2.1.2 Hub

HUB là một thiết bị liên kết mạng được sử dụng rộng rãi. HUB còn là thành phần trung tâm trong cấu trúc mạng hình sao (Star). Mạng Star sử dụng sự phân chia tín hiệu trong HUB để đưa các tín hiệu ra các đường cáp khác nhau. Do vậy, có 3 loại HUB có thể sử dụng trong mạng là: HUB chủ động, HUB thụ động và HUB lai.

- ***HUB chủ động:*** Hầu hết các HUB đều là HUB chủ động, chúng tái tạo và truyền lại tín hiệu giống như bộ lặp. HUB thường có nhiều cổng nên thỉnh thoảng chúng còn được gọi là bộ lặp đa cổng. HUB chủ động đưa ra các tín hiệu mạnh hơn do đó cho phép đoạn cáp dài hơn.

Hình 5-4. Thiết bị kết nối mạng HUB.

- *HUB thụ động:* Các HUB thụ động hoạt động như các điểm kết nối, chúng không tái tạo hoặc khuếch đại tín hiệu.
- *HUB lai:* Các HUB thích ứng với nhiều loại cáp khác nhau được gọi là HUB lai.

5.2.1.3 Cầu nối (Bridge)

Cầu nối là một thiết bị hoạt động ở tầng liên kết dữ liệu. Dùng để nối hai hoặc nhiều đoạn (*segment*) của mạng LAN khác nhau.

Hình 5-5. Cầu nối.

- Chức năng của cầu nối :

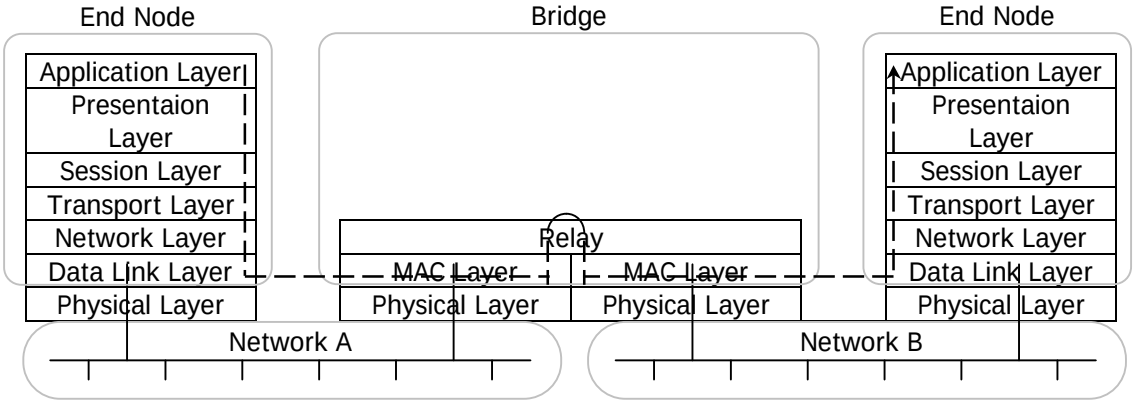
- Mở rộng khoảng cách của phân đoạn mạng, tăng số lượng máy tính trên mạng.
- Lọc những gói dữ liệu để gửi đi (hay không gửi) cho đoạn nối, hoặc gửi trả lại nơi xuất phát.
- Phân chia một mạng lớn thành hai mạng nhỏ nhằm cô lập lưu lượng, tăng tốc độ mạng. Nếu lưu lương từ một nhóm máy tính trở nên quá tải và làm giảm hiệu suất toàn mạng thì cầu nối có thể cô lập máy tính hoặc bộ phận này.
- Làm giảm hiện tượng tắc nghẽn do số lượng máy tính nối vào mạng qua lớn : Cầu nối có thể tiếp nhận một mạng quá tải và chia nó thành hai mạng riêng biệt, nhằm giảm bớt lưu lượng truyền trên mỗi đoạn mạng và do đó mỗi mạng sẽ hoạt động hiệu quả hơn.
- Kết nối các phương tiện truyền dẫn khác nhau, như cáp xoắn đôi và cáp quang.
- Kết nối các đoạn mạng sử dụng phương thức truy nhập đường truyền khác nhau, chẳng hạn CSMA/CD và chuyển thẻ bài.

- Nguyên lý hoạt dộng

- Cầu nối không phân biệt giữa giao thức này với giao thức khác, chỉ có nhiệm vụ chuyển lưu lượng của tất cả các giao thức dọc theo mạng. Vì giao thức nào cũng di chuyển ngang qua cầu nối, nên tuỳ thuộc vào từng máy tính quyết định chúng có thể nhận diện được giao thức nào.

- Cầu nối hoạt động trên nguyên tắc mỗi nút mạng có một địa chỉ riêng. Cầu nối chuyển gói dữ liệu dự trên địa chỉ của nút đích (địa chỉ MAC). Khi dữ liệu truyền qua cầu nối, thông tin địa chỉ của máy tính được lưu trong RAM của cầu nối dùng để xây dựng bảng địa chỉ dựa trên địa chỉ nguồn của gói tin.

Giao diện Bridge chỉ chứa tầng 1 và tầng con MAC, có chức năng chuyển đổi khuôn dạng của các đơn vị dữ liệu (frame) của các giao thức khác nhau và gửi chúng tới các mạng cục bộ đích có kèm theo phối hợp tốc độ.



Hình 5-6. Sơ đồ kiến trúc của Bridge trong mô hình OSI.

Ví dụ một Bridge nối giữa IEEE 820.3 và IEEE 820.5. Cầu nối này có hai card mạng: card Token Ring và card Ethermet để giao tiếp với hai mạng.

5.2.1.4 Bộ dẫn đường (router)

Trong môi trường gồm nhiều đoạn mạng với giao thức và kiến trúc mạng khác nhau, cầu nối không thể đảm bảo truyền thông nhanh trong tất cả các đoạn mạng. Mạng có độ phức tạp như vậy cần một thiết bị không những biết địa chỉ của mỗi đoạn mạng, mà còn quyết định tuyến đường tốt nhất để truyền dữ liệu và lọc lưu lượng quảng bá trên các đoạn mạng cục bộ. Thiết bị như vậy được gọi là bộ định tuyến.

Hình 5-7. Bộ định tuyến.

- Chức năng của bộ định tuyến :

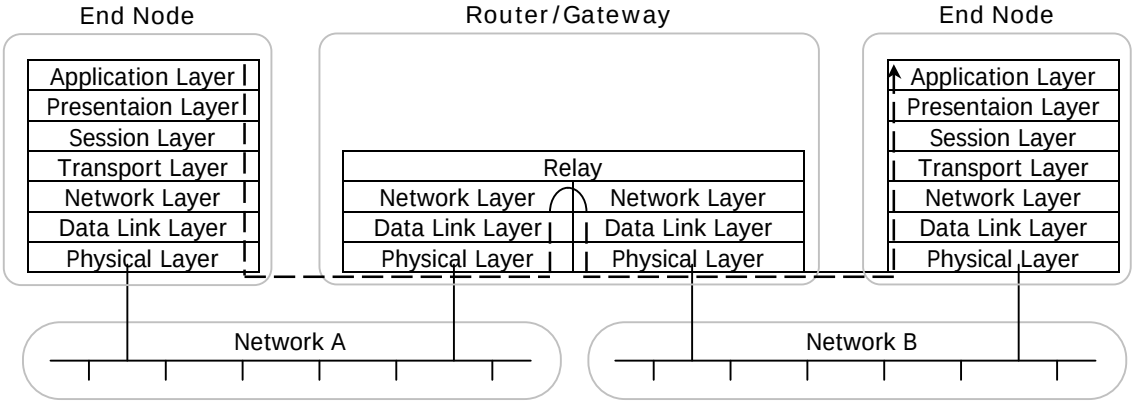
- Chuyển đổi và định tuyến gói dữ liệu qua nhiều mạng dựa trên địa chỉ phân lớp của mạng, cung cấp các dịch vụ như bảo mật, quản lý lưu thông...
- Phân chia một mạng lớn thành nhiều mạng nhỏ, và có thể kết nối nhiều đoạn mạng với nhau.
- Lọc gói tin và cô lập lưu lượng mạng : hoạt động như một rào cản an toàn giữa các đoạn mạng (do có thể lọc dữ liệu).
- Ngăn chặn tình trạng quảng bá vì chúng không chuyển tiếp các gói tin quảng bá, cải thiện việc phân phát gói dữ liệu.
- Các bộ định tuyến có thể chia sẽ thông tin trạng thái và thông tin định tuyến với nhau và sử dụng thông tin này để bỏ qua các kết nối hỏng hoặc chậm.

- Nguyên lý hoạt động :

Trong bộ định tuyến có một bảng định tuyến chứa các địa chỉ mạng. Tuy nhiên, địa chỉ mạng có thể được lưu trữ tuỳ thuộc vào giao thức mạng đang chạy. Bộ định tuyến sử dụng bảng định tuyến để xác định địa chỉ đích cho dữ liệu nhận được. Bảng này liệt kê các thông tin sau:

- Địa chỉ mạng đã kết nối.
- Cách kết nối tới các mạng khác.
- Phí tổn truyền dữ liệu qua các lộ trình đó.

Khi bộ định tuyến nhận được một gói dữ liệu cần gởi đến mạng ở xa, nó kiểm tra bảng đinh tuyến và chọn đường đi tối ưu (theo một tiêu chuẩn nào đó) để gởi gói dữ liệu đến đích.



Hình 5-8. Sơ đồ kiến trúc của Router trong mô hình OSI.

- Truyền dữ liệu qua bộ định tuyến

Trong mọi trường hợp, khi một trạm xác định rằng nó phải gởi một gói dữ liệu tới một trạm trên một mạng khác. Công việc đầu tiên trạm này cần làm là lấy địa chỉ vật lý MAC của Router (địa chỉ cổng nối ngầm định). Sau đó nó điền thông tin trong trường địa chỉ vật lý đích của gói dữ liệu bằng địa chỉ vật lý MAC của Router, và trường thông tin địa chỉ đích ở tầng mạng (chẳng hạn địa chỉ IP nếu dùng giao thức TCP/IP) bằng địa chỉ của trạm đích.

Khi Router kiểm tra địa chỉ đích, nó xác định xem nó biết hay không biết cách chuyển tiếp gói dữ liệu đến bước nhảy tiếp theo (Router kế tiếp trên đường đi) bằng cách kiểm tra địa chỉ. Nếu địa chỉ mạng đích nằm trong gói dữ liệu không có bảng định tuyến, Router thường bỏ gói dữ liệu đi. Trong trường hợp địa chỉ mạng đích có bảng định tuyến, Router thay địa chỉ vật lý đích bằng địa chỉ vật lý của bước nhảy tiếp theo và truyền gói dữ liệu đến bước nhảy tiếp theo.

Như vậy, khi một gói tin được chuyển qua liên mạng, địa chỉ vật lý đích của nó thay đổi, nhưng địa chỉ của giao thức không đổi.

Bộ định tuyến được chia thành 2 loại, tuỳ theo cách sử dụng chúng. Bộ định tuyến cục bộ (Local Router) nối các đoạn mạng ở gần nhau. Hai bộ định tuyến ở xa nhau (Remote Router) nối hai đoạn mạng ở xa qua các kênh truyền thông.

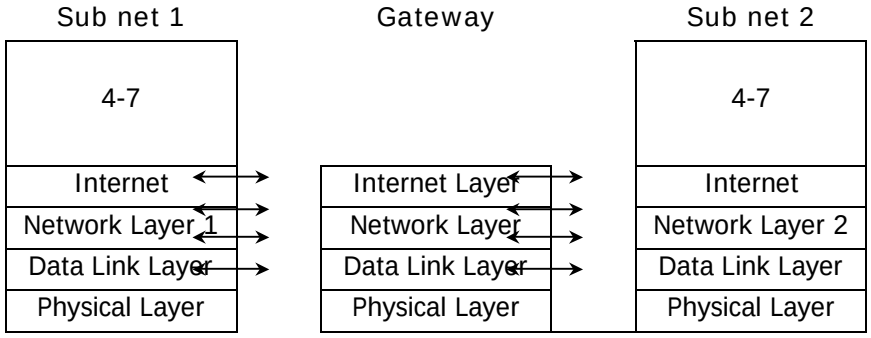
5.2.1.5 Bộ chuyển mạch

Chức năng chính của bô chuyển mạch (switch) là cùng một lúc duy trì nhiều cầu nối giữa các thiết bị mạng bằng cách dựa vào một loại đường truyền xương sống (backbone) nội tại tốc độ cao. Switch có nhiều cổng, mỗi cổng có thể hỗ trợ toàn bộ Ọthernet LAN hoặc Token Ring. Bộ chuyển mạch kết nối một số LAN riêng biệt và cung cấp khả năng lọc gói dữ liệu giữa chúng.

Các bộ chuyển mạch là loại thiết bị mạng mới, hiện đang được sử dụng rộng rãi vì Switch cho phép chuyển sang chế độ truyền không đồng bộ ATM.

5.2.1.6 Gateway

Hoạt động ở mức mạng, thực hiện ghép nối với WAN. Nguyên lý chung của nối kết này là tạo ra 1 tầng "*liên mạng*" (internet) chung trong tất cả các kiến trúc của mạng con tham gia nối kết. Tầng liên mạng thường là tầng con nằm ngay trên tầng 3 mô hình OSI.



Hình 5-9. Sơ đồ kiến trúc của gateway trong mô hình OSI.

Tầng con Internet được cài đặt trong tất cả các trạm cũng như trong các giao diện kết nối (gateway), Tầng này cung cấp dịch vụ truyền thông liên mạng với hai chức năng chính :

- Chuyển đổi các đơn vị dữ liệu của giao thức (Protocol Data Unit - PDU)
- Chọn đường đi cho các PDU này.

Các gói tin ở tầng con Internet lưu thông trong mạng theo phương pháp '*gói/bóc*' (encapsulation/decasulation). Khi một datagram được truyền từ mạng con này sang mạng con khác thông qua gateway thì nó được bổ sung thêm vào (hoặc tách ra) các phần thông tin điều khiển cần thiết tương ứng với các mạng con.

5.3 Giao thức liên mạng IP

Giao thức IP (Internet Protocol) hoạt động ở tầng mạng, cung cấp dịch vụ dữ liệu không liên kết (connectionless) cho nhiều giao thức liên kết dữ liệu khác. Đơn vị dữ liệu dùng trong giao thức IP được gọi là *datagram*, hay còn gọi là khung tin IP.

- Chức năng của giao thức IP :

- Định nghĩa gói tin Datagram là đơn vị dữ liệu cơ bản của việc truyền tin trên mạng Internet.
- Xác định mô hình đánh địa chỉ cho các khung tin và quản lý các quá trình trao đổi, xử lý các khung tin này.
- Chọn đường cho các datagram trên mạng
- Cung cấp cơ chế trên gói tin trên mạng hiệu quả nhất.
- Phân đoạn và tổng hợp các gói tin.

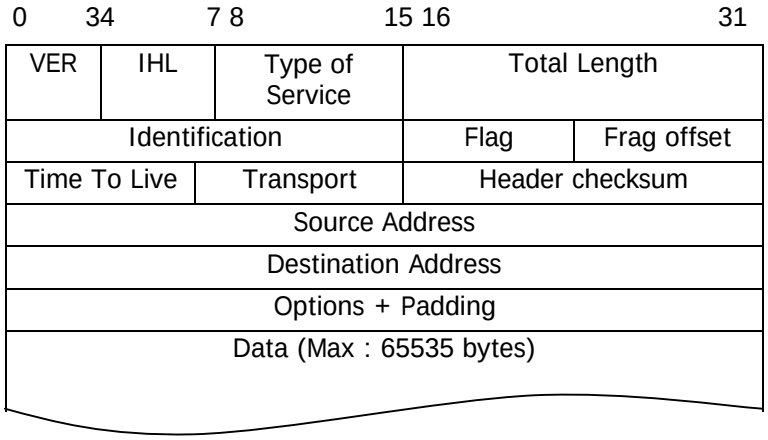
- Tính chất của giao thức IP :

- Hoạt động theo phương thức không kết nối : IP không chuyển các thông tin điều khiển trước khi truyền dữ liệu.

- Không tin cậy : giao thức IP không có khả năng phát hiện và khắc phục lỗi., không quan tâm đến vấn đề dữ liệu có được nhận một cách chính xác hay không. Do đó, các gói dữ liệu có thể bị thất lạc, bị trùng lặp, bị chuyển chậm hoặc đi không đúng thứ tự, mỗi gói dữ liệu được xử lý độc lập với nhau và có thể gửi theo những đường định tuyến khác nhau.

5.3.1 Cấu trúc khung tin IP

IP Header được gắn cho mỗi datagram, chứa các thông tin cần thiết cho sự hoạt động của gói tin trên mạng. Cấu trúc khung tin IP như hình sau :



Hình 5-10. Cấu trúc khung tin IP.

VER (4 bit)

Chứa phiên bản giao thức IP đang dùng. Phiên bản hiện nay là IPV4.

Một phần của giao thức IP quy định rằng phần mềm nhận dữ liệu trước tiên phải kiểm tra phiên bản của IP trong các khung tin đến, trước khi phân tích tiếp phần còn lại của Header và dữ liệu. Nếu như không đúng phiên bản thì lớp IP của máy nhận sẽ từ chối và bỏ qua toàn bộ nội dung của khung tin đến.

IHL (Internet Header length) (4 bit)

Chứa chiều dài của Header IP do máy gửi dữ liệu tạo nên, chiều dài này được tính theo các word có chiều dài 32 bit. Header ngắn nhất có chiều dài là 5 word (20 byte), nhưng do việc dùng các trường lựa chọn có thể làm tăng chiều dài của Header lên đến 6 word (24 byte). IHL dùng để giao thức IP được vị trí kết thúc của Header và bắt đầu phần dữ liệu của khung tin.

Type of Service - Loại dịch vụ (8 bits)

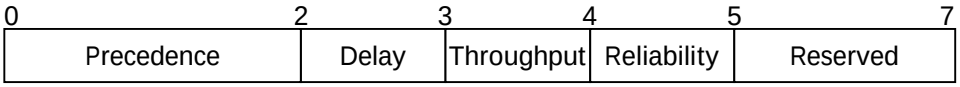
Trường này chứa các thông tin về quyền ưu tiên của việc truyền datagram và các ảnh hưởng có thể xảy ra trong quá trình truyền các datagram

đó. IP chuẩn không yêu chỉ ra các hành động cụ thể dựa trên giá trị của trường *Type of Service*. IP chỉ định sử dụng nó trong việc thiết lập các tuỳ chọn cho các mạng con và nó sẽ truyền qua trong bước nhảy tới.

Ví dụ, việc truy nhập vào mạng Token Ring cần thiết có các mức độ ưu tiên được xác định. IP có thể chuyển các mức độ ưu tiên của nó sang các mức độ ưu tiên tương ứng của mạng Token Ring.

Một số máy tính và bộ chọn đường (*router*) không quan tâm đến giá trị của trường này trong khi một số khác lại dựa vào đây để quyết định đường truyền.

Cấu trúc của trường như sau :



Cấu trúc của trường Type of Service

Precedence (3 bít) : chỉ thị về quyền ưu tiên gửi datagram, cụ thể là :

111 - Network Control (cao nhất) 011 - Flash
110 - Internetwork Control 10 - Immediate
101 - CRITIC/ECP 001 - Priority
100 - Flag Overrride 000 - Routime (thấp nhất)

D (Delay) - 1 bit : chỉ độ trễ yêu cầu
D = 0 độ trễ bình thường
D = 1 độ trễ thấp

T (Throughput) - 1 bit : chỉ thông lượng yêu cầu
T = 0 thông lượng bình thường
T = 1 thông lượng cao

R (Reliability) - 1 bít chỉ độ tin cậy yêu cầu
R = 0 độ tin cậy bình thường
R = 1 độ tin cậy cao

Ba bit đầu tiên của trường này là để chỉ ra quyền của khung tin đó, với các giá trị từ 0 (bình thường) đến 7 (Mạng điều khiển). Nếu giá trị của phần này càng cao thì khung tin đó càng quan trọng và trên lý thuyết thì khung tin này phải được chuyển đến đích nhanh hơn. Nhưng trên thực tế thì TCP/IP và các phần cứng dùng giao thức TCP/IP đều bỏ qua trường này và coi tất cả các khung tin có độ ưu tiên như nhau.

Ba bit tiếp theo là ba cờ 1-bit để điều khiển thời gian trễ, độ tin cậy, và thông lượng (throughput) của khung tin. Nếu tất cả các bit đều là 0 thì có nghĩa là đặt ở chế độ bình thường. Nếu bit thứ nhất là 1 thì có nghĩa là thời gian trễ thấp, truyền nhanh và độ tin cậy cao cho từng cờ. Còn hai bit còn lại của trường này không dùng. Phần lớn các bit của trường này đều bị bỏ qua khi thực hiện IP, và tất cả các khung tin đều được đặt thời gian trễ, thời gian truyền, và độ tin cậy như nhau.

Trong thực tế, hầu hết tất cả các bit của trường loại dịch vụ đều được đặt về giá trị 0 bởi vì sự khác nhau về quyền, thời gian trễ, thời gian truyền, độ tin cậy giữa các máy hầu như không tồn tại trừ khi một mạng mới được thành lập.

Total Length (16 bits) - Chiều dài gói tin

Trường này cho biết toàn bộ chiều dài của khung tin (datagram) bao gồm phần Header và phần dữ liệu, đơn vị tính bằng byte. Độ lớn của trường này là 16 bit do đó mà chiều dài của khung tin tối đa là 65535 byte.

Indentification (16 bits) -Trường định danh

Trường này chứa một giá trị đặc trưng do máy gửi khung tin tạo ra, cùng với các tham số khác (như Source Address và Destination Address), tham số này dùng để định danh duy nhất một khung tin trong khoảng thời gian nó tồn tại trên liên mạng.

Số trong trường này được cần đến khi sắp xếp các khung tin để đảm bảo rằng các khung tin không bị lẫn lộn với nhau. Khi lớp IP nhận được một đoạn dữ liệu từ các lớp cao hơn thì nó sẽ gán các số định danh này vào. Nếu như khung tin đã được tách (bằng kỹ thật tách thông tin) thì tất cả các khung tin sẽ mang cùng một số định danh như nhau.

Flags (3 bits) - Các cờ

Trường này có chiều dài 3 bit, liên quan đến sự phân đoạn các datagram.

Bit 0 : Dùng để dự trữ - chưa sử dụng, luôn có giá trị 0

Bit 1 : (DF) = 0 (May Fragment)

=1 (Don’t Fragment)

Bit 2 : (MF) = 0 (Last Fragment)

=1 (More Fragment)

Nếu như cờ DF có giá trị là 1 thì có nghĩa là khung tin không thể tách ra được trong bất cứ trường hợp nào. Nếu như mà phần mềm của lớp IP hiện tại không thể gửi khung tin đến nới nhận nếu như không tách ra, mà hiện tại bit cờ đang là 1 thì khi đó khung tin sẽ bị huỷ bỏ và một thông báo lỗi được gửi đến thiết bị phát.

Nếu router không thể truyền nguyên cả một datagram mà bit này được thiết lập bằng 1 thì datagram đó sẽ bị loại bỏ và nó sẽ có một thông báo lỗi gửi đến máy phát. Bất kì một người quản lý mạng nào cũng có thể sử dụng cách này để kiểm tra độ lớn của các datagram có thể được truyền trên các phần khác nhau trên mạng kết hợp.

Nếu như cờ MF là 1 có nghĩa là khung tin hiện tại vẫn đang còn các gói tin khác nữa đang đến, do đó mà phải cần đến việc sắp xếp lại để khôi phục lại message ban đầu. Khung tin cuối cùng đến sẽ lớn hơn các khung tin bình thường vì nó còn chứa thêm phần MF=0 để báo cho máy nhận biết là đã hết các khung tin cần thiết không cần phải đợi thêm nữa. Có thể là các khung tin đến không đúng với thứ tự chúng đã được phát đi, do đó cờ MF còn được dùng cùng với trường Fragment Off để chỉ cho máy nhận được thứ tự của toàn bộ message ban đầu.

Fragment Offset (13 bits)

Nếu mà cờ MF bằng 1 (tức là có sự tách thông tin từ một khung tin lớn), khi đó fragment offset chứa .vị trí của các message con trong message ban đầu trong khung tin hiện thời. Điều này cho phép IP sắp xếp lại các khung tin thành message ban đầu t theo đúng trật tự.

Offset thường được để ở đầu message. Trường này có chiều dài là 13 bit, do vậy offset được tính theo đơn vị 8 byte, tương ứng với gói lớn nhất là 65535 byte. Việc dùng số định danh để chỉ rằng khung tin đến là thuộc bản tin nào, lớp IP ở máy nhận có thể dùng fragment offset để sắp xếp lại message ban đầu.

TTL (Time to Live - Thời gian sống)

Trường này cho biết khoảng thời gian tính bằng giây mà một khung tin có thể tồn tại trên mạng trước khi nó bị huỷ bỏ. Giá trị này được nút gửi khung tin đi ấn định.

Các chuẩn của TCP/IP quy định rằng trường TTL phải được giảm đi ít nhất là 1 giây cho mỗi nút xử lý khung tin đó, thậm chí là thời gian xử lý có thể nhỏ hơn 1 giây. Khi một gateway nhận được một khung tin thì thời gian đến được đính vào khung tin do đó nếu như khung tin đó phải chờ để được xử lý. Bởi vậy nếu một gateway nào đó mà bị quá tải và không thể lấy khung tin về, khi đó bộ đếm thời gian của trường TTL sẽ tự động giảm đi trong quá trình chờ để được xử lý. Nếu trường TTL giảm về 0 thì khi đó khung tin đó phải được nút hiện thời hủy bỏ, sẽ có một thông báo gửi về máy gửi.

Hầu hết các TCP/IP cài đặt giá trị trường TTL khoảng 60 hoặc cao hơn, nghĩa là datagram có thể đi qau 60 router hay hop để đến đích. Trường TTL được thiết kế để tránh việc các gói dữ liệu cứ chuyển vòng quanh trên mạng mà không có đường ra.

Giao thức giao vận (Transport Protocol)

Trường này chứa số định danh của giao thức giao vận mà đã xử lý khung tin. Số định danh này do trung tâm thông tin mạng Internet NIC ấn định. Hiện nay đã có khoảng 50 giao thức giao vận được ấn định. Hai giao thức quan trọng nhất là : ICMP (Internet Control message Protocol) và TCP.

Header checksum

Dùng để tính checksum của trường Header để làm cho quá trình xử lý thông tin được nhanh hơn. Do trường TTL bị giảm đi 1 giây mỗi khi được xử lý, trường checksum cũng thay đổi tại các máy mà khung tin đi qua. Thuật toán checksum là một thuật toán nhanh và có hiệu quả, nhưng có một số trường hợp bị sai chẳng hạn mất hoàn toàn một từ 16 bit mà 16 bit này đều bằng 0. Tuy nhiên trường checksum do cả TCP và UDP để đóng gói, các lỗi này sẽ được phát hiện khi khung tin được tập hợp để truyền trên mạng.

Source Address (32 bits) : chứa địa chỉ IP 32 bit của máy gửi.

Destination Address (32 bits) : chứa địa chỉ IP 32 bit của máy nhận.

Hai trường trên được tạo ra cùng với khung tin và không bị thay đổi trong quá trình truyền.

Options (32 bits) - Phần lựa chọn

Phần lựa chọn được tạo ra từ một vài mã mà các mã này có độ dài có thể thay đổi được. Nếu như có nhiều lựa chọn trong khung tin, thì các lựa chọn đó được đặt liên tục nhau trong phần Header của IP. Tất cả các lựa chọn này được điều khiển bằng một byte có ba trường: ***Cờ copy*** có độ dài 1 bit, ***loại lựa chọn*** có độ dài 2 bit, và ***trường số lựa chọn*** có độ dài 5 bit. Trường cờ copy được dùng để quy định là lựa chọn sẽ được thực hiện như thế nào nếu ở một gateway nào đó cần đến kỹ thuật tách thông tin. Nếu như cờ này có giá trị là 0 thì có nghĩa là lựa chọn đó sẽ được copy vào khung tin thứ nhất mà không copy vào các khung tin tiếp theo sau. Nếu như cờ này có giá trị là 1 thì có nghĩa là lựa chọn đó sẽ được sao chép vào tất cả các khung tin.

Các lựa chọn quan trọng là Record route và Timestamp.

Record route

Trường Record Route (*Bản ghi chọn đường)* chứa danh sách dự trữ của các route mà datagram đã đi qua trên đường tìm tới đích. Mỗi lần đi qua một router thì trường này sẽ bổ sung một địa chỉ của router đó vào danh sách của nó. Độ dài của trường này do máy nguồn xác lập, do đó rất có thể là nó sẽ bị

đầy trước khi datagram tìm được đến đích. Trong trường hợp này thì các địa chỉ của các router sau sẽ không được thêm vào danh sách của nó.

Timestamp : Có 3 định danh cho trường Timestamp. Trường này có thể chứa:

- Danh sách của 32 bit Timestamp.
- Danh sách của địa chỉ IP và các cặp Timestamp tương ứng.

Danh sách của các địa chỉ cho trước bởi máy nguồn. Một nút bất kỳ được ghi vào trường này chỉ khi địa chỉ của nó là mục kế tiếp trong danh sách này. Trường này có thể bị đầy nếu rơi vào hai trường hợp đầu, trong trường hợp này sẽ có trường ghi tràn (overllow field) dùng để đếm số nút mà không thể ghi vào timestamp được.

Padding (Độ dài thay đổi)

Nội dung của phần **Padding** phụ thuộc vào phần **Options** như thế nào. Phần Padding thường được dùng để bảo đảm rằng chiều dài Header của khung tin luôn là một số nguyên bội số của 32.

Data : Vùng dữ liệu có độ dài thay đổi, nhưng luôn là bội số của 8 bits, và tối đa là 65535 bytes.

5.3.2 Địa chỉ IP

Mỗi thiết bị nối vào mạng TCP/IP được gán một địa chỉ IP duy nhất (mỗi card mạng sẽ có địa chỉ IP riêng). Khi sử dụng mạng cục bộ không kết nối với các mạng khác, người sử dụng có thể gán địa chỉ IP tuỳ ý cho các máy trạm. Tuy nhiên, đối với các site Internet thì địa chỉ IP phải được cung cấp từ trung tâm quản lý thông tin mạng trên thế giới (NIC - Network Information Center).

Địa chỉ của IP có độ dài 32 bit, được chia làm 4 phần, mỗi phần 1 byte, phân cách nhau bằng dấu chấm. Dạng tổng quát : ***x.y.z.t*** với $0 \leq x,y,z,t \leq 255$

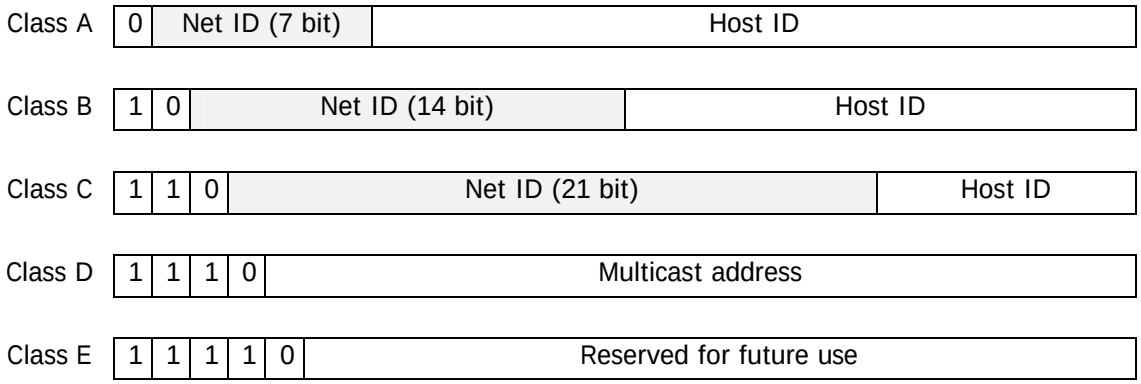
Ví dụ: 128.83.12.14 hoặc 0x80530C0E Hex.

Địa chỉ IP bao gồm hai phần thông tin: địa chỉ mạng (network address) và địa chỉ máy (host address): NetworkID.HostID

Khi đề nghị NIC cung cấp địa chỉ IP ta sẽ không nhận được địa chỉ tương ứng của máy trạm, thay vào đó là địa chỉ mạng và ta có quyền gán địa chỉ cho các máy trạm của mạng trong phạm vi địa chỉ được cung cấp.

5.3.2.1 Các lớp địa chỉ IP

Địa chỉ IP thuộc một trong E lớp địa chỉ, từ lớp A đến E. Các lớp địa chỉ nhằm để phân loại các mạng có quy mô khác nhau.



Hnh 5-11. Các lớp địa chỉ IP.

1. Lớp A (1≤ x ≤ 126) : NetworkID= x, HostID=y.z.t

- Cho phép định danh 126 mạng, với tối đa 2^{24} (= 167.772) máy trạm trên mỗi mạng, lớp A giới hạn số subnetwork trong Internet.
- Các mạng lớp A thuộc loại mạng diện rộng (very large), như mạng quốc gia

2. Lớp B (128≤ x ≤ 191) : NetworkID= x.y, HostID=z.t

- Cho phép định danh đến 16384 mạng, với tối đa 2^{16} (=65.536) host trên mỗi mạng.
- Mạng lớp B thuộc loại mạng trung bình như mạng University Campuses.

3. Lớp C (192 ≤ x ≤ 223) : NetworkID= x.y.z, HostID=t

- Giới hạn số trạm trong mạng lớn nhất là 256, có 21 bit cho địa chỉ mạng. Cho phép định danh đến 2 triệu mạng, với tối đa 254 host trên mỗi mạng.
- Mạng lớp C được sử dụng cho các loại LAN, như các mạng Enterprise-wide.

4. Lớp D (224≤ x ≤ 239)

- Địa chỉ lớp D dùng cho các giao thức đặc biệt (Internet Group management Protocol - IGMP) và các giao thức khác.

5. Lớp E (240≤ x ≤ 255) : Để dành cho sự phát triển về sau.

- Các máy trong cùng một mạng phải có địa chỉ mạng giống nhau.
- Các mạng khác nhau có địa chỉ mạng khác nhau.

5.3.2.2 Các địa chỉ IP đặc biệt

1. Địa chỉ quay vòng : 127.y.z.t

Tất cả các gói tin được gửi đến địa chỉ 127.0.0.0 sẽ được gửi ngược trở lại máy tính. Gói tin này được sao chép từ nơi truyền đến bộ đệm nơi nhận trên cùng một máy tính. Địa chỉ loopback có thể được sử dụng như một địa chỉ kiểm tra

nhanh xem phần mềm TCP/IP có được cấu hình thích hợp. Trên hệ điều hành Windows địa chỉ loopback là 127.0.0.1 còn Unix là 127.1.*.

2. Mặt nạ mạng (Netmask)

Mặt nạ mạng của một địa chỉ IP là một giá trị 32 bits trong đó các bit tương ứng với phần địa chỉ mạng bằng 1, các bit của phần máy bằng 0.

Ví dụ : Địa chỉ IP lớp B có mặt nạ mạng là 255.255.255.0 sẽ cho địa chỉ mạng con là 180.10.15.0

3. Địa chỉ quảng bá (broadcast address)

Địa chỉ này có các bit của phần HostID bằng 1, được sử dụng khi muốn chuyển một gói tin đến mọi máy tính trong mạng con.

Ví dụ một mạng con có địa chỉ là 180.10.0.0 sẽ có địa chỉ quảng bá là 180.10.255.255. Tương tự, một mạng con có địa chỉ là 180.10.15.0 sẽ có địa chỉ quảng bá là 180.10.15.255.

Đặc biệt địa chỉ 255.255.255.255 quảng bá cục bộ (local broadcast) hay còn gọi là limited broadcast có thể sử dụng trong các LAN.

Địa chỉ 0.0.0.0 cũng được sử dụng trong bảng định tuyến để chỉ đến điểm vào mạng cho địa chỉ bộ định tuyến mặc định.

5.4 Phân chia mạng con

Để thuận tiện cho việc quản lý và định hướng dữ liệu trên mạng lớn, người ta thường tổ chức mạng IP theo cơ chế địa chỉ phân cấp : mỗi mạng được chia nhỏ thành nhiều mạng con, mỗi mạng con thực hiện các dvc về địa chỉ trong nội bộ mạng đó. Sự phân cấp này cho phép giảm khối lượng công việc chọn đường cho các gói tin trong toàn liên mạng.

Mỗi mạng con chịu trách nhiệm cho việc chọn đường cho các gói tin IP trong mạng của mình, các gói tin này được nhận ra nhờ phần địa chỉ mạng của nó. Trong các mạng loại A, B, C thì phần địa chỉ này có độ dài cố định. Tuy nhiên, để tạo sự linh hoạt trong việc phân chia mạng con thì địa chỉ mạng có thể mở rộng sang các bit của địa chỉ máy. Đó là kỹ thuật phân chia mạng con.

Ví dụ một mạng loại B có địa chỉ mạng là 203.160.9.0 và mặt nạ mạng là 255.255.255.0 (địa chỉ mạng dài 24 bit). Người ta cần chia mạng này thành 4 mạng cục bộ riêng, do đó sẽ lấy thêm 2 bit cho địa chỉ mạng (26 bit). Vậy ta có địa chỉ các mạng con này là :

Địa chỉ mạng 1 :

203	160	9	0
11001011	10100000	00001001	00000000

Địa chỉ mạng 2 :

203	160	9	64
11001011	10100000	00001001	01000000

Địa chỉ mạng 3 :

203	160	9	128
11001011	10100000	00001001	10000000

Địa chỉ mạng 4 :

203	160	9	192
11001011	10100000	00001001	11000000

Mặt nạ của các mạng con này là : 255.255.255.192

255	255	255	192
11111111	11111111	11111111	11000000

Việc phân chia mạng được tiến hành bởi người quản trị hệ thống và thường dựa trên ranh giới vật lý giữa các nhánh mạng. Khi có gói dữ liệu cần chuyển đi, bộ định tuyến sẽ dùng mặt nạ mạng để kiểm tra gói dữ liệu này thuộc mạng con nội bộ hay thuộc mạng ngoài. Sự phân chia mạng riêng thành các mạng con chỉ có ý nghĩa bên trong mạng đó.

Nếu kết nối Internet thông qua một mạng LAN, điều quan trọng là phải sử dụng đúng mặt nạ mạng. Cũng giống như địa chỉ IP, một mặt nạ mạng con có thể được gán một cách riêng lẻ hay có thể tự động thông qua DHCP (Dynamic Host Configuration Protocol).

5.5 Hoạt động của giao thức IP

Nếu địa chỉ đích của gói tin IP không nằm trên cùng mạng với máy chủ nguồn thì giao thức IP trong máy chủ hướng gói tin đến bộ định tuyến nội bộ. Nếu bộ định tuyến này không được nối đến mạng đích, gói tin sẽ được gởi đến một bộ định tuyến khác. Cứ thế cho đến khi tới trạm đích. Việc quy định truyền theo đường truyền nào của router dựa trên bảng đường truyền (*routing table*). Các bộ định tuyến có thể phát hiện :

- Một mạng mới đã được thêm vào liên mạng
- Đường dẫn đến trạm đích đã bị hỏng

Các bước thực hiện bởi một thực thể IP như sau :

- *Đối với thực thể IP ở trạm nguồn*

- Khi nhận được lệnh SEND từ tầng trên, nó thực hiện các bước như sau:
- Tạo một IP datagram dựa trên các tham số của lệnh SEND
- Tính checksum và ghép vàp phần đầu của datagram
- Ra quyết định chọn đường
- Chuyển datagram xuống tầng dưới

• *Đối với gateway*

- Khi nhận được datagram quá cảnh, nó thực hiện các tác động như sau :
- Tính checksum, nếu không đúng thì loại bỏ datagram
- Giảm giá trị tham số thời gian tồn tại. Nếu hết thời gian thì loại bỏ datagram
- Ra quyết định chọn đường
- Phân loại datagram nếu cần
- Kiến tạo lại phần đầu IP bao gồm giá trị mới của vùng TTL, checksum, Fragmentation.
- Chuyển datagram xuống tầng dưới để truyền qua mạng.

• *Tại trạm đích*

- Tính checksum, nếu không đúng thì loại bỏ datagram.
- Tập hợp các đoạn của datagram.
- Chuyển dữ liệu và các tham số điều khiển lên tầng trên.

Như vậy, do gói tin IP không sửa đổi, đơn giản nên hiệu suất đường truyền cao. Vì gói tin IP cung cấp dịch vụ giao nhận gói tin không tin cậy nên cần có giao thức ICMP để hỗ trợ, các bản tin ICMP được đóng gói và chuyển tải trong các gói tin IP. Tầng TCP đảm nhận việc bảo đảm các datagram được truyền đến đích một cách an toàn và đầy đủ.

5.6 Các giao thức liên quan đến IP

5.6.1 Giao thức phân giải địa chỉ ARP

Địa chỉ IP được dùng để *định danh các host và mạng ở tầng mạng* của mô hình OSI, và chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm đó trên cùng một mạng cục bộ (Ethernet, Token Ring, ...). Trên một LAN như vậy, hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau.

Vấn đề đặt ra là phải thực hiện ánh xạ địa chỉ IP (32 bits) và địa chỉ vật lý (48 bits) của một trạm. Giao thức ARP (Address Resolution Protocol) đã được xây dựng để chuyển đổi từ địa chỉ IP sang địa chỉ vật lý khi cần thiết. Ngược lại, giao

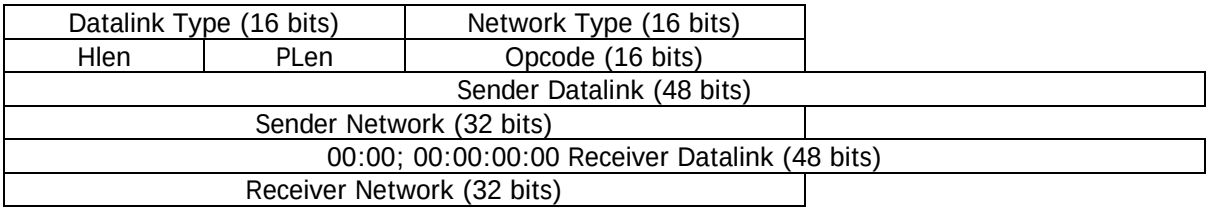
thức RARP (Reserse Address Resolution Protocol) được dùng để chuyển đổi từ địa chỉ vật lý sang địa chỉ IP.

Cả hai giao thức ARP và RARP đều không phải là bộ phận của IP, IP sẽ dùng đến chúng khi cần.

Mỗi ghép nối mạng có *địa chỉ giao thức mạng* (IP address) và *địa chỉ giao thức liên kết dữ liệu* (Datalink Protocol Address) riêng. Do đó cần có bảng ánh xạ giữa hai địa chỉ này (địa chỉ ảo và địa chỉ vật lý). Bảng địa chỉ này có thể làm bằng tay, nhưng do khối lượng địa chỉ lớn, tăng khá nhanh, nên người ta giải quyết thông qua thủ tục "*Tìm giải pháp cho địa chỉ*" (Address Resolution Protocol -ARP).

Các gói tin ARP được đóng gói trong khung dữ liệu liên kết (data link frame). Đối với mạng Ethernet, kiểu trường (type field) sẽ là 0x0806.

ARP ánh xạ địa chỉ IP sang địa chỉ liên kết dữ liệu (datalink address). Trạm tin sẽ gửi gói tin yêu cầu ARP (request packet) với khuôn dạng gói tin như hình sau.



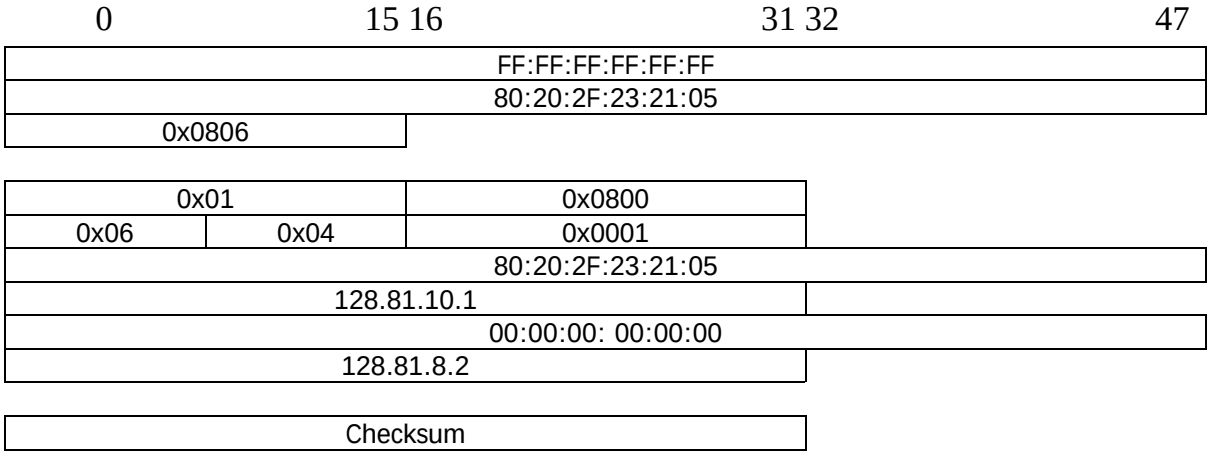
Hình 5-12. Khuôn dạng gói tin ARP.

- Data link type: Loại dữ liệu liên kết, với mạng Ethernet thì trường này có giá trị là 0x0001
- Network type : Loại địa chỉ mạng, Ethernet type used for IP (0x0800)
- Hlen : Độ rộng của phần địa chỉ dữ liệu liên kết, với mạng Ethernet độ rộng là 6 bytes
- Plen : Độ rộng của địa chỉ mạng, trong giao thức IP, phần này là 4 byte
- Opcode : Có giá trị là 0x0001 cho thủ tục yêu cầu ARP, 0x0002 cho ARP trả lời.
- Sender datalink and sender network : Địa chỉ vật lý và địa chỉ ảo (địa chỉ mạng) của người gửi
- Receive datalink and receive network : Địa chỉ vật lý và địa chỉ ảo (địa chỉ mạng) của người nhận

Ví dụ: Trạm A muốn gửi trạm B một gói tin IP. Cả hai máy A, B đều có cùng có địa chỉ mạng IP và cùng kết nối vào mạng Ethernet như hình sau :

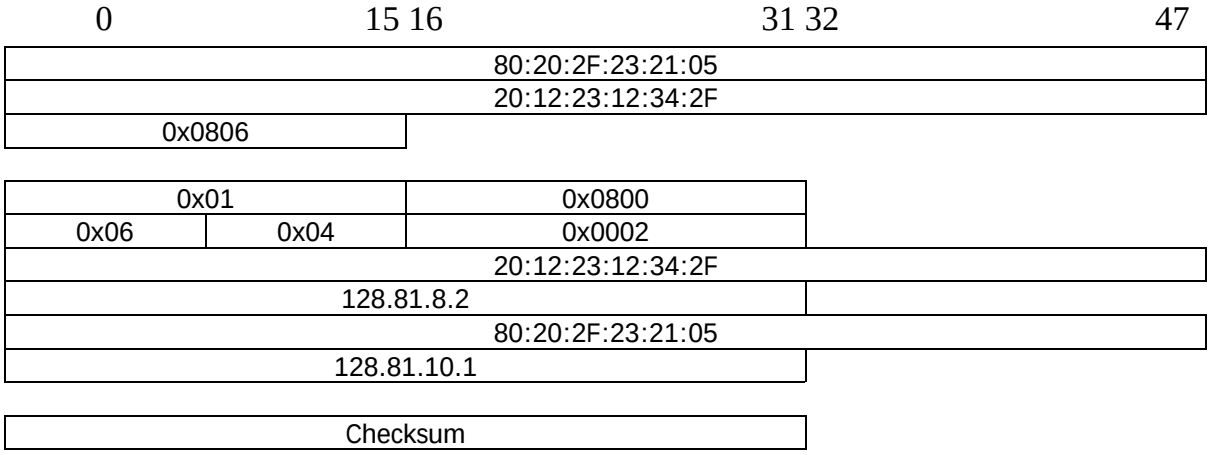
Trạm A biết được địa chỉ mạng của trạm B nhưng không biết địa chỉ vật lý của trạm B. Trạm A cần hỏi địa chỉ vật lý của trạm B để gửi tin. Khi đó trạm A phát đi một gói tin ARP yêu cầu (ARP request packet) đóng gói trong khung tin Ethernet.

- Quá trình gởi yêu cầu ARP



Hình 5-13. Khuôn dạng gói tin ARP yêu cầu.

Gói tin yêu cầu ARP (ARP request packet) được gửi tới các trạm, chỉ trạm B là đúng địa chỉ IP. Trạm B sẽ tạo ARP trả lời :



Hình 5-14. Khuôn dạng gói tin ARP trả lời.

Trạm B bổ sung IP_ to_Ethernet Address entry của host A và ARP cache của B

Trạm A bổ sung IP_to_Ethernet Address entry của host B và ARP cache của A

Như vậy bảng ánh xạ tự động bổ sung những đường dẫn (entry) mới mà nó biết, đồng thời cũng huỷ bỏ những đường dẫn (entry) mà nó không dùng đến.

5.6.2 Giao thức RARP (Reverse Address Resolution Protocol)

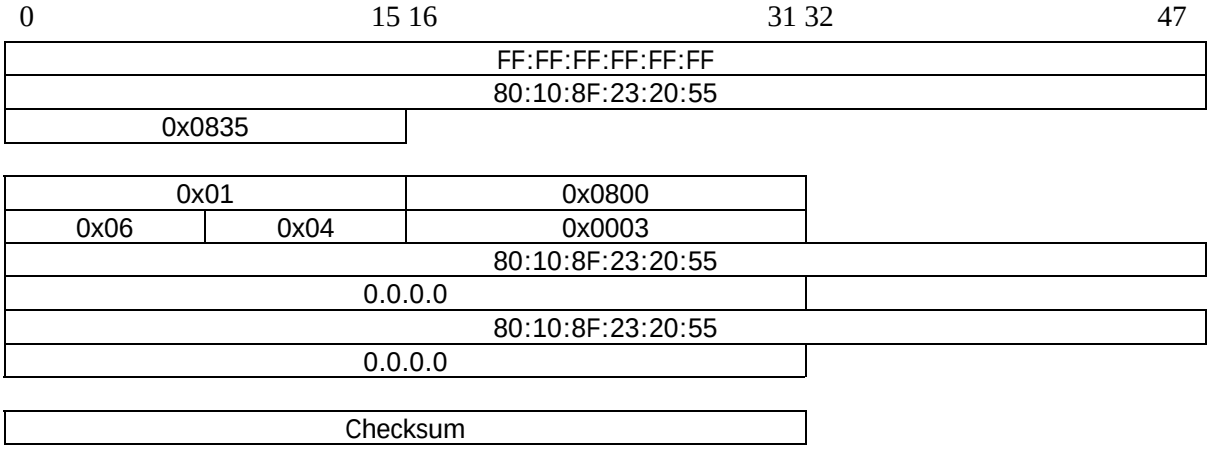
Đôi khi ta cần ánh xạ ngược lại.

Ví dụ một trạm không ổ đĩa, biết địa chỉ vật lý (datalink address) tức là địa chỉ card mạng giữ ở bộ nhớ ROM, nhưng không biết địa chỉ IP vì không có ổ đĩa. Khi này cần ánh xạ từ địa chỉ vật lý sang địa chỉ mạng.

Ta cũng làm như trên, nhưng thay kiểu trường từ 0x0806 bằng 0835.

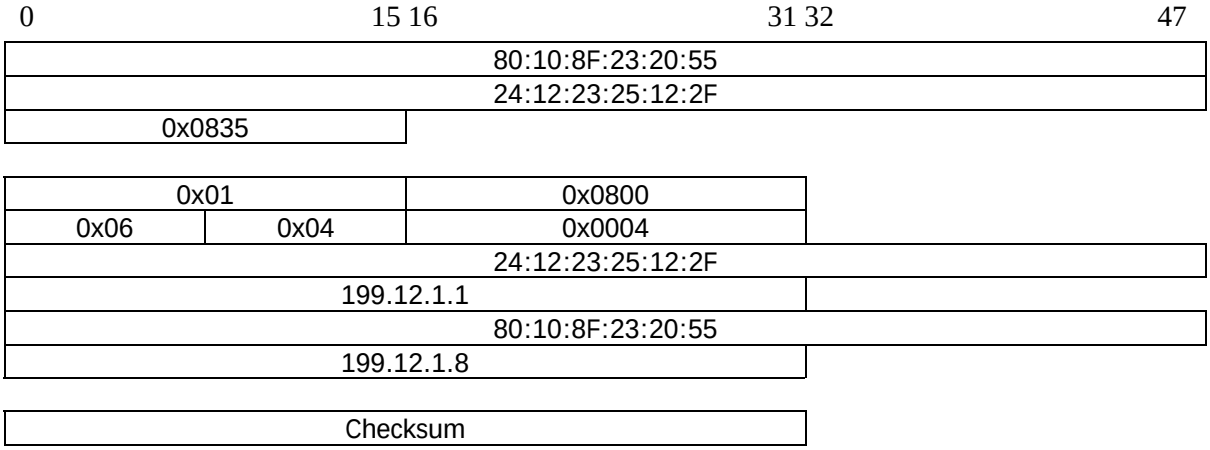
Yêu cầu chuyển đổi (reverse request) là 0x0003 và trả lời chuyển đổi (reverse reply) là 0x0004.

- Quá trình gởi yêu cầu RARP



Hình 5-15. Khuôn dạng gói tin RARP yêu cầu.

- Quá trình gởi trả lời RARP



Hình 5-16. Khuôn dạng gói tin trả lời RARP .

5.6.3 Giao thức ICMP

Giao thức ICMP (Internet Control Message Protocol) thực hiện truyền các thông tin điều khiển (các báo cáo về các tình trạng lỗi trên mạng, ...) giữa các

gateway hoặc các máy chủ trên liên mạng theo giao thức IP. Tình trạng lỗi có thể là: một datagram không thể đến được đích của nó, hoặc một router không đủ bộ nhớ đệm để lưu và chuyển một datagram, ... Một thông báo ICMP được khởi tạo và chuyển cho IP. IP sẽ bọc (*encapsulate*) thông báo đó với một IP header và truyền đến cho router hoặc trạm đích.

5.6.3.1 Ccác thành phần của thông báo ICMP hỗ trợ xác định lỗi và truy vấn

Thông báo ICMP được chia làm 2 loại: thông báo lỗi ICMP và thông báo truy vấn ICMP.

Các thông báo ICMP khác nhau về định dạng tuỳ vào chức năng của từng loại, nhưng kiến trúc tổng quát bao gồm 2 phần: phần đầu (ICMP header) và phần dữ liệu (ICMP data).

Phần đầu của thông báo ICMP luôn bắt đầu bằng 3 trường:

a. TYPE: 8 bits, xác định loại thông báo ICMP.

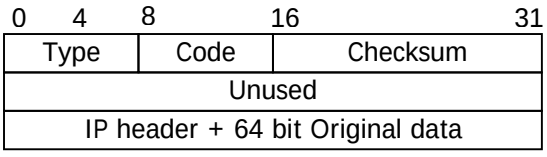
b. CODE: 8 bits, cung cấp thông tin chi tiết của từng loại thông báo ICMP.

c. CHECKSUM: 16 bits, xác định sự toàn vẹn dữ liệu trong quá trình truyền.

1. Các thông báo lỗi ICMP

Về mặt kỹ thuật, ICMP được thiết kế để cung cấp các thông tin về trạng thái không ổn định và thực hiện thông báo các trường hợp lỗi phát sinh của hệ thống phần cứng cũng như phần mềm làm ngăn chặn, huỷ bỏ quá trình gửi, nhận hoặc xử lý các datagram trên mạng Internet trước khi được chuyển đến đích cuối cùng.

Có 5 loại thông báo lỗi ICMP trong bảng I.1 và các thông báo có dạng chung như hình sau :



Hình 5-17. Dạng chung thông báo lỗi của ICMP

Type	Thông báo lỗi ICMP
3	Destination Unreachable
4	Source Quench
5	Redirect
11	Time Exceeded
12	Parameter Problem

Bảng I.1: Các loại thông báo lỗi của ICMP

Original IP header: 20-60 bytes chứa IP header của gói bị lỗi.

Original data: 8 bytes, chứa nội dung 64 bits đầu tiên của gói dữ liệu bị lỗi.

- *Destination Unreachable*

Các thông báo ICMP Destination Unreachable được tạo ra khi không thể chuyển đến 1 đích được xác định trong IP datagram. Bao gồm các loại lỗi sau:

Code	Nội dung thông báo ICMP
0	Network Unreachable
1	Host Unreachable
2	Protocol Unreachable
3	Port Unreachable
4	Fragmentation needed and DF flag set
5	Source Route Fail
6	Destination Network unknown
7	Destination Host unknown
8	Source Host Isolated
9	Communication with Destination Network is Administratively Prohibited
10	Communication with Destination Host is Administratively Prohibited

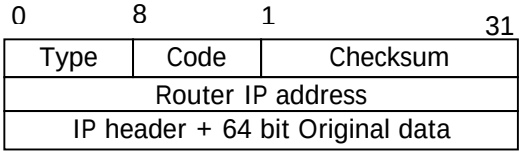
Bảng 5-1. Các lỗi của ICMP Destination Unreachable

- *Source Quench :* Khi vùng đệm của hệ thống nhận không đủ chỗ trống lưu trữ, hệ thống sẽ phát ra thông báo Source Quench. Trường CẶDQ của thông báo này luôn luôn nhận giá trị 0.
- *Redirect :* Một thông báo ICMP Redirect được tạo ra bởi 1 router trong trường hợp nó nhận thấy rằng một máy tính đang sử dụng con đường định tuyến không tối ưu.

Trường CẶDQ nhận 4 giá trị trong bảng và có định dạng như hình sau:

Code	Nội dung
0	Redirect for the network (or subnet)
1	Redirect for the host
2	Redirect for the type of service and network
3	Redirect for the type of service and host

Bảng 5-2. Các lỗi của ICMP Redirect



Hình 5-18. Dạng ICMP Redirect

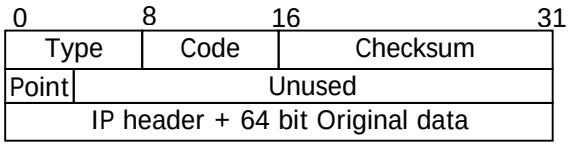
Router ip address là địa chỉ của bộ định tuyến mà máy nguồn sẽ dùng để trỏ máy đích.

- *Time Exceeded :* Router sẽ huỷ bỏ, không xử lý 1 datagram khi giá trị TTL của nó bằng 0 và phát ra một thông báo ICMP Time Exceeded. Có 2 loại ICMP Time Exceeded như sau:

Code	Nội dung
0	Bộ đếm thời gian sống TTL của 1 datagram bằng 0
1	Quá thời gian đợi để kết hợp các gói bị phân mảnh

Bảng 5-3. Các lỗi của ICMP Time Qxceeded.

- *Parameter Problem :* Thông báo này được gửi đi khi có lỗi xuất hiện ở phần các tham số chọn lựa của datagram gửi đến. Trường CẶDỌ của thông báo này nhận 3 giá trị trong bảng và có định dạng như hình sau :



Hình 5-19. Dạng ICMP Parameter Problem

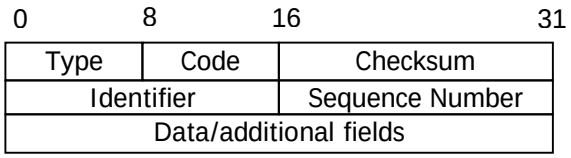
Code	Giải thích
0	Có một lỗi đặc biệt trong lược đồ dữ liệu.
1	Phần option của IP header chưa định nghĩa.
2	Lỗi Header Length và (hoặc) Total Packet Length trong IP header.

Bảng 5-4. Các lỗi của ICMP Parameter Problem

Pointer: xác định vị trí gây ra lỗi trong datagram.

2. Các thông báo truy vấn ICMP

ICMP được sử dụng trong việc khảo sát các đặc trưng chung của mạng với 2 loại thông báo request và reply. Có 8 loại thông báo truy vấn ICMP được liệt kê trong bảng và có định dạng như hình sau :



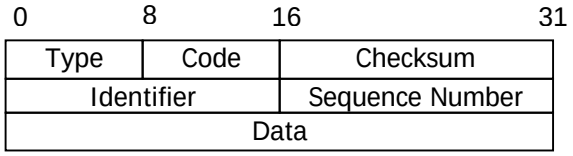
Hình 5-20. Dạng ICMP truy vấn.

Type	Loại thông báo
0	Echo Reply
8	Echo Request
13	Timestamp Request
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply

Bảng 5-5. Các loại thông báo truy vấn ICMP.

- Identifier được sử dụng để phân biệt các thông báo được gởi đến các host khác nhau.
- Sequence number được sử dụng để phân biệt các thông báo được gửi đến cùng một host.
- Data/additional fields được dùng theo từng loại thông báo truy vấn ICMP.

- *Echo Request vµ Echo Reply*



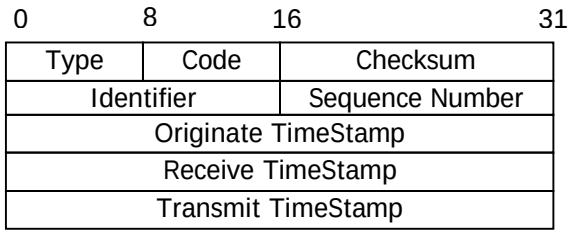
Hình 5-21. Dạng ICMP Echo Request & Reply.

Người ta sử dụng ICMP Echo để xác định xem một địa chỉ IP đích còn hoạt động hay không bằng cách gửi thông báo ICMP Echo Request đến hệ thống đích và chờ xem nếu nhận được thông báo ICMP Echo Reply thì sẽ xác định đích đấy vẫn còn hoạt động ngược lại thì đã bị down. Định dạng thông báo như trong hình sau :

Kích thước của DATA thay đổi tuỳ thuộc vào từng loại hệ điều hành. Trong hệ điều hành UNIX, kích thước của nó là 56 bytes, trong Microsoft Windows là 32 bytes,...

- *Timestamp Request và Timestamp Reply*

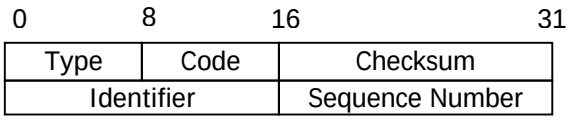
Mỗi máy đều có 1 đồng hồ riêng xác định thời gian vận hành của nó, quá trình hoạt động trong những hệ thống phần mềm phân tán thì sự khác biệt nhau lớn về thời gian giữa các máy tính sẽ gây ra nhiều vấn đề khó khăn. ICMP cung cấp một cơ chế cho phép lấy thời gian từ một máy khác và có định dạng như hình sau.



Hình 5-22. ICMP Timestamp Request & Reply

- Originate timestamp là thời gian máy nguồn thực hiện gửi báo.
- Receive timestamp là thời gian đầu tiên máy đích nhận được thông báo.
- Transmit timestamp là thời gian cuối bên đích xử lý thông báo và gửi đi.

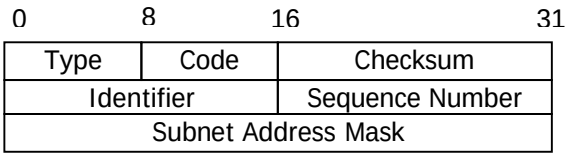
- *Information request và reply*



Hình 5-23. ICMP information request & reply

Được sử dụng nhằm hỗ trợ các hệ thống máy trạm không đĩa khi khởi động; cho phép các máy tính tìm ra địa chỉ Internet của chúng lúc khởi động hệ thống.

- *Address Mask Request và Reply*



H×nh 5-24. ICMP Ađress Mask Request & Reply

Để biết subnet mask, máy sẽ gửi một thông báo ICMP Address Mask Request đến 1 router và chờ nhận thông báo ICMP Address Mask Reply. Subnet Address Mask chứa địa chỉ của mặt nạ con của mạng.

Các bộ định tuyến phát bản tin ICMP để báo cho các trạm biết : gói tin không tới, hoặc tồn tại đường đi tốt hơn. Một số trường hợp có thể xảy ra là :

- *Destination unreachable* (không tới được đích): Bản tin không tới được đích do có lỗi hoặc không tìm được đường đi.
- *Routing redirect* (đổi đường đi): Thay đổi đường đi của bản tin do tồn tại đường đi tối ưu hơn (yêu cầu đổi đường đi).
- *Time expirect* (hết thời gian): Hết thời hạn khi TTL về 0 (timeout).
- *Echo request và cho echo reply* : Xuất hiện yêu cầu và trả lời.

  ICMP được dùng vào việc gỡ rối mạng cho biết tình trạng của mạng.

Lệnh Ping ***(Packet Internet Oroper)*** được dùng để hỏi (query) hệ thống (máy tính) khác để đảm bảo rằng một kết nối vẫn đang hoạt động (active). Lệnh Ping hoạt động bằng cách gửi ra một yêu cầu phản hồi (echo request) ICMP (Internet Control Message Protocol). Nếu như phần mềm IP của máy tính nhận được yêu cầu ICMP đó, nó đưa ra một trả lời phản hồi (echo reply) ngay lập tức. Máy gửi lại tiếp tục gửi một yêu cầu phản xạ cho đến khi lệnh ping được kết thúc bằng một tổ hợp phím thoát (Ctrl+C hoặc phím Delete trên UNIX).

5.7 Phiên bản IPv6

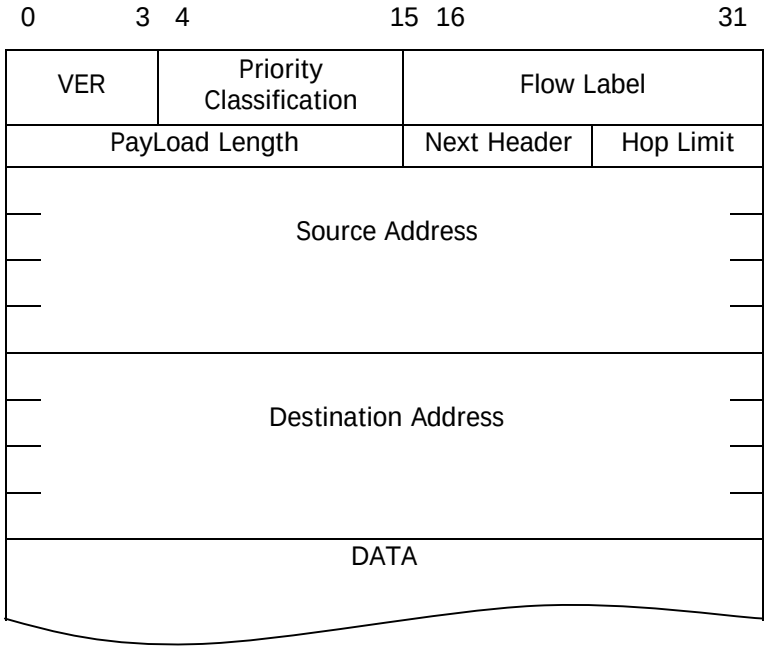
Với sự phát triển nhanh chóng của Internet thì địa chỉ IP 32 bit không thể đáp ứng được nhu cầu sử dụng Internet. Để khắc phục điều này phiên bản IP6 (IP Next Generation) đang được phát triển. Phiên bản IPv6 có các thay đổi như sau :

- Sử dụng 128 bit địa chỉ mạng thay cho 32 bit địa chỉ như phiên bản IPv4.
- Mở rộng phần Header cho ứng dụng và lựa chọn của khung tin.
- Hỗ trợ các loại dữ liệu audio và video.

- Có các giao thức mở rộng : cho phép bổ sung nhiều thông tin vào một datagram.

5.7.1 Khung tin IPng v6

Phần Header của các khung tin Ipng đã được thay đổi so với phiên bản 4. Phần lớn sự thay đổi của IPng là địa chỉ IP 128 bit và bỏ các trường không cần thiết. Cấu tạo của khung tin IPng như sau :



H×nh 5-25. CÊu t¹o cña gãi tin IPv6.

5.8 Định tuyến trên Internet

5.8.1 Bảng chọn đường

Một số phương thức thông thường xây dựng một bảng chọn đường (routing *table*) như sau :

- Bảng cố định được tạo ra dựa vào sơ đồ của mạng, bảng này liên tục được thay đổi và được cập nhật lại mỗi khi có sự thay đổi vật lý ở bất cứ nơi nào của mạng.
- Bảng động được dùng để ước lượng về đường truyền và các thông điệp từ các nút khác để điều chỉnh lại thông tin của bảng bên trong.
- Bảng dẫn đường cố định chính được tải về từ một trung tâm của các nút mạng trong một khoảng thời gian nhất định hoặc được tải về khi cần thiết.

Mỗi một phương thức đều có các ưu, nhược điểm của nó. Bảng động được đặt ở từng nút mạng hoặc được tải về trong những khoảng thời gian nhất định từ một

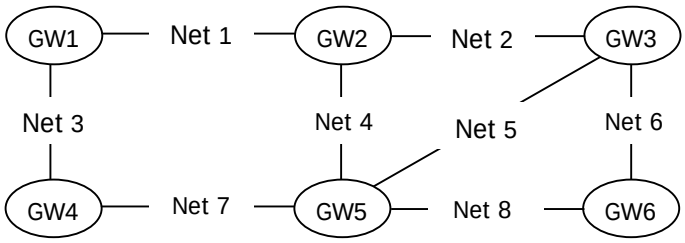
nơi chứa bảng cố định, nó không phức tạp và thích ứng với những thay đổi nhanh chóng trên mạng. Bảng chính thường là tốt hơn bảng cố định bởi vì quản lý một bảng ở trung tâm sẽ dễ dàng hơn quản lý từng bảng được đặt tại mỗi nút mạng.

5.8.2 Xây dựng bảng chọn đường cho các Router/Gateway

Trong liên mạng, tại mỗi cổng phải có một bảng chọn đường để chỉ ra muốn đến mạng đích nào thì phải đến cổng tiếp theo là cổng nào. Bảng chọn đường gồm hai phần : phần bên trái là mạng đích, nơi muốn đến, phần bên phải là khoảng cách tới đó và cổng tiếp theo.

Để xây dựng bảng chọn đường, từ cổng đang đứng ta xét các mạng cạnh đó, sau đó là các mạng ở cạnh các cổng tiếp theo và cứ thế cho đến hết các mạng trong liên mạng.

Ví dụ 1: Lập bảng chọn đường cho các router.gateway của liên mạng sau :



GW1	
Neti	D,G
1	0,1
2	1,2
3	0,1
4	1,2
5	2,2(4)
6	2,2
7	1,4
8	2,2(4)

GW2	
Neti	D,G
1	0,2
2	0,2
3	1,1
4	0,2
5	1,3(5)
6	1,3
7	1,5
8	1,5

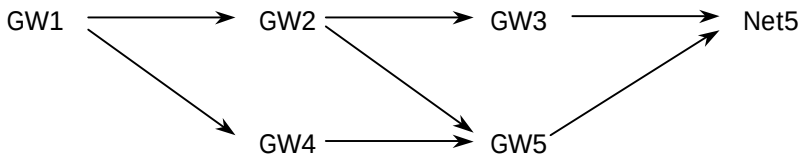
GW3	
Neti	D,G
1	1,2
2	0,3
3	2,2
4	1,2(5)
5	0,3
6	0,3
7	1,5
8	1,5(6)

GW4	
Neti	D,G
1	1,2
2	2,1(5)
3	0,4
4	1,5
5	1,5
6	2,5
7	0,4
8	1,5

GW5	
Neti	D,G
1	1,2
2	1,2
3	1,4
4	0,5
5	0,5
6	1,3(6)
7	0,5
8	0,5

GW6	
Neti	D,G
1	2,3(5)
2	1,3
3	2,5
4	1,5
5	1,3(5)
6	0,6
7	1,5
8	0,6

Dựa vào bảng chọn đường, tìm đường đi từ GW1 tới Net 5 như sau :



Đối với nhiều host, bảng dẫn đường tĩnh hoạt động như sau :

- Nếu đích nằm trong mạng cục bộ, dữ liệu được gởi đến máy đích
- Nếu đích nằm trên mạng ở xa, dữ liệu được chuyển tiếp đến gateway cục bộ.

Tuỳ thuộc vào kích cỡ của mạng mà các giao thức chọn đường khác nhau sẽ được sử dụng. Giao thức chọn đường trong một hệ thống nội bộ là RIP (Routing Information Protocol). Giao thức chọn đường giữa các hệ thống là EGP (External Gateway Protocol) và BGP (Border Gateway Protocol).

5.9 Mạng X.25

Vào những năm cuối thập niên 70, người ta phải cần đến một loạt các giao thức để cung cấp cho những người sử dụng mạng diện rộng WAN kết nối thông qua mạng dữ liệu công cộng (Public Data Networks - PDNs). Các loại hình PDNs như TELENET và TYMNET đã đạt được những thành công đáng ghi nhận, nhưng việc tiêu chuẩn hóa giao thức dường như còn ngoài tầm những người sử dụng mạng PDNs do việc đòi hỏi tính tương thích của thiết bị ngày một cao và đồng thời chi phí phải thấp. Kết quả của sự nỗ lực không ngừng này là sự ra đời của một loạt giao thức, trong đó X.25 được xem là giao thức phổ biến nhất.

Mạng X.25 và các giao thức liên quan do một tổ chức Quốc gia gọi là Hiệp hội Viễn thông Quốc tế (ITU) quản lý. Ban chịu trách nhiệm về các nghiệp vụ truyền tín hiệu âm thanh và dữ liệu của ITU gọi là ủy ban Tư vấn Quốc tế về Điện thoại và Điện báo (CCITT). Các thành viên của CCITT bao gồm FCC, PTTs Âu châu, các doanh nghiệp truyền thông và nhiều hãng máy tính, truyền dữ liệu khác. Do nhiều thành quả đóng góp trực tiếp có tính kế thừa, mạng X.25 thực sự được xem là mạng tiêu chuẩn có tính toàn cầu.

5.9.1 Cơ sở kỹ thuật

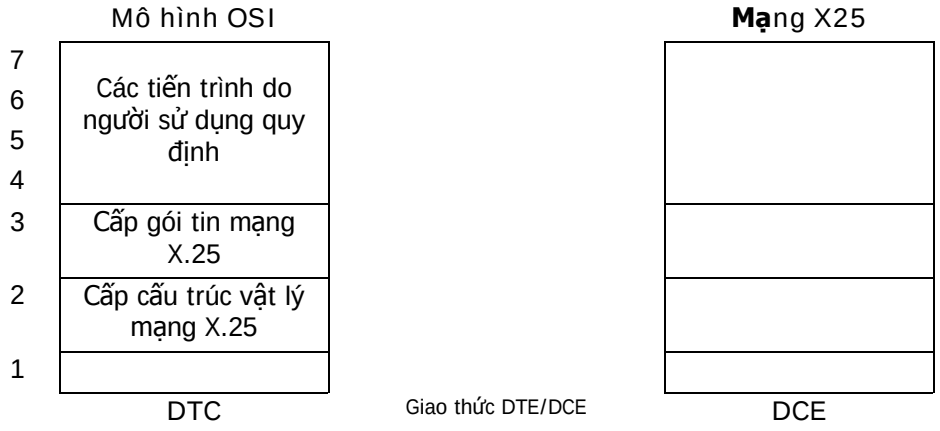
Mạng X.25 là một mạng điện thoại dùng để truyền dữ liệu. Để bắt đầu thực hiện quá trình giao tiếp, một máy tính cần phải liên kết với một máy khác để yêu cầu thực hiện giao tiếp. Máy được yêu cầu liên kết có thể chấp nhận hoặc từ chối việc giao tiếp. Nếu liên kết được chấp nhận, hai hệ thống có thể bắt đầu truyền tải thông tin qua lại hai chiều đồng thời với nhau. Cả hai bên đều có thể chấm dứt việc giao tiếp vào bất cứ thời điểm nào tùy ý.

Các đặc tính của mạng X.25 cho phép xác định quá trình tương tác từ nút-đến-nút (point-to-point) giữa các thiết bị truyền dữ liệu đầu cuối (Data Terminal Equipment - DTE) với các thiết bị kết cuối mạch truyền dữ liệu (Data Circuit-terminating Equipment - DCE). DTEs (bao gồm các trạm đầu cuối và máy chủ của người sử dụng mạng) kết nối với DCEs (bao gồm modem, các gói tin và các cổng truy cập PDN, thường đặt tại các trạm truyền thông), DCEs lại nối kết vào kênh chuyển mạch gói (Packet Switching Exchanges - PSEs) và các DCEs khác trong mạng PSNs và cuối cùng đến một DTE khác.

Một DTE có thể xem là một trạm đầu cuối nhưng không thực hiện đầy đủ các chức năng của mạng X.25. Các DTE được nối kết với DCE thông qua một thiết bị chuyển đổi gọi là thiết bị ghép/tách gói tin (Packet Assembler/Disassembler - PAD).

Quá trình hoạt động của mạch ghép nối từ trạm đầu cuối đến PAD, các dịch vụ do PAD cung cấp và các tương tác giữa PAD và các máy chủ do CCITT quy định.

Sơ đồ đặc tính của mạng X.25 kiểu phân tầng từ 1 tới 3 theo mô hình tham chiếu cho việc nối kết các hệ thống mở OSI. Tầng 3 của mạng X.25 mô tả các quy trình định dạng và chuyển mạch gói giữa các thành tố tầng 3 ngang cấp. Tầng 2 của mạng X.25 do các thủ tục truy cập liên kết cân bằng (Link Access Procedure Balance - LAPB) kiểm soát. LAPB xác lập các đơn vị gói tin (packet framing) cho các liên kết DTE/DCE. Tầng 1 của mạng X.25 xác lập các thủ tục về điện và cơ để kích hoạt và chấm dứt quá trình kết nối vật lý của DTE và DCE. Mối quan hệ này được minh họa theo hình vẽ dưới đây. Chú ý rằng tầng 2 và 3 cũng tham chiếu theo tiêu chuẩn ISO 7776 (LAPB) và ISO 8208 (các tầng gói tin mạng X.25).



Hình 5-26. Mối quan hệ giữa các tầng trong mạng X.25.

Quá trình giao tiếp từ nút-tới-nút (end-to-end) giữa các DTEs được thực hiện hoàn thiện thông qua một sự kết nối song phương gọi là liên kết truyền ảo (virtual circuit). Các liên kết ảo cho phép các hệ mạng khác nhau có thể giao tiếp được với nhau thông qua mọi nút liên kết trung gian mà không cần đến các bộ phận chuyên dụng để định rõ các liên kết vật lý. Các liên kết ảo hoặc có thể duy trì vĩnh viễn hoặc có thể tạm thời. Liên kết ảo vĩnh viễn được gọi là PVCs (Permanent Virtual Circuits), liên kết ảo tạm thời được gọi là SVCs (Switched Virtual Circuits). PVCs chủ yếu áp dụng cho phương thức truyền dữ liệu thường xuyên còn SVCs được áp dụng cho phương thức truyền dữ liệu không thường xuyên. Tầng 3 của mạng X.25 liên quan tới phương thức giao tiếp từ nút tới nút bao gồm cả hai liên kết ảo PVCs và SVCs.

Một khi đã thiết lập liên kết ảo, PTE có thể thực hiện truyền một gói tin đến một PTE khác bằng cách chuyển gói tin đến DCE thông qua một liên kết ảo thích hợp. Sau đó DCE sẽ tiến hành ền ưu tiên của liên kết ảo để định ra thức truyền gói tin lên mạng X.25. Các giao thức của tầng 3 mạng X.25 sẽ tiến hành chèn thông tin

vào giữa các DTE được kiểm soát bởi DCE của mạng phía nhận gói tin rồi sau đó được chuyển đến DTE đích.

5.10 Kỹ thuật FRAME RELAY

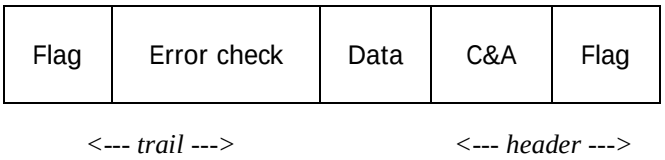
Bước sang thập kỷ 80 và đầu thập kỷ 90, công nghệ thông tin có những bước tiến đặc biệt là chế tạo và sử dụng cáp quang vào mạng truyền dẫn tạo nên chất lượng thông tin rất cao. Sử dụng giao thức X25 để truyền đa số liệu trên mạng cáp quang, dữ liệu nhận được có thể đánh giá là đạt yêu cầu. Tuy nhiên người ta nhận thấy rằng sử dụng giao thức này làm mất rất nhiều thời gian để truyền số liệu trên mạng cáp quang. Do đó công nghệ Frame Relay ra đời có thể chuyển nhận các khung lớn tới 4096 byte trong khi đó gói tiêu chuẩn của X25 khuyến cáo dùng là 128 byte, không cần thời gian cho việc hỏi đáp, phát hiện lỗi và sửa lỗi ở lớp 3 (*No protocol at Network Layer*) nên Frame Relay có khả năng chuyển tải nhanh hơn hàng chục lần so với X25 ở cùng tốc độ. Frame Relay rất thích hợp cho truyền số liệu tốc độ cao và cho kết nối LAN to LAN và cả cho âm thanh, nhưng điều kiện tiên quyết để sử dụng công nghệ Frame relay là chất lượng mạng truyền dẫn phải cao.

Frame-Relay bắt đầu được đưa ra như tiêu chuẩn của một trong những giao thức truyền số liệu từ năm 1984 trong hội nghị của ủy ban Tư vấn Quốc tế về Điệnthoại và Điện báo CCITT và cũng được Viện tiêu chuẩn quốc gia Mỹ ANSI đưa thành tiêu chuẩn của ANSI vào năm đó.

Mục tiêu chính của Frame-Relay cũng giống như của nhiều tiêu chuẩn khác, đó là tạo ra một giao diện chuẩn để kết nối thiết bị - của các nhà sản xuất thiết bị khác nhau - giữa người dùng và mạng UNI (*User to Network Interface*). Frame-Relay được thiết kế nhằm cung cấp dịch vụ chuyển khung nhanh cho các ứng dụng số liệu tương tự như X.25 hay ATM.

Mạng truyền số liệu theo công nghệ chuyển mạch gói X.25 chỉ có thể phục vụ cho các nhu cầu truyền số liệu tốc độ thấp (tối đa tới 128 Kbps) nhưng nó có tính an toàn cao, khắc phục được các yếu điểm của một mạng truyền dẫn chất lượng kém. Với các công nghệ truyền dẫn hiện nay, vấn đề nâng cấp chất lượng các đường truyền dẫn không còn quá phức tạp như trước kia. Vì vậy, chúng ta còn có thể chọn hướng phát triển là xây dựng mạng truyền số liệu theo công nghệ Frame-relay và tiến tới công nghệ ATM.

5.10.1 Khuôn dạng gói dữ liệu Frame-Relay



Hình 5-27. Khuôn dạng gói dữ liệu Frame-Relay.

- Flag: Cờ
- Error check: Trường kiểm tra lỗi
- Data: Trường dữ liệu
- C&A: Trường địa chỉ và điều khiển

Để thực hiện nhiệm vụ truyền số liệu, mạng Frame-Relay sẽ phải giải quyết vấn đề tắc nghẽn thông tin trên mạng, thực chất đây là vấn đề của tầng Mạng trong mô hình 7 tầng. Frame-Relay làm việc ở tầng Liên kết nhưng cũng phải giải quyết vấn đề này để đảm bảo khả năng lưu chuyển thông tin. Hầu hết các mạng truyền số liệu đều sử dụng kỹ thuật điều khiển luồng để giải quyết vấn đề tắc nghẽn. Có hai phương pháp được sử dụng khi xảy ra tắc nghẽn trong mạng: thông báo cho người dùng, router, chuyển mạch về sự cố tắc nghẽn xảy ra và thực hiện các công việc nhằm hiệu chỉnh luồng thông tin. Cả hai phương pháp này mạng Frame-Relay đều dùng đến các bit BECN (Backward Explicit Congestion Notification) và bit FECN (Forward Explicit Congestion Notification) trong trường điều khiển.

Bit FECN được thiết lập khi có tắc nghẽn để thông báo rằng thủ tục xử lý tắt nghẽn đã được khởi tạo, và tương ứng với lưu lượng bị nghẽn từ hướng của Frame có bit FECN tới. Ngược lại, bit BECN cũng được thiết lập khi có tắc nghẽn để thông báo rằng thủ tục xử lý nghẽn đã được khởi tạo, nhưng tương ứng với lưu l-ượng bị nghẽn từ hướng ngược với Frame có bit BECN tới. Khi các bit này được thiết lập thì mạng phải dùng đến một liên kết logic dự phòng để chuyển các thông tin để xử lý nghẽn, đó là liên kết với mã nhận dạng DLCI (Data Link Connection Identifier) số 1023. Các liên kết với mã nhận dạng nhỏ hơn được dùng để truyền số liệu của người dùng.

BÀI TẬP

1. Viết sơ đồ mô tả thuật giải hoạt động chọn đường trên mạng.
2. Khảo sát cấu trúc và hoạt động của giao thức điều khiển ICMP
3. Tìm hiểu các lệnh của hệ điều hành Windows và Linux để xem và thay đổi các thông số bảng chọn đường.

Chương 6

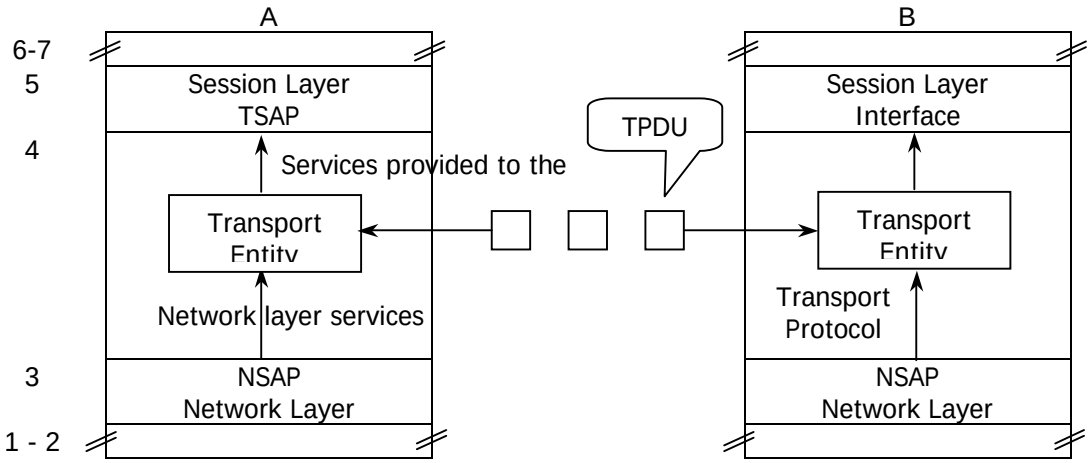
TẦNG GIAO VẬN

Tầng giao vận làm nhiệm vụ thiết lập, duy trì và huỷ bỏ các cuộc giao tiếp giữa hai máy, đảm bảo việc dữ liệu truyền giống hoàn toàn dữ liệu nhận. Dữ liệu qua các mạng con có thể bị lỗi, tập tin tầng giao vận thực hiện cải thiện chất lượng dịch vụ, đảm bảo dữ liệu được truyền một cách chính xác và truyền lại nếu như phát hiện thấy lỗi. Tầng giao vận *quản lý dữ liệu gửi*, *xác định trật tự* của dữ liệu và *độ ưu tiên* của dữ liệu đó.

6.1 Các vấn đề của tầng giao vận

6.1.1 Cung cấp dịch vụ cho tầng phiên

Để thực hiện mục tiêu chuyển giao dữ liệu tin cậy, an toàn cho tầng 5, tầng 4 phải dùng các dịch vụ được cung cấp từ tầng 3 (network layer). Phần cứng và phần mền trong phần 4 để thực hiện công việc coi là thực thể giao vận (*transport entity*). Mối quan hệ giữa các lớp 3, 4, 5, được mô tả bởi hình sau:



Hình 6-1. Mối quan hệ giữa các thực thể trong tầng Phiên.

Có hai dịch vụ mạng nên cũng có hai dịch vụ giao vận: *dịch vụ có kết nối* và *không kết nối*.

Do dữ liệu qua các subnet có thể sai sót, người sử dụng không có được điều khiển trên subnet hoặc tăng cường quản lý lỗi ở tầng hai. Chỉ có khả năng đặt thêm một tầng trên lớp 3 để cải thiện chất lượng dịch vụ (QoS). Nếu giữa chúng một tầng giao vận được kết nối mạng được kết thúc đột ngột và không biết được sự cố gì đã xảy ra, nó có thể thiết lập một kết nối mới ở lớp mạng tới tầng giao vận ở xa và gởi yêu cầu hỏi số liệu nào đến, số liệu nào không tự nó biết được sai sót xảy ra ở đâu. Tầng 4 có thể phát hiện mất gói tin, số liệu bị biến đổi, N-RESET ở lớp mạng. Tầng 1 -> 4 cung cấp dịch vụ giao vận. Tầng 5 ->7 sử dụng dịch vụ giao vận

- Các hàm dịch vụ của tầng giao vận có kết nối

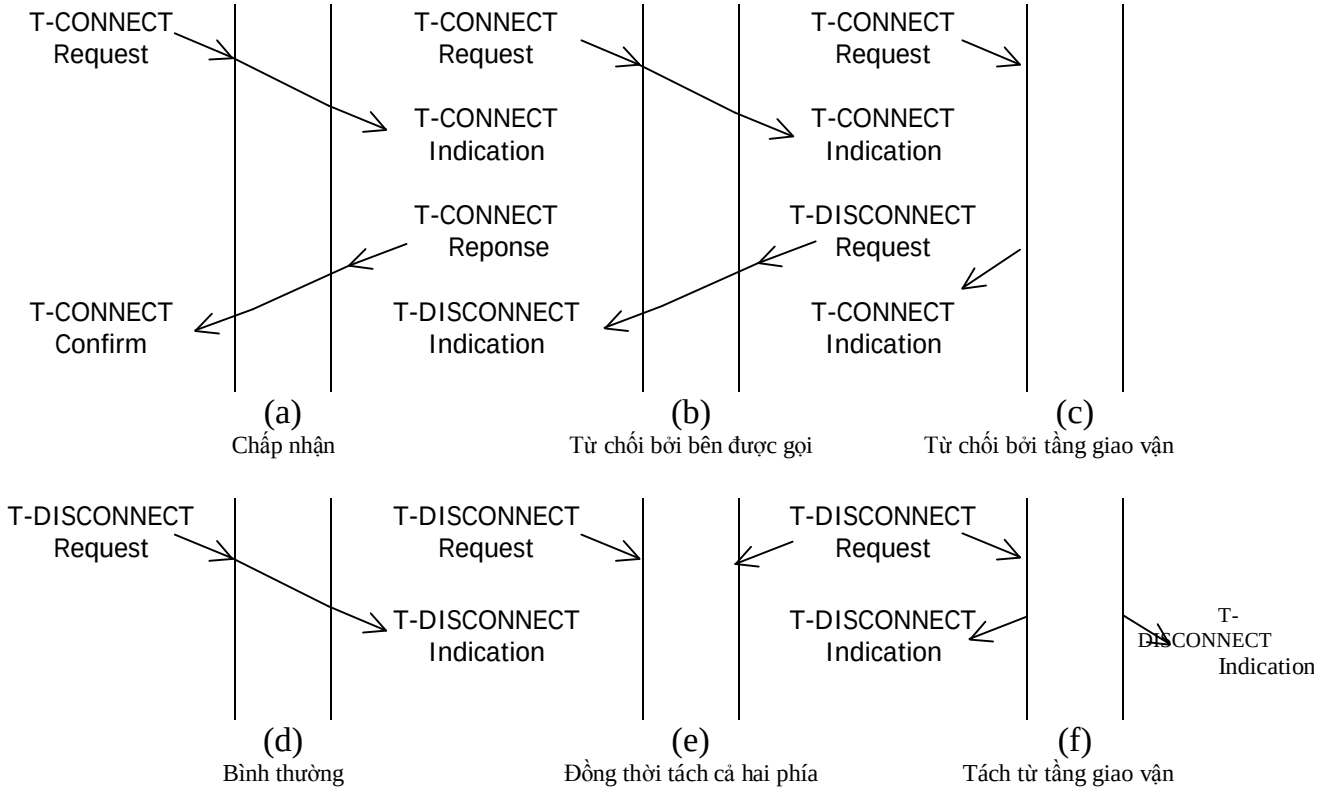
Ngoài phần giao thức chuẩn, ISO còn định nghĩa các dịch vụ mà tầng Giao vận cung cấp cho các thực thể ở tầng Phiên trong trường hợp có liên kết, dưới dạng một tập hợp các hàm dịch vụ nguyên thủy(services primitives) như sau :

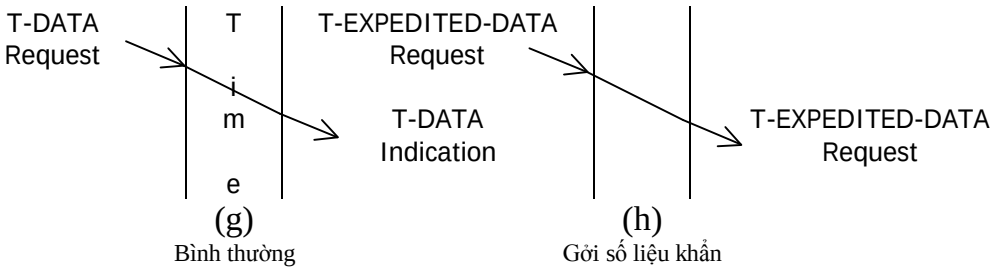
T-CONNECT request (callce, caller, exp wanted, qos, user data)
T-CONNECT indication (callce, caller, exp wanted, qos, user data)
T-CONNECT response (qos, responder, exp wanted, user data)
T-CONNECT confirm (qos, responder, exp wanted, user data)
T-DISCONNECT request (user data)
T-DISCONNECT indication (reason, user data)
T-DATA request (user data)
T-DATA indication (reason, user data)
T-EXPEDITED-DATA request (user data)
T-EXPEDITED-DATA indication (reason, user data)

- Các hàm dịch vụ của tầng giao vận không có kết nối : Chỉ có hai hàm dịch vụ được định nghĩa :

T-UNITDATA request (callce, caller, QoS, user data)
T-UNITDATA indication (callce, caller, QoS, user data)

- Quan hệ giữa các hàm OSI nguyên thủy : Quá trình nối, tách và trao đổi dữ liệu diễn ra như sau :





Hình 6-2. Quan hệ giữa các hàm OSI nguyên thủy.

Giải thích

(a) Quá trình nối được chấp nhận
(b) Quá trình nối bị từ chối bởi bên được gọi
(c) Quá trình nối bị từ chối bởi tầng Giao vận do lỗi của người sử dụng hoặc người chung cấp dịch vụ giao vận gây nên.
(d) Quá trình tách bình thường
(e) Quá trình tách đồng thời cả hai phía
(f) Quá trình tách từ tầng Giao vận
(g) Quá trình trao đổi dữ liệu bình thường
(h) Quá trình trao đổi dữ liệu khẩn

Trong hình (c) trên, việc từ chối có thể do lỗi của người sử dụng hoặc người cung cấp dịch vụ giao vận gây nên. Khi đó, không có gì được phát qua mạng vì vậy đầu kia không nghe được gì cả. Có những qui tắc cho người sử dụng các hàm dịch vụ giao vận. Ví dụ, không được dùng T-DISCONNECT.request khi tiếp nối chưa được thiết lập.

6.1.2 Chất lượng dịch vụ QoS

Chức năng cơ bản của tầng 4 là tăng cường chất lượng dịch vụ được cung cấp bởi tầng 3. Nếu lớp chất lượng chưa tốt, tầng Giao vận sẽ khắc phục khoảng ngăn cách giữa những gì mà người sử dụng tầng Giao vận muốn và những gì mà lớp mạng cung cấp. Các tham số của chất lượng dịch vụ QoS (Quality of Service) bao gồm :

- *Thời gian thiết lập liên kết* là thời gian từ khi gởi yêu cầu tới thời điểm nhận được xác nhận liên kết.
- *Xác nhận không thành công của thiết lập liên kết* - là tỷ lệ yêu cầu liên kết không được chấp nhận trong một thời hạn tối đa.
- *Lưu lượng của liên kết* do số byte hữu ích có thể truyền trong một giây, lưu lượng được tính *trong một cuộc trao đổi* hoặc dựa vào *khả năng của mạng theo 2 chiều.*

- *Thời gian trễ (*Độ trễ truyền dẫn - transmit delay) là khoảng thời gian giữa thời điểm mà người sử dụng dịch vụ của tầng Giao vận bên phát gởi thông báo tời thời điểm thực thể của tầng Giao vận bên thu nhận được. Đánh giá theo 2 chiều.
- *Tỷ lệ lỗi* là tỷ số giữa tin báo bị lỗi (hoặc mất) trên tổng số tin báo được truyền trong một chu kỳ định trước.
- *Xác nhận sự cố truyền:* tỷ số giữa thời gian có sự cố với thời gian cả chu kỳ quan sát.
- *Thời gian hủy liên kết* là thời gian từ khi một người sử dụng phát huy cầu huỷ liên kết đến khi liên kết được huỷ thật sự tại thiết bị đầu cuối từ xa.
- *Xác suất lỗi khi hủy liên kết* là tỷ lệ số yêu cầu hủy liên kết không được thực hiện trong thời gian lớn nhất.
- *Khả năng bảo vệ* là khả năng của người sử dụng cấm thiết bị đầu cuối bên ngoài truy nhập bất hợp pháp hay thay đổi dữ liệu truyền.
- *Thông số ưu tiên:* cho phép người sử dụng có quyền ưu tiên được phục vụ cao hơn đối với một liên kết.
- *Thông số hủy bỏ* cho phép tầng giao vận tự quyết định huỷ liên kết khi có tắc nghẽn hay các vấn đề bên trong mạng.

Người sử dụng khi yêu cầu liên kết sẽ gởi tất cả các thông số với các giá trị yêu cầu tới tầng giao vận và bắt đầu quá trình đàm thoại với các thông số đó.

So sánh các hàm cơ bản của dịch vụ giao vận và dịch vụ mạng, ta thấy các dịch vụ mạng và giao vận gần giống nhau. Sự khác nhau là dịch vụ mạng cho phép người sử dụng xử lý Acknowledgements và N-RQSQTS. Ngược lại, dịch vụ giao vận không quan tâm đến vì dịch vụ lớp giao vận là tin cậy, không có lỗi. Dịch vụ mạng được dùng bởi tầng giao vận.

6.1.3 Các lớp giao thức của tầng giao vận

Các dịch vụ tầng giao vận bảo đảm bằng các giao thức giữa 2 thực thể của tầng cũng tương tự như giao thức của tầng liên kết dữ liệu nó giải quyết vấn đề lỗi, điều khiển lưu lượng và bảo đảm trình tự mảng tin.

tầng liên kết dữ liệu, hai IMP truyền tin trực tiếp qua đường kênh vật lý. ở tầng giao vận, đường kênh vật lý này được thay bằng subnet. Sự khác nhau này kéo theo sự khác nhau về xây dựng các thủ tục. ở tầng giao vận phải xác định địa chỉ nơi nhận, ở tầng liên kết dữ liệu thì không cần vì chỉ có một đường truyền tin giữa hai điểm. Quá trình kết nối ở tầng giao vận cũng phức tạp hơn ở tầng liên kết giữ liệu.

Tầng giao vận đòi hỏi khả năng lưu trữ trong mạng (subnet) để giữ những gói tin bị sự cố và đòi hỏi thủ tục đặc biệt. tầng giao vận số các kết nối lớn hơn nên các vấn đề bộ đệm và điều khiển dòng phức tạp hơn.

Từ quan điểm thiết kế thủ tục giao vận, các dịch vụ được cho bởi mạng quan trọng hơn các tính chất thực tế của mạng, mặc dù cái sau bị ảnh hưởng mạnh bởi cái trước. Tuy vậy, trong một phạm vi nào đó, dịch vụ mức mạng có thể che những mặt ít được chú ý của mạng và cung cấp ghép nối tốt hơn. Để tiện lợi xem xét các thủ tục giao vận, ta chia các dịch vụ trên mạng thành 3 nhóm :

Nhóm	Ý nghĩa
Nhóm A	- Hoàn thiện, tỷ lệ các gói tin bị mất, trùng lặp hoặc bị hỏng không đáng kể. - Lệnh N-RESET có thể bỏ qua. - Tầng giao vận đơn giản, không cần các dịch vụ phục hồi và sắp xếp lại thứ tự gói tin. - Thường là mạng cục bộ.
Nhóm B	- Gói tin bị mất, nhưng kiểm soát được. - Thỉnh thoảng tầng mạng gởi lệnh N-RESET do tắc nghẽn, hỏng phần cứng, vấn đề phần mềm. - Thông thường là mạng đường dài • Giao thức tầng Giao vận có nhiệm vụ: - Thiết lập tại liên kết. Đồng bộ lại - Theo dõi toàn bộ yêu cầu khởi động lại cho NSD.
Nhóm C	- Truyền tin không tin cậy, không liên kết - Mạng đường dài, kết nối nhiều mạng con - Giao thức của tầng giao vận phức tạp, phải có khả năng phục hồi lỗi khi xảy ra sự cố và sắp xếp lại thứ tự các gói tin.

Bảng 6-1. Các nhóm dịch vụ của tầng Giao vận.

Dịch vụ mạng xấu thì giao thức của tầng giao vận sẽ phức tạp hơn. OSI đã nhận thức vấn đề này và chia giao thức của tầng giao vận thành 5 lớp ứng với các loại mạng như sau :

Lớp	Ý nghĩa
Lớp O *Mạng loại A*	- Lớp thủ tục đơn giản - Kết nối mạng khi có yêu cầu giao vận không phải giải quyết lỗi - Chủ yếu tạo ra trình tự, điều khiển dòng dữ liệu để tầng mạng hoạt động tốt. - Bao gồm cơ cấu thiết lập và huỷ liên kết ở tầng giao diện.
Lớp 1 *Mạng loại B*	Có tính chất tương tự lớp 0, ngoài ra còn thêm: - Khởi động lại mạng sau khi N-RESET. Giao thức có khả năng báo nhận (ACK) và truyền dữ liệu khẩn. - Đồng bộ lại và sau đó nối lại liên lạc giữa các thực thể giao vận đã bị gián đoạn - Lớp 1 không kiểm tra lỗi và kiểm soát dòng dữ liệu.
Lớp 2 *Mạng loại A*	Lớp 2 là phiên bản của lớp 0 và được xây dựng cho mạng tin cậy và có thêm một số chức năng như sau : - Sự ghép kênh : Hai hay nhiều liên kết của tầng giao vận có thể dùng chung một kết nối ở tầng mạng. - Sử dụng khi nhiều liên kết ở tầng giao vận được mở đồng thời, nối liên kết có lưu lượng nhỏ. Ví dụ như hệ thống đặt vé máy bay cho phép tiết kiệm đường truyền.
Lớp 3 *Mạng loại B*	Là tổ hợp lớp 1 và lớp 2 - Cho phép dồn kênh - Khởi động lại - Điều khiển dòng dữ liệu.
Lớp 4 *Mạng loại C*	Lớp 4 có hầu hết các chức năng của lớp trước và bổ sung thêm một số khả năng kiểm soát luồng dữ liệu. - Phải có biện pháp giải quyết vấn đề mất gói tin, gói tin bị hỏng - Phải giải quyết yêu cầu khởi động lại - Thủ tục Giao vận phức tạp nhất.

Bảng 6-2. Các lớp dịch vụ của tầng Giao vận.

Dịch vụ không có kết nối đặt tất cả sự phức tạp và thủ tục Giao vận.

6.1.4 Thủ tục giao vận trên X. 25

Thủ tục X. 25 là thủ tục có nối và tin cậy, coi như lớp mạng loại A. Do đó thủ tục giao vận trên X.25 là thủ tục giao vận lớp 0 mô hình OSI. Thủ tục này được thể hiện qua các hàm dịch vụ cơ bản và quá trình nối, tách, trao đổi số liệu của thủ tục.

6.1.4.1 Các hàm dịch vụ cơ bản

Các hàm dịch vụ cơ bản được thực hiện bằng các chương trình con minh họa bằng ngôn ngữ Pascal

1. Hàm Connect thực hiện T-CONNECT .request

   connum = CONNECT(local, remote)

Hàm dịch vụ này để thiết lập kết nối tầng giao vận giữa 2 máy. Nếu kết nối thành công, hàm trả về một số dương, ngược lại hàm trả về số âm.

2. Hàm Listen thực hiện T-CONNECT.indication

   connum = LISTEN (local)

Hàm này dùng để thông báo tiếp nhận yêu cầu kết nối

3. Hàm Disconnect thực hiiện T-DISCONNECT.request

   status = DISCONNECT (commun)

Hàm này dùng để kết thúc kết nối, tham số commun cho biết kết nối nào sẽ bị ngắt, kết quả thực hiện sẽ được gán cho biến status với giá trị OK hoặc error

4. Hàm Send thực hiện T-DATA.request

   status = SEND (commun, buffer, bytes)

Hàm này để phát nội dung ở buffer với kích thước là bytes cho số kết nối đặt ở commun. Kết quả đặt ở status.

5. Hàm Receive thực hiện T-DATA.indication

   status = RECEIVE (commun buffer, bytes)

Hàm này để nhận tin vào buffer với kích thước là giá trị ở biến bytes. Kết quả thực hiện đặt vào status giá trị OK hoặc error.

Chương 7

HỌ GIAO THỨC TCP/IP

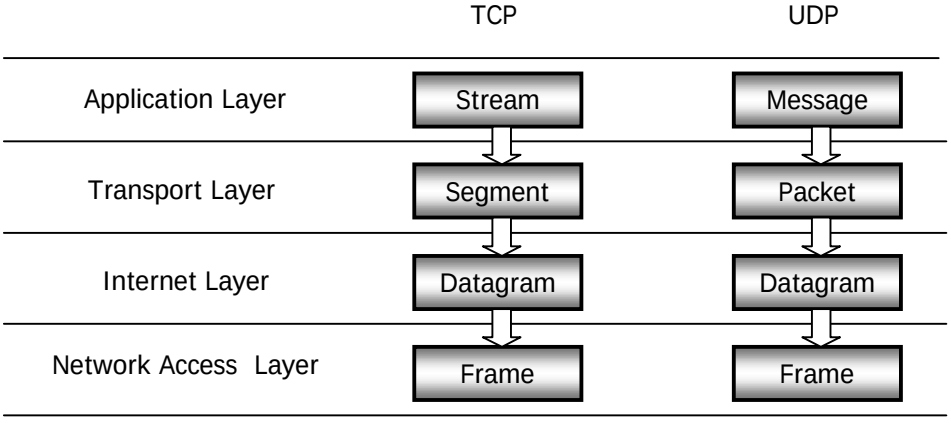
Do đặc tính của mô hình OSI là một mô hình tham chiếu, việc áp dụng mô hình OSI vào thực tế thường có hiệu suất kém do dữ liệu phải truyền qua tất cả các lớp của mô hình OSI ở cả hai máy, mô hình OSI là tiêu chuẩn để các nhà phát triển dựa vào mà phát triển các mô hình khác tối ưu hơn. Có rất nhiều mô hình khác nhau như NetBIOS, IPX/SPX, TCP/IP, tuy nhiên mô hình TCP/IP hiện nay đang được sử dụng phổ biến nhất.

TCP/IP thực chất là một họ giao thức cùng làm việc với nhau để cung cấp ptn truyền thông liên mạng. Mô hình TCP/IP có những tính chất chung như sau :

- TCP /IP độc lập với phần cứng mạng vật lý, điều này cho phép TCP/IP hoạt động trên nhiều mạng khác nhau như Ethernet, Token Ring, X25, dial up,...
- TCP/IP sử dụng sơ đồ đánh địa chỉ toàn cục duy nhất : mỗi máy tính trên mạng TCP/IP có một địa chỉ xác định duy nhất. Mỗi gói tin gởi trên mạng có một tiêu đề chứa địa chỉ nguồn và đích.
- Chuẩn giao thức mở : TCP/IP có thể thực hiện trên bất kỳ phần chứng hay hệ điều hành nào.
- Hoạt động theo mô hình Client/Server.
- Cung cấp các giao thức ứng dụng : cung cấp cho người lập trình phương thức truyền dữ liệu trên mạng giữa các ứng dụng mà còn cung cấp nhiều giao thức ở mức ứng dụng như giao thức truyền nhận mail, truyền file, . . .
- TCP/IP hỗ trợ cho liên mạng (internetworking) và định tuyến, các giao thức mức cao được chuẩn hoá thích hợp và cung cấp sẵn các dịch vụ người dùng.

7.1 Mô hình TCP/IP

Cấu trúc của bộ giao thức TCP/IP có bốn tầng, được mô tả như hình vẽ sau



Hình 7-1. Kiến trúc TCP/IP và các đơn vị dữ liệu.

Chức năng của các tầng như sau :

1. Tầng truy cập mạng NAL (Network Access Layer)

- Cung cấp cho hệ thống phương thức để truyền dữ liệu trên các thiết bị phần cứng vật lý khác nhau của mạng.
- Đóng gói các lược đồ dữ liệu IP (IP datagram) vào các *frame* truyền trên mạng và việc ánh xạ các địa chỉ IP thành các địa chỉ vật lý tương ứng dùng cho mạng trước khi truyền xuống kênh vật lý.
- Định nghĩa cách thức truyền các khối dữ liệu IP : Các giao thức ở lớp này phải biết chi tiết các phần cấu trúc vật lý mạng ở dưới nó để định dạng chính xác các dữ liệu sẽ được truyền phụ thuộc vào từng loại mạng vật lý cụ thể.

Lớp truy cập mạng NAL của mô hình kiến trúc TCP/IP tương đương với ba lớp thấp nhất của mô hình OSI là Network layer, Datalink layer, và Physical layer.

2. Tầng mạng

Tầng mạng chịu trách nhiệm định tuyến các thông báo (message) qua các mạng vật lý khác nhau, liên mạng, giao thức ở lớp này là IP là giao thức quan trọng nhất vì IP cung cấp dịch vụ giao nhận gói tin cơ bản trên các mạng TCP/IP, mọi giao thức ở các lớp trên và bên dưới tầng mạng đều sử dụng giao thức IP để thực hiện việc giao nhận dữ liệu. Hơn nữa IP bổ sung một hệ thống địa chỉ logic được gọi là địa chỉ IP, được sử dụng bởi lớp Internet và các lớp cao hơn để nhận diện các thiết bị và thực hiện định tuyến liên mạng.

3. Tầng Giao vận (Host to Host Transport Layer)

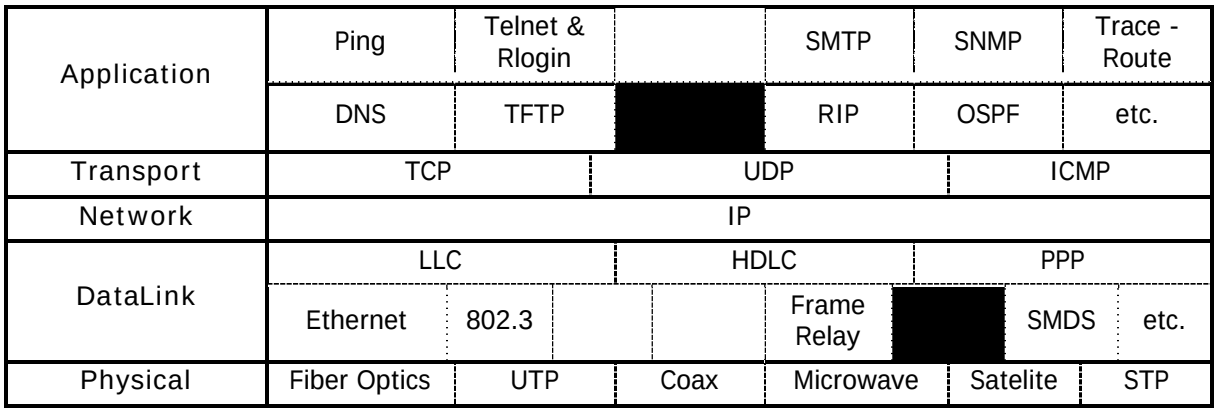
- Cung cấp phương tiện liên lạc từ một chương trình ứng dụng này đến chương trình ứng dụng khác, chịu trách nhiệm đảm bảo toàn vẹn dữ liệu đầu cuối.
- Trong lớp này có 2 giao thức quan trọng nhất:

- Transmission Control Protocol (TCP) : Về chức năng TCP tương đương với lớp giao thức đầy đủ nhất của giao thức chuẩn Transport của OSI. Tuy nhiên, khác với mô hình ISO, TCP sử dụng phương thức trao đổi các dòng dữ liệu (data stream) giữa người sử dụng.
- User Datagram Protocol (UDP) : cung cấp dịch vụ giao nhận dữ liệu theo kiểu “không liên kết” (connectionless), không cần phải thực hiện thiết lập liên kết logic giữa một cặp thực thể UDP trước khi chúng trao đổi dữ liệu với nhau.

4. Tầng ứng dụng (Application Layer)

Bao gồm tất cả các tiến trình sử dụng các giao thức của lớp Transport để truyền dữ liệu. Có nhiều giao thức ứng dụng ở lớp này, phần lớn là nhằm cung cấp cho người dùng các dịch vụ ứng dụng, sử dụng 2 giao thức chính TCP và UDP.

Tầng ứng dụng cung cấp các dịch vụ trên Internet như thư điện tử (SMTP), truyền file (FTP), v.v.. Tầng dưới là phần mạng để định tuyến địa chỉ đến.



Hình 7-2. Họ giao thức TCP/IP.

Telnet	Tele Comunication	Dịch vụ truy cập từ xa.
FTP	File Transfer Protocol	Dịch vụ truyền File.
SMTP	Simple Mail Transfer Protocol	Dịch vụ truyền thư đơn giản.
DNS	Domain Name System	Hệ thống tên miền
SNMP	Simple Network Management Protocol	Giao thức quản lý mạng đơn giản
RPC	Remote Procedure Call	Thủ tục gọi từ xa
RIP	Routing Information Protocol	Giao thức định tuyến thông tin
TCP	Transmission Control Protocol	Giao thức TCP
UDP	User Datagram Protocol	Giao thức dữ liệu của người dùng.
IP	Internet Protocol	Giao thức IP
ICMP	Internet Control Message Protocol	G.thức kiểm soát message giữa các mạng.
FDDI	Fiber Distributed Data Multiplexing	

7.2 Giao thức TCP

Tầng Giao vận sử dụng hai giao thức chính là TCP và UDP. Giao thức TCP (Transmission Control Protocol) đảm bảo độ tin cậy giữa nơi gởi và nơi nhận (end-to-end) trong điều kiện lớp mạng loại C không tin cậy. Dòng số liệu có chiều dài tuỳ ý được phân thành những đoạn không vượt quá 64KB, gửi đi đến đầu bên kia lại được gộp lại thành bản tin ban đầu.

- Chức năng của giao thức TCP :

Chức năng	Giải thích
Phát hiện lỗi	Bằng cách sử dụng một trường checksum để kiểm tra lỗi bất cứ khi nào datagram được cắt ra trong quá trình truyền.
Truyền lại	TCP sẽ truyền lại các gói tin bị mất hoặc bị sai hỏng trong quá trình truyền.
Đánh số thứ tự	Cho phép bên gởi đã phát đi các gói tin theo một trật tự, bên nhận đã nhận và kết hợp các gói tin theo một trật tự đã định
Báo nhận và kiểm soát luồng	Bên TCP nhận sẽ gửi một đoạn báo nhận xác định một số chức năng trong quá trình truyền tin.
Phát gói tin đến đúng ứng dụng yêu cầu	Mỗi đoạn gói tin TCP có một số hiệu cổng nguồn và đích, là giá trị duy nhất để xác định một phiên làm việc.

- Tính chất của giao thức TCP :

Tính chất	Giải thích
Tin cậy	TCP cung cấp khả năng tin cậy bằng cách gởi lại dữ liệu đến khi bên nhận có một báo nhận hỏng. Đơn vị dữ liệu mà TCP truyền đi là segment và được giao thức IP phân ra thành các datagram.
Hướng kết nối	TCP thiết lập kết nối logic giữa các máy khi truyền dữ liệu, hoạt động theo cơ chế "bắt tay" (handshake), và có nhiệm vụ đồng bộ việc kết nối giữa hai máy.
Dòng dữ liệu	TCP xử lý dữ liệu dưới dạng một dòng nối tiếp các byte, theo cơ chế đánh số thứ tự gói tin.

7.2.1 Khuôn dạng gói tin TCP

TCP là một giao thức có liên kết (*connection - oriented*) nghĩa là cần phải thiết lập liên kết logic giữa một cặp thực thể TCP trước khi chúng trao đổi dữ liệu với nhau, có 3 giai đoạn : ***thiết lập liên kết, truyền tải dữ liệu*** và ***hủy liên kết***. Đơn vị dữ liệu của TCP được gọi là **segment** (đoạn dữ liệu). Cấu trúc đơn vị dữ liệu của TCP được mô tả như hình sau :

Source Port - Số hiệu cổng nguồn (16 bits)

Xác định số hiệu cổng của trạm nguồn - User TCP cục bộ (thường là một chương trình ứng dụng trên lớp cao hơn).

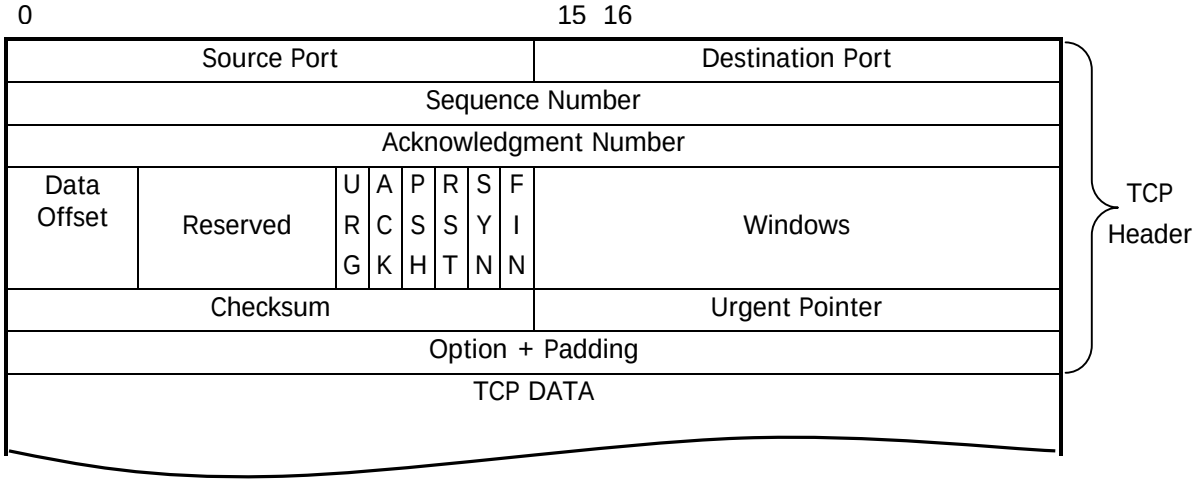
Destination Port - Số hiệu cổng đích (16 bits)

Xác định số hiệu cổng của trạm đích của máy ở xa. Dùng để nhận diện các tiến trình điểm đầu mút ở kênh ảo TCP.

Sequence Number - Số thứ thự (32 bits)

Trường này chứa một số chỉ vị trí hiện tại của khối tin trong Message. Số này cũng được các phiên bản khác nhau của TCP để cung cấp số thứ tự của khối tin ban đầu (ISN).

Đây là số hiệu byte đầu tiên của segment trừ khi bit SYN được thiết lập. Nếu bit SYN được thiết lập thì Sequence Number là số hiệu tuần tự khởi đầu (ISN) và byte dữ liệu đầu tiên là ISN+1.



Hình 7-3. Cấu trúc của gói tin TCP.

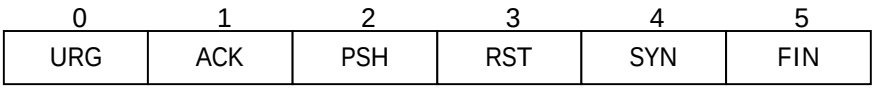
Acknowledgment Number - Số phúc đáp (32 bits)

Dùng để chỉ ra số hiệu của segment (khối tin) sắp được truyền tiếp theo mà trạm đích đang chờ để nhận. Dùng báo nhận tốt các Segment mà trạm nguồn đã gửi cho trạm đích. Ngoài ra nó cũng chỉ ra số thứ tự của khối tin nhận được sau cùng; nó chỉ ra số thứ tự của khối tin nhận được cộng thêm 1.

Data offset (32 bits) : Trường này dùng để chỉ ra vị trí bắt đầu của trường dữ liệu.

Reserved (6 bits) : Chưa dùng đến, dành sử dụng về sau. Các bit được đặt bằng 0.

Control Bits - Các bit điều khiển



Cờ URG : Nếu có giá trị là 1 thì thì trường urgent pointer rất quan trọng.
Cờ ACK : Nếu có giá trị là 1 thì trường Acknowledgment rất quan trọng.
Cờ PSH : Nếu thiết lập thì tức là chức năng PUSH sắp được thực hiện.
Cờ RST : Nếu được thiết lập thì kết nối hiện tại sắp được khởi tạo lại.
Cờ SYN : Chỉ ra số thứ tự của đoạn tin sẽ được đồng bộ hoá. Cờ này được dùng khi mà kết nối được thiết lập.
Cờ FIN : Nếu cờ này thiết lập, nó chỉ ra rằng phía gửi không còn dữ liệu để gửi nữa. Điều này tương đương với việc đánh dấu kết thúc quá trình truyền.

Window - Cửa sổ (16 bits)

Trường này cấp phát thẻ dùng để kiểm soát luồng dữ liệu theo cơ chế cửa sổ. Đây là số lượng các byte dữ liệu khối tin mà phía thu có thể chấp nhận được.

Checksum (16 bits)

Chứa mã kiểm soát lỗi (theo phương pháp CRC) cho toàn bộ segment.

Urgent Pointer - Con trỏ khẩn (16 bits)

Trường này được dùng khi mà cờ URG được thiết lập; con trỏ này trỏ tới số hiệu tuần tự của các byte đi theo sau dữ liệu khẩn, cho phép bên nhận biết được độ dài của dữ liệu khẩn.

Options (có độ dài thay đổi)

Trường này dùng để xác định các Option của TCP. Mỗi lựa chọn bao gồm một số (1 byte) để chỉ ra lựa chọn đó, một số chỉ giá trị của các byte trong trường Option, và các giá trị lựa chọn. Hiện nay với TCP mới có 3 Option được định nghĩa, như sau:

Số 0 : Cuối danh sách các lựa chọn
Số 1 : Không hoạt động (*No Operation*)
Số 2 : Kích cỡ lớn nhất của một Segment

Trường Options chỉ để xác định kích thước lớn nhất của bộ đệm mà TCP nhận có thể chấp nhận được. Bởi vì TCP dùng trường dữ liệu có chiều dài thay đổi được nên có thể có trường hợp là máy gửi sẽ tạo ra một đoạn tin mà phía nhận không thể chấp nhận được.

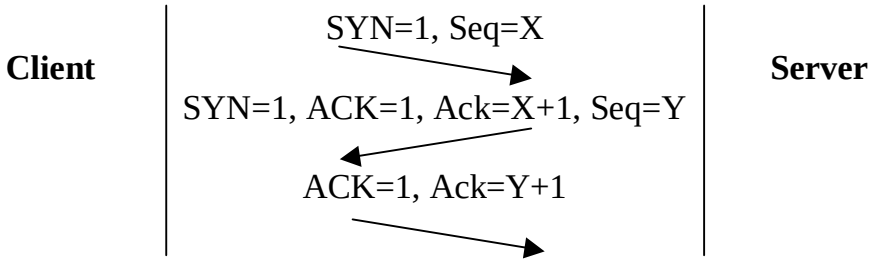
Padding :

Dùng để bổ sung vào Header để bảo đảm rằng phần Header luôn là bội số của 32 bit. Phần thêm vào bao gồm toàn số 0.

TCP Data (Có độ dài thay đổi)

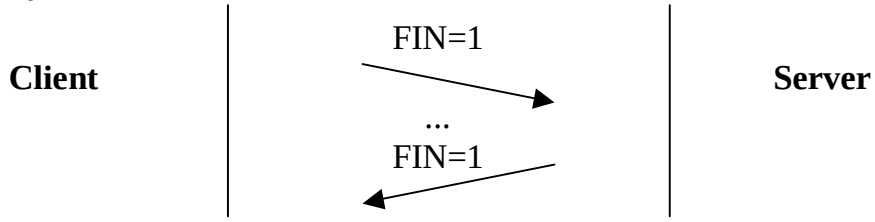
Chứa dữ liệu của tầng trên, độ dài tối đa ngầm định là 536 bytes. Giá trị có thể điều chỉnh bằng cách khai báo trong vùng Options.

7.2.2 Quá trình nối-tách



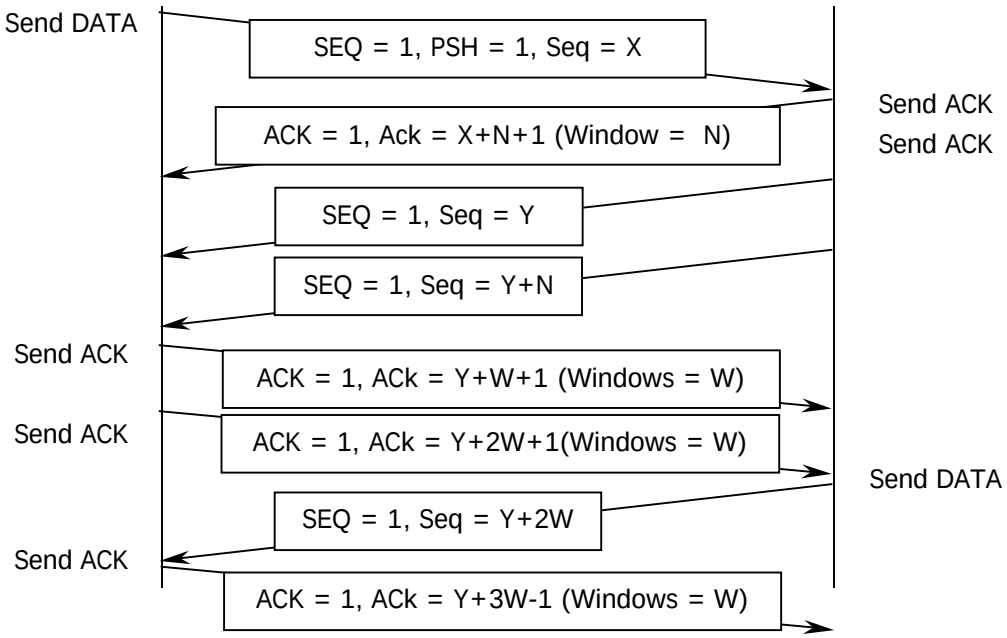
Quá trình thiết lập kết nối bằng thủ tục bắt tay 3 lần (three-way hand). Client gửi bản tin với SYN=1 (yêu cầu kết nối). Server nhận được, gửi bản tin với SYN=1 và ACK=1. Client lại đáp lại với bản tin ACK=1.

Kết thúc kết nối bằng thủ tục bắt tay hai lần (two-way hand). Bên kết thúc gởi số liệu, gởi bản tin với FIN=1, TCP cho phép nhận tiếp tục số liệu cho đến khi bên kia gửi bản tin FIN=1



Ngoài ra, thủ tục TCP/IP còn dùng để kết nối giữa LAN và WAN như một thủ tục cho mạng LAN.

7.2.3 Quá trình trao đổi dữ liệu



H×nh 7-4. S¬ ®å qu¸ tr×nh trao ®æi d÷ liÖu cña TCP.

W= maximun Segment size (W>N)

2W = Windows limit

7.2.4 Thứ tự thực hiện ứng dụng TCP/IP

Sự kết hợp của thủ tục TCP và IP thực sự là sự kết hợp giữa các mạng máy tính nối với nhau cho phép người dùng các mạng máy tính nối với nhau cho phép người dùng các mạng khác nhau liên lạc và làm việc được với nhau.

Thủ tục TCP là thủ tục tại đầu cuối, còn IP dùng để chạy trên mạng. Khi người sử dụng thủ tục TCP tạo được phân đoạn TCP và kết hợp vào IP để tạo thành IP datagram. Router căn cứ vào địa chỉ IP trong gói tin và thông tin chứa trong bảng định tuyến để chuyển gói này đi tới các router sau. Khi gói tin IP đến router cuối cùng, router này tìm và chuyển gói tin đến địa chỉ hệ thống đầu cuối.

Nếu IP datagram không chuyển tới đầu cuối được vì một lý do nào đó, nó sẽ bị hủy bỏ và giao thức IP không còn thông báo được điều này cho người sử dụng biết. Giao thức TCP cung cấp mối liên hệ tin cậy giữa các đầu cuối, đảm bảo dữ liệu phát đi đúng địa chỉ, không bị thiếu hay phát lặp nghĩa là tại điểm cuối cùng thủ tục TCP sẽ đọc số thứ tự trong phân đoạn TCP để biết gói tin bị thiếu hay gói đã nhận rồi và báo lại cho bên phát biết.

Gói tin IP không phụ thuộc vào các giao thức cụ thể của các mạng khác nhau mà nó đi qua (X.25 hay Frame relay v.v..). Với IP các mạng chỉ đơn thuần là đường dẫn các Router. Ta có thể hình dung IP datagram như một phong bì bình thường, người gửi thư không quan tâm đến bức thư đến được người nhận bằng ô tô, tàu hỏa hay máy bay.

Sự kết hợp giữa thủ tục TCP và IP giúp người dùng sử dụng được các dịch vụ trao đổi trên Internet thực hiện qua các bước chính sau đây:

Bước 1: Các dữ liệu ứng dụng kết hợp với số thứ tự để hình thành phân đoạn TCP.

Người sử dụng dùng dịch vụ trên mạng như thư điện tử, Telnet hay FTP v.v.. có nghĩa là đưa các dữ liệu của người dùng vào phần dữ liệu của gói tin TCP. Giao thức TCP sẽ đưa vào phần header của gói tin các thông tin sau:

- Số hiệu cổng quy định của Internet.
- Số thứ tự Segment gửi đi.
- Thông báo cho bên gửi biết đã nhận được Segment thứ mấy (ACK)
- Số byte cần phát.

Bước 2: Kiến tạo ra gói tin IP datagram

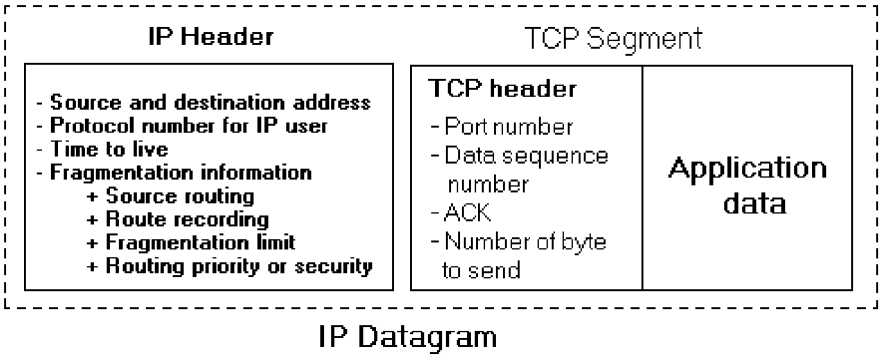
Trên cơ sở của gói tin TCP, IP thêm các thông tin sau đây vào để tạo thành IP Datagram.

- Địa chỉ phát và nhận : Router sử dụng địa chỉ này để định tuyến.
- Số thủ tục (Protocol number): định nghĩa thủ tục mà IP thực hiện.
- Thời gian tồn tại (Time to live): định nghĩa số Router bắt buộc Datagram phải đi qua trước khi nó bị hủy bỏ.
- Thông tin về các phân đoạn bị chia nhỏ trong quá trình chuyển đi trên mạng.

Kích thước gói tin thay đổi tùy thuộc vào mạng khác nhau, chẳng hạn như kích thước gói tin trong mạng Ethernet là 1500 bytes còn mạng X.25 chỉ có 128 bytes.

- Các thông tin tùy chọn

- Source Routing (Định tuyến bên phát): cung cấp danh sách các Router sử dụng.
- Route recording (Ghi lại tuyến đường đã đi qua): Thông tin sẽ yêu cầu mỗi Router ghi lại địa chỉ IP khi nó chuyển datagram qua, dùng để thống kê được số liệu của đường dẫn trong Internet.
- Fragmentation limit (Giới hạn phân mảnh): Định nghĩa cỡ lớn nhất (tính theo byte) của một datagram có thể chuyển đi mà không cần phải chia nhỏ.
- Routing priority or security (Ưu tiên hoặc bảo đảm an toàn cho Datagram): chỉ rõ tuyến nào dành ưu tiên hay tuyến nào bảo đảm được an toàn cho datagram.



Hình 7-5. Cấu trúc của IP Datagram.

Như vậy Datagram thực chất là hình thức một gói tin chứa dữ liệu thông tin được dùng trong internet.

Bước 4: Chuyển gói đến địa chỉ đích

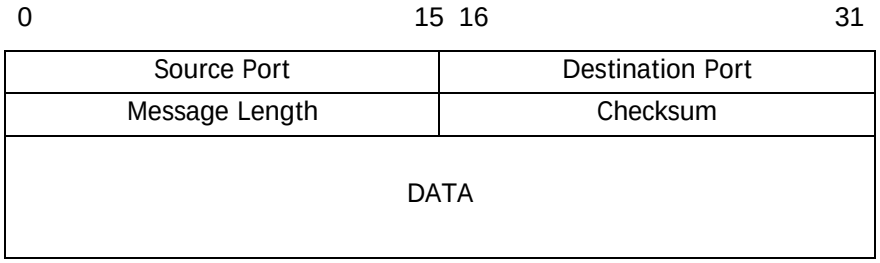
Các IP Datagram chuyển qua các lớp dưới đưa vào và định tuyến để tìm tới địa chỉ đến qua mạng căn cứ vào địa chỉ vật lý của mạng lưới ví dụ như địa chỉ mạng X.25, mạng Frame relay hoặc ngay bản thân của Internet. Tất cả các thông tin này đều nằm trong bảng định tuyến trong các Router. Các mạng X.25 hay Frame relay chỉ làm nhiệm vụ chuyển tải các Datagram.

Tại phía đầu cuối thu, TCP tách IP datagram để lấy phân đoạn TCP xử lý dữ liệu thông tin, đối chiếu số thứ tự, phát hiện những gói thiếu thiếu hay đã nhận được, đồng thời cũng nhận được thông báo (ACK) từ phía phát báo cho biết bên đấy đã nhận được gói thứ mấy do bên này phát đi.

Phía thu thông báo (ACK) cho bên phát biết số dữ liệu đã nhận được đồng thời cũng yêu cầu phát lại những gói tin thiếu nếu có.

7.3 Giao thức UDP

Giao thức UDP (User Datagram Protocol) cho phép người sử dụng gởi bản tin mà không cần thiết lập liên kết, do đó không bảo đảm việc giao nhận chính xác hoặc thứ tự bản tin. Giao thức UDP dùng cho dịch vụ không tin cậy 100%. Thực tế trong các mạng 99% bản tin UDP được giao nhận đúng đích. Do ít chức năng phức tạp nên UDP hoạt động nhanh hơn so với TCP.



Hình 7-6. Khuôn dạng của UDP Datagram.

Các trường có ý nghĩa như sau:

- *Source Port* - Số hiệu cổng nguồn (của máy gửi): Một trường có thể lựa chọn được với số hiệu cổng. Nếu một số hiệu cổng không xác định thì trường này có giá trị là 0.
- *Destination Port* - Số hiệu cổng trên máy nhận.
- *Message Length* - Chiều dài của dữ liệu trong đó cả phần Header và dũ liệu.
- *Trường Checksum*: là 16 bit bù một của phép tổng bù một của trường dữ liệu, có cả phần pseudoHeader giống như của TCP.

Trường checksum của UDP cũng có thể lựa chọn được, nhưng không được dùng. Không một checksum nào được dùng cho phần dữ liệu vì phần checksum của IP chỉ dùng cho phần Header IP mà thôi. Nếu phần checksum không được dùng thì các bit của trường này được thiết lập là 0.

Giao thức UDP được sử dụng trong một số tình huống đặc biệt :

- Khi truyền một dữ liệu nhỏ thì dùng UDP có hiệu quả hơn so với việc kết nối và hủy kết nối khi sử dụng TCP.
- Các ứng dụng hỏi đáp, mong muốn trả lời trong một thời gian ngắn sau khi người sử dụng gởi đi yêu cầu. Trả lời cũng là một cơ chế báo nhận. Người ta sử dụng giao thức UDP như trong các dịch vụ ứng dụng không yêu cầu độ chính xác cao như thông báo giờ hay các dịch vụ gởi nhắn tin, tỷ giá .. .

- Một số mô hình nén để truyền các thông tin audio, video, có thể chấp nhận được một vài gói dữ liệu bị hỏng hay thất lạc.
- Một vài ứng dụng có đô tin cậy riêng trong khi truyền dữ liệu thì nên dùng UDP hơn là TCP.

7.4 Cổng và Socket

7.4.1 Số hiệu cổng

Khi một máy khách kết nối vào máy chủ thì có thể yêu cầu nhiều dịch vụ khác nhau trên máy chủ. Mỗi dịch vụ đều có cách gởi và nhận dữ liệu theo quy ước riêng. TCP và UDP chỉ chịu trách nhiệm đưa dữ liệu từ một máy tính náy đến một máy tính khác, còn dữ liệu đó được gởi đến dịch vụ theo cách nào thì phải thông qua cổng của dịch vụ.

Cổng được đặc trưng bởi một số có giá trị từ 0 đến 65535. Các cổng chuẩn từ 0 - 1023 là cổng được dùng cho các dịch vụ phổ biến như FTP, eMAIL, POP3, HTTP, ... Không thể có hai tiến trình cùng sử dụng chung một số hiệu cổng.

Các số hiệu cổng (Port Numbers) được dùng thông dụng trong thực tế :

UDP Port		TCP Port	
0	Reversed	0	Reversed
7	Echo	1	TCP Multiplexor
37	Time	20	FTP_ Data Connection
42	Name Server	21	FTP_ Command Connection
53	Domain Name Server	23	TELNET
69	Trivial File Transfer Program (TFTP)	25	SMTP
514	System Log	42	Name Server
. . . . .		53	Domain Name Server
		79	Finger_ find a active user
		80	HTTP

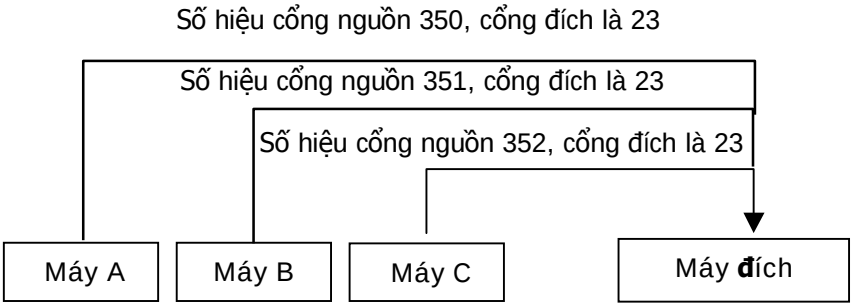
7.4.2 Socket

Mỗi socket xác định một điểm cuối trong liên kết truyền thông hai chiều giữa các tiến trình giao tiếp trên mạng, là đối tượng mà qua đó các dịch vụ ứng dụng truyền hoặc nhận các gói dữ liệu trên mạng. Khi cần gởi dữ liệu đi, các tiến trình ghi dữ liệu vào socket, khi có dữ liệu đến, các tiến trình sẽ đọc socket để lấy dữ liệu.

Trong những năm 80, do nhu cầu cần có một giao diện lập trình ứng dụng API (Application Programming Interface) để phát triển các trình ứng dụng trên mạng TCP/IP, giao diện socket đã được xây dựng lần đầu tiên trên hệ điều hành UNIX. Loại Berkelay Socket (Berkeley Software Distribution - BSD, tại Trường Đại học

- Loại socket : Stream socket hoặc Datagram socket.

Một liên kết giữa hai máy trên với nhau được xác định bởi một cặp socket : Socket (Host1, Port1) và Socket (Host2, Port2). Số **Socket** là duy nhất cho phép một tiến trình có thể giao tiếp với một tiến trình khác trên mạng.



Hình 7-7. Nhiều máy nguồn nối với một máy đích.

Một liên kết có thể được thiết lập theo một trong hai cách : chủ động (active) hoặc bị động. Các thực thể tầng trên sử dụng TCP thông qua bằng cách gọi các hàm dịch vụ nguyên thủy. Dịch vụ TCP được thiết lập nhờ một liên kết logic giữa một cặp Socket. Một Socket có thể tham gia nhiều liên kết với các Socket ở xa khác nhau. Vì các khung tin được đưa qua cổng đều có đầy đủ các thông tin về socket (với địa chỉ IP), cho nên không có xung đột dữ liệu xảy ra.

7.5 Mô hình giao tiếp Client/Server

TCP/IP phụ thuộc vào khái niệm máy khách (Client) và máy chủ (Server). Thuật ngữ Server dùng để chỉ những chương trình cung cấp các dịch vụ thông qua mạng. Các Server nhận đảm nhiệm chức năng đáp ứng các yêu cầu của máy khách, thực hiện việc phục vụ và trả lại kết quả. Thuật ngữ Client dùng để chỉ các chương trình ứng dụng gởi các yêu cầu đến Server và chờ kết quả trả về.

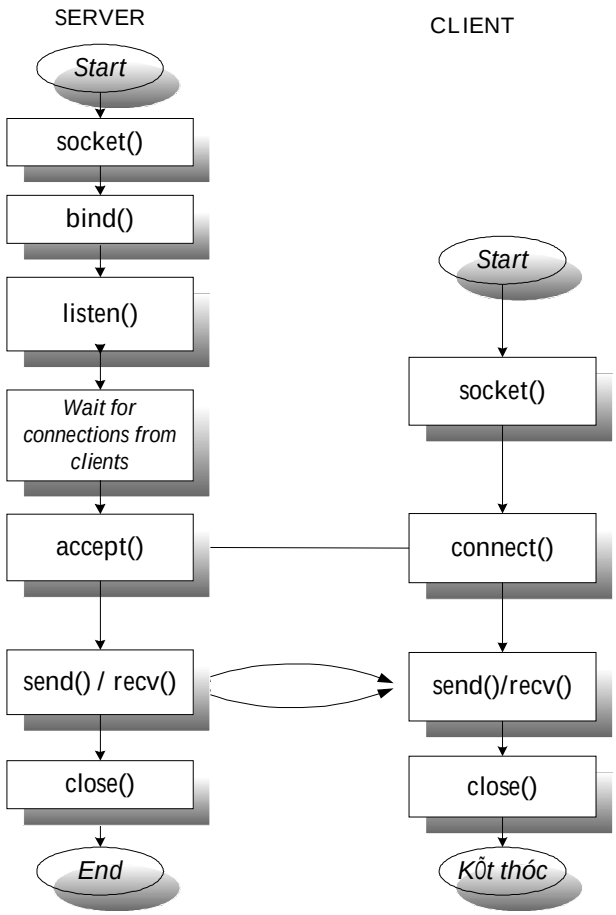
Các chương trình Client và Server thường thực thi trên các máy khác nhau. Mỗi chương trình Server có thể cùng đáp ứng cho nhiều chương trình Client trên nhiều máy tính khác nhau cùng một lúc.

7.5.1 Quá trình trao đổi dữ liệu dùng Stream Socket

Stream socket dựa trên nền giao thức TCP đòi hỏi phải tạo một kết nối trước khi hai bên có thể truyền hoặc nhận dữ liệu cho nhau. Stream Socket cung cấp một dòng các byte dữ liệu không có phân cách có thể truyền hai chiều. Các dòng dữ liệu có thể tin cậy được phân phát tuần tự, dữ liệu không trùng lắp, nghĩa là các gói dữ liệu được phân phát theo thứ tự được phát, và mỗi lần chỉ có một gói riêng biệt được truyền.

Dạng socket này rất thích hợp với mô hình Client/Server. Server sẽ tạo một socket, gán cho nó một tên (cung cấp một địa IP của máy và một port để giao tiếp), và đợi client nối kết đến socket. Bên client cũng tạo một socket và nối kết đến tên socket trên server. Khi server phát hiện có yêu cầu kết nối từ client, nó sẽ tạo một socket mới và sử dụng socket mới đó để giao tiếp với client. Socket cũ tiếp tục đợi kết nối từ các client khác.

Sơ đồ trao đổi dữ liệu giữa Client/Server bằng cách dùng Socket được biểu diễn như sau :



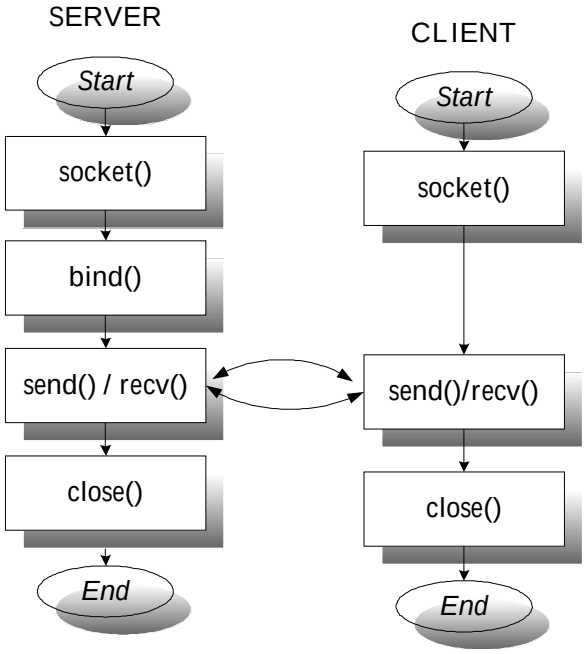
Hình 7-8. Sơ đồ trao đổi dữ liệu giữa Client/Server bằng StreamSocket.

7.5.2 Quá trình trao đổi dữ liệu dùng Datagram Socket

Datagram Socket dựa trên giao thức UDP không đòi hỏi phải thiết lập một kết nối trước khi truyền và nhận dữ liệu. Dữ liệu chỉ là một gói đơn, vì vậy dạng socket này thường dùng để truyền các mẫu tin, không cần nhiều các header lớp ứng dụng. Dạng socket này cung cấp luồng dữ liệu không bảo đảm theo thứ tự hoặc không bị trùng lắp, không bảo đảm dữ liệu sẽ đến được nơi nhận. Dữ liệu có thể đến không theo thứ tự được phát và có khả năng bị trùng lặp. Nhưng sự phân cách giữa các

mẫu tin thì được duy trì. Trong mạng LAN datagram có khả năng tin cậy tương đối tốt, nhưng trong mạng WAN, như mạng Internet thì không được đảm bảo.

§ Lưu đồ client/server sử dụng giao thức UDP



Hình 7-9. Sơ đồ trao đổi dữ liệu giữa Client/Server bằng DatagramSocket.

7.5.3 Ví dụ chương trình client/server

Trong ví dụ dưới đây chương trình server thực hiện các bước thiết lập cho việc chờ đợi một kết nối từ chương trình client. Sau khi thiết lập kết nối với client, cả hai thực hiện một số thao tác truyền và nhận thông tin rồi kết thúc chương trình.

7.5.3.1 Mã lệnh chương trình Server

- Tạo ra một socket với hàm *socket()*.
- Ràng buộc socket với một địa chỉ bằng hàm *bind ()*.
- Dùng hàm *listen()* để chờ đợi một kết nối.
- Nhận bất kỳ thông tin nào yêu cầu kết nối bằng hàm *accept()*.
- Nhận các thông báo gửi đến bằng hàm *read()* và gửi thông báo đến client bằng hàm *write()*.
-

```
/* mksock.c make and bind to a socket - userver*/
#include<stdio.h>
#include<sys/socket.h>
#include<sys/un.h>
#include<unistd.h>

void die(char * message);
void copyData(int from, int to);
```

```
int main(void) {
  struct sockaddr_un address;
  int sock,conn;
  size_t addrLength;
  if ((sock=socket(PF_UNIX,SOCK_STREAM,0))<0)
   die("socket");
/*unlik("./sample_socket");*/
  address.sun_family=AF_UNIX;
  strcpy(address.sun_path,"./sample_socket");

addrLength=sizeof(address.sun_family)+strlen(address.sun_path);
if(bind(sock,(struct sockaddr *)&address,addrLength))
  die("bind");
if(listen(sock,5))
  die("listen");
while((conn=accept(sock,(struct sockaddr*)&address,&addrLength))>=0) {
  printf("---getting data\n");
  copyData(conn,1);
  printf("---done\n");
  close(conn);
}
if (conn<0) die("accept");
close(sock);
return 0;
}
void die(char * message){
  perror(message);
  exit(1);
}
void copyData(int from,int to){
  char buf[1024];
  int amount;
  while ((amount=read(from,buf,sizeof(buf)))>0){
          if(write(to,buf,amount)!=amount){
                  die ("write");
                  return;
  }
}
if (amount<0) die("read");
}
```

7.5.3.2 Mã lệnh chương trình client

Từ chương trình client , để thực hiện được một kết nối đến server và truyền nhận thông tin chỉ cần thực hiện 2 bước cơ bản như sau:

- Tạo một *socket()* tương ứng với chương trình *server* cụ thể .
- Yêu cầu đến server thực hiện kết nối bằng cách gọi hàm *connect()*.

Nếu một kết nối được tạo ra, client có thể gửi yêu cầu bằng hàm *write()* và nhận các đáp ứng phản hồi bằng hàm *read()*.

```
/* sockconn.c - connect to a socket - uclient*/
#include<sys/socket.h>
#include<sys/un.h>
#include<unistd.h>

void die (char * message);
```

```
void copyData(int from, int to);

int main(void){
  struct sockaddr_un address;
  int sock;
  size_t addrLength;

  if ((sock=socket(PF_UNIX,SOCK_STREAM,0))<0)        die("socket");
  address.sun_family=AF_UNIX;
  strcpy(address.sun_path,"./sample_socket");

  addrLength=sizeof (address.sun_family) + strlen(address.sun_path);
  if(connect(sock,(struct sockaddr *)& address,addrLength))   die("connect");
  copyData(0,sock);
  close(sock);
  return 0;
}
void die(char * message){
  perror(message);
  exit(1);
}
void copyData(int from, int to){
  char buf[1024];
  int amount;
  while ((amount=read(from,buf,sizeof(buf)))>0){
    if(write(to,buf,amount)!=amount) {
                    die("write");
                    return;
    }
  }
if (amount<0)      die("read");
}
```

BÀI TẬP

1. Tìm hiểu các mô tả Socket và cấu trúc dữ liệu của socket mà hệ điều hành cấp phát để lưu trữ các thông tin cần thiết cho kết nối mạng.
2. Tìm hiểu các thư viện lập trình WinSock trên hệ điều hành Windows.
3. Viết các chương trình giao tiếp Client/Server theo mô hình giao tiếp TCP/IP hoặc UDP/IP.

Chương 8

TẦNG PHIÊN

Tầng phiên (Session Layer) làm nhiệm vụ tổ chức và đồng bộ sự chuyển đổi dữ liệu giữa các tiến trình ứng dụng khác nhau. Tầng Phiên làm việc với tầng ứng dụng để cung cấp các tập dữ liệu, được gọi là các điểm đồng bộ, các điểm này cho phép một ứng dụng biết quá trình truyền và nhận dữ liệu được thực hiện như thế nào.

Tầng phiên chịu trách nhiệm thiết lập và duy trì một phiên truyền thông giữa hai trạm hoặc nút mạng. Một phiên truyền thông qua một mạng hoạt động có phần giống với một cuộc gọi qua các đường dây điện thoại. Tầng Phiên cố gắng thiết lập một phiên truyền thông giữa hai nút trên một mạng. Cả hai nút đều thừa nhận phiên truyền thông này thường sẽ được gán một số hiệu nhận diện. Mỗi nút có thể ngắt phiên truyền thông giữa hai nút trên một mạng được gọi là *một cổng luận lý* (Socket). Khi một phiên truyền thông được thiết lập, một cổng luận lý sẽ được mở ra. Một phiên truyền thông được kết thúc được gọi là *một cổng luận lý bị đóng* (Close Socket).

Mục tiêu của tầng phiên là có khả năng cung cấp cho người sử dụng các chức năng cần thiết để quản lý các phiên ứng dụng cụ thể như:

- Điều phối việc trao đổi dữ liệu giữa các ứng dụng bằng cách thiết lập và giải phóng (một cách ***logic***) các phiên (hay gọi là các hội thoại ***dialogues***).
- Cung cấp các điểm đồng bộ hóa để kiểm soát việc trao đổi dữ liệu.
- áp đặt các qui tắc cho các tương tác giữa các ứng dụng của người sử dụng.
- Cung cấp cơ chế lấy lượt (nắm quyền) trong các quá trình trao đổi dữ liệu.

Trong tầng phiên thì vấn đề đồng bộ hóa được thực hiện tương tự như một cơ chế kiểm tra / phục hồi (*check point/reset*).Trong một hệ quản trị tập tin, dịch vụ này cho phép người sử dụng xác định các điểm đồng bộ hóa trong dòng dữ liệu và có thể khôi phục lại việc hội thoại bắt đầu từ một trong các điểm đó.

8.1 Dịch vụ OSI cho tầng Phiên

Tầng phiên làm việc quản lý các cuộc thoại giữa hai máy tính bằng cách thiết lặp, quản lý, và kết thúc các phiên truyền thông.

8.1.1 Cung cấp cho người sử dụng dịch vụ tầng phiên (SS-user)

- Thiết lập một liên kết với một người sử dụng dịch vụ tầng phiên khác, trao đổi dữ liệu với người sử dụng đó một cách đồng bộ và hủy bỏ liên kết một cách có trật tự khi không dùng đến nữa.

- Thương lượng về việc dùng các thẻ bài (TOKEN) để trao đổi dữ liệu,đồng bộ hóa và hủy bỏ liên kết ,sắp xếp phương thức trao đổi dữ liệu (half-duplex hoặc full-duplex).
- Thiết lập các điểm đồng bộ hóa trong các hội thoại và khi xảy ra sự cố thì có thể khôi phục lại việc hội thoại bắt đầu từ một điểm đồng bộ hóa đã thỏa thuận.
- Ngắt hội thoại và khôi phục lại hội thoại sau đó từ một điểm xác định trước.

Các dịch vụ xác định điểm đồng bộ hóa là nhằm vào hai mục đích :

1) Các điểm đồng bộ hóa có thể phân tách các phần của một hội thoại.
2) Các điểm đồng bộ hóa có thể dùng để phục hồi lỗi.

Các điểm đồng bộ hóa chính dùng để cấu trúc quá trình trao đổi dữ liệu thành một chuỗi các đơn vị hội thoại (dialogue), mỗi điểm này phải được xác nhận và người sử dụng sẽ bị hạn chế trong một số dịch vụ nhất định cho tới khi nhận được một sự xác nhận mới. Một điểm đồng bộ hóa chính được dùng để tách biệt các hai đơn vị hội thoại liên tiếp.

Các điểm đồng bộ hóa phụ được dùng để cấu trúc quá trình trao đổi dữ liệu ở trong một đơn vị hội thoại, và các điểm này không cần phải được xác định trước.Việc dùng các điểm đồng hóa phụ trong quá trình truyền tập nó sẽ ngăn chặn việc truyền lại dữ liệu với một khối lượng lớn

Một đơn vị hội thoại là một Activity (hành động) nguyên tử trong đó mọi hành động truyền thông không có liên quan gì đến bất kỳ một hoạt động truyền thông nào trước và sau đó. Một hành động bao gồm nhiều đơn vị hội thoại, và đây cũng chính là một tập hợp logic các nhiệm vụ liên quan với nhau; ở một thời điểm thì chỉ có một activity trên một liên kết phiên nhưng một activity thì có thể diễn ra trên nhiều liên kết phiên, nó có thể bị ngắt và sau đó có thể khôi phục lại trong một liên kết phiên khác, một vòng đời của một liên kết phiên thì có thể có nhiều Activity liên tiếp.

8.1.2 Điều khiển trao đổi dữ liệu

Việc trao đổi dữ liệu xảy như sau để thực hiện một trong ba phương thức như sau : hai chiều đồng thời (*full-duplex*), hai chiều luân phiên (*haft -duplex*), một chiều (*simplex*).

8.1.2.1 Trao đổi dữ liệu một chiều

Liên quan đến các đợt chuyển giao dữ liệu một chiều. Báo cháy là một ví dụ, nó gửi một thông điệp báo động đến trạm chống cháy, nhưng không thể (và không cần) nhận các thông điệp từ trạm chống cháy.

Với phương thức một chiều thì ít xảy ra: chẳng hạn như dữ liệu được gửi đến một đối tượng tạm thời không làm việc, thì chỉ có một chương trình nhận với một nhiệm vụ duy nhất là tiếp nhận dữ liệu đến và giữ lại.

8.1.2.2 Trao đổi dữ liệu hai chiều luân phiên

Liên quan đến các đợt chuyển giao dữ liệu hai chiều, ở đó các luồng dữ liệu mỗi lần đi theo mỗi hướng. Khi một thiết bị hoàn tất một phiên truyền, nó phải " trả lại " vật tải cho thiết bị kia để đến phiên thiết bị đó được truyền.

Với phương thức luân phiên hai chiều thì nảy sinh các vấn đề như sau :

- Các đối tượng sử dụng phiên phải “lấy lượt” để truyền dữ liệu (điển hình của phương thức này là dùng cho các ứng dụng hỏi đáp).
- Thực thể tầng phiên *(session entity)* duy trì tương tác luân phiên bằng cách báo cho các đối tượng khi đến lược họ sẽ truyền dữ liệu.

8.1.2.3 Trao đổi dữ liệu hai chiều đồng thời.

Cho phép tiến hành các đợt chuyển giao dữ liệu hai chiều đồng thời bằng cách cung cấp cho mỗi thiết bị một kênh truyền thông riêng biệt. Điện thoại tiếng là những thiết bị song công đầy đủ, và một trong hai bên của một cuộc đàm thoại có thể nói bất kỳ lúc nào. Hầu hết các môđem máy tính đều có thể hoạt động theo chế độ song công đầy đủ.

Chế độ truyền thông bán song công có thể dẫn đến tình trạng băng thông bị lãng phí trong quãng thời gian mà đợt truyền thông đang quay trả. Trong khi đó, chế độ truyền thông song công đầy đủ thường yêu cầu một ban thông lớn hơn so với chế độ truyền thông bán song công

Với phương thức hai chiều đồng thời thì cả hai bên cùng đồng thời gửi dữ liệu cùng một lúc, một khi phương thức này đã được thỏa thuận thì không đòi hỏi phải có nhiệm vụ quản trị tương tác đặt biệt đây cũng là một phương thức phổ biến nhất.

8.1.3 Điều hành phiên làm việc

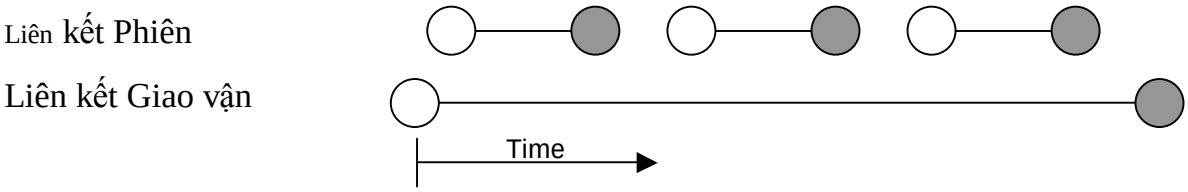
Phiên làm việc (session) là một cuộc thoại chính thức giữa một bên yêu cầu dich vụ và một bên cung cấp dịch vụ. Các phiên bản làm việc thường có ít nhất ba giai đoạn :

- *Thiết lập tuyến liên kết :* Bên yêu cầu dịch vụ sẽ yêu cầu khởi phát một dịch vụ. Trong quá trình xác lập, phiên truyền thông được thiết lập và các quy tắc được thoả thuận.
- *Chuyển giao dữ liệu* : Do các quy tắc được thoả thuận trong khi xác lập, nên mỗi bên của cuộc thoại sẽ biết nội dung mong đợi. Phiên truyền thông sẽ hữu hiệu và các lỗi cũng dễ phát hiện.
- *Giải phóng các kết nối* : Khi hoàn tất phiên làm việc, cuộc thoại kết thúc trong trật tự.

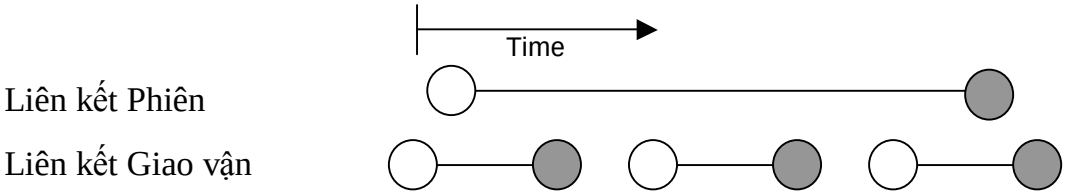
8.1.4 Liên kết phiên

Tầng Phiên thực hiện đặt tương ứng liên kết phiên với các liên kết giao vận. Trong một quá trình liên kết có thể xảy ra 2 trường hợp :

1. Một liên kết giao vận thiết lập với nhiều liên kết phiên liên tiếp :



2. Nhiều liên kết giao vận sử dụng cùng một liên kết phi[a]n:



Ký hiệu : : Thiết lập liên kết

: Giải phóng liên kết

8.2 Giao thức chuẩn tầng phiên

Giao thức chuẩn tầng phiên sử dụng tới 34 loại đơn vị dữ liệu (SPDU) khác nhau,và có khuôn dạng tổng quát như sau :

Trong đó :

- SI: Định danh của loại SPDU (một trong 34 loại)

- LI(length indicator): Chỉ độ dài của vùng tham số(parameters)
- PARAMETERS: vùng khai báo các tham số SPDU, mỗi loại SPDU có danh sách tham số riêng. Mỗi tham số được khai báo dưới dạng tổng quát gồm 3 vùng con : parameter identifier, length indecation, parameter value và chúng được gọi theo đơn vị pi hoặc PGI (mỗi đơn vị PGI gồm có 3 vùng con: PGI, LENGTH INDICATION, PARAMETER VALUE).
- User data: chứa dữ liệu của người sử dụng.

8.2.1 Các loại SPDU, các tham số và chức năng

SPDU	PARAMENTERS	FUNCTION
CONNECT	Connection ID, Protocol Options, Version Number, Serial Number, Token setting, Maximum TSDU size, Requirements, Calling SSAP, Called SSAP, User Data.	Initiate session Connection
ACCEPT	Same as CONNECT SPDU.	Etablist SESSION CONNECTION
REFUSE	Connection ID, Transport disconnect, Requirements, Version number, Season.	Reject connection request
FINISH	Transport Disconnect, User Data.	Initiate Orderly Release
DISC0NNECT	User Data.	Acknowledge orderly Release
NOT FINISHED	User Data.	Reject Orderly Release
ABORT	Transport disconnect, Protocol Error Code, User Data.	Abnormal connection Release
ABORT ACCEPT	Transport disconnect, Protocol Error Code, User Data.	Acknowledge Abort
DATA TRANSFER	Enclosure item,User Data.	Transfer normal Data
EXPEDITED	User data.	Transfer typed data
CAPABILITY DATA ACK	User Data.	Acknowledge Capability data
GIVE TOKENS	Tokens.	Transfer tokens
PLEASE TOKENS	Tokens , User Data.	Request token Assignment
GIVE TOKENS CONFIRM	-	Transfer all tokens
GIVE TOKENS ACK	-	Acknowledge all tokens
MONOR SYNC POINT	Confirm required flag, Serial number, User data.	Define minor sync point
MINOR SYNC ACK	Serial number, User Data.	Acknowledge minor sync point
MAJOR SYNC POINT	End of activity flag, Serial number, User Data.	Define major sync point
MAJOR SYNC ACK	Serial number,User data.	Acknowledge major sync point
RESYNCHRONIZED	Tokens sittings, resync type, serial number, user data.	Resynchorize
RESYNCHRONIZED ACK	Tokens settings, Serial number, User Data.	Acknowledge resynchronize

PREPERE	Type.	Notify type SPDU is coming
EXCEPTION REDORT	SPDU bit pattem.	Protocol Error detected
EXCEPTION DATA	Reason, User Data.	Put protocol in Error state
ACTIVITY START	Activity ID, User data.	Signal beginning of activity
ACTIVITY RESUME	Connect ID, Old activity ID, New Activity ID, User data.	Signal resumption of activity
ACTIVITY INTERRUPT	Reason.	Interrupt activity
ACTIVITY INTERRUPT ACK	-	Acknowledge interrupt
ACTIVITY DISCARD	Reason.	Cancel activity
ACTIVITY DISCARD ACK	-	Acknowledge cancellation
ACTIVITY END	Serial number/User data.	Signal activity end
ACTIVITY END ACK	Serial number/User data.	Acknowledge activity end

Tầng Phiên đóng một vai trò quan trọng trong việc trao đổi thông tin giữa các máy Client với máy Server. Nhưng thông tin mà chúng ta cần truyền tải thì được chia nhỏ ra thành các khung (hay gói) trước khi chúng được truyền tải qua một mạng. Mỗi tầng của mô hình 7 tầng OSI đều có thể bổ sung thêm các thông tin vào đoạn đầu và đoạn cuối của một khung dữ liệu vvà sau đó các thông tin này sẽ được đọc bởi tầng tương đương ở máy trạm tiếp nhận. Và một số tầng khác có thể bổ sung thêm phần đầu(header) và cả một phần đuôi(trailer) vào khung dữ liệu có sẵn. Sau đó, khung dữ liệu này truyền chuyển tới tầng tương đương trên trạm tiếp nhận.

Chương 9

TẦNG TRÌNH DIỄN

Tầng Trình diễn có nhiệm vụ phân cách giữa các tầng cao hơn và các tầng thấp hơn từ định dạng dữ liệu của tầng ứng dụng, chuyển đổi định dạng dữ liệu từ định dạng của tầng ứng dụng thành định dạng thông thường, gọi là "trình diễn hợp với quy tắc". Tầng Trình diễn xử lý dữ liệu không phụ thuộc vào máy tính từ tầng ứng dụng thành dữ liệu có định dạng phụ thuộc vào máy tính để chuyển cho các tầng thấp hơn.

Tầng trình diễn xử lý cú pháp, hoặc các quy tắc văn phạm, cần thiết cho phiên truyền thông giữa hai máy tính, bảo đảm cho các hệ thống cuối truyền th6ng có kết quả khi chúng sử dụng các dạng biểu diễn dữ liệu khác nhau. Tầng này trình bày một dạng thức dữ liệu đồng dạng cho tầng ứng dụng.

9.1 Vai trò và chức năng

Mục đích của tầng trình diễn là đảm bảo cho các hệ thống cuối có thể truyền thông có kết quả ngay cả khi chúng sử dụng các biểu diễn dữ liệu khác nhau .Để đạt được điều đó nó cung cấp một biểu diễn chung để dùng trong truyền thông và cho phép chuyển đổi từ biểu diễn cục bộ sang biểu diễn chung đó.

Tồn tại 3 dạng cú pháp thông tin được trao đổi giữa các thực thể ứng dụng :

- Cú pháp dùng bởi thực thể ứng dụng nguồn.
- Cú pháp dùng bởi thực thể ứng dụng đích.
- Cú pháp dùng bởi giữa các thực thể trình diễn ,loại cú pháp này gọi là cú pháp truyền (transfer syntax).

Tầng trình diễn đảm nhận việc chuyển đổi biểu diễn thông tin giữa cú pháp truyền và mỗi một cú pháp kia khi có yêu cầu

Chú ý rằng không tồn tại một cú pháp truyền xác định trước duy nhất cho mọi hoạt động trao đổi dữ liệu. Cú pháp truyền được duy nhất cho mọi hoạt động trao đổi dữ liệu. Cú pháp truyền được sử dụng trên một liên kết cụ thể của tầng trình diễn phải được thương lượng giữa các thực thể trình diễn tương ứng. Mỗi bên lựa chọn một cú pháp truyền sao cho có thể sẵn sàng được chuyển đổi sang cú pháp người sử dụng và ngược lại. Ngoài ra cú pháp truyền được chọn phải phản ánh các yêu cầu dịch vụ khác chẳng hạn nhu cầu nén dữ liệu .việc thương lượng cú pháp truyền sử dụng có thể được thay đổi trong vòng đời liên kết đó .Tầng trình diễn chỉ liên quan đến cú pháp truyền vì thế trong giao thức sẽ không quan tâm đến các cú pháp sử dụng bởi thực thể ứng dụng. Tuy nhiên mỗi thực thể trình diễn phải chịu trách nhiệm chuyển đổi giữa cú pháp người sử dụng và cú pháp truyền.

Các khái niệm liên quan đến bối cảnh của tầng trình diễn : Khi qua ranh giới giữa hai tầng trình diễn và tầng phiên có một sự thay đổi quan trọng trong cách nhìn dữ liệu. Đối với tầng phiên trở xuống tham số User Data trong các service primitives được đặc tả dưới dạng nhị phân (một chuỗi các byte). Giá trị này có thể được đưa vào trực tiếp trong các SDU (Service Data Unit) để chuyển giữa các tầng trong một hệ thống và trong các PDU (Protocol Data Unit) để chuyển giữa các tầng đồng mức ở hệ thống kết nối với nhau. Tuy nhiên tầng ứng dụng lại liên quan chặt chẽ với cách nhìn dữ liệu của người sử dụng nói chung cách nhìn đó là một tập thông tin có cấu trúc nào đó như là văn bản (text) trong một tài liệu một tệp về nhân sự hoặc một cơ sở dữ liệu .... Người sử dụng chỉ quan tâm đến ngữ nghĩa (semantics) của dữ liệu. Do đó tầng trình diễn ở giữa chỉ có nhiệm vụ cung cấp phương thức biểu diễn dữ liệu và chuyển đổi thành các giá trị nhị phân dùng cho các tầng dưới nghĩa là tất cả những gì liên quan đến cú pháp của dữ liệu

Tuy nhiên trong thực tế không thể tách bạch hoàn toàn giữa cú pháp và ngữ nghĩa và ngữ nghĩa dữ liệu. Nếu tầng ứng dụng không biết gì về cú pháp thì tầng trình diễn không biết gì về nhữ nghĩa thì không thể nào hoàn tất được việc kết hợp ngữ nghĩa với cú pháp dùng để tạo ra một biểu diễn cụ thể các giá trị dữ liệu cho dịch vụ phiên.

ở tầng ứng dụng thông tin được biểu diễn dưới dạng cú pháp trừu tượng (abstract syntax) liên quan đến các kiểu dữ liệu (data values) cú pháp trừu tượng này đặc tả một cách nhìn hình thức dữ liệu độc lập với mọi biểu diễn cụ thể.

Do vậy một cú pháp trừu tượng có nhiều đặc điểm giống kiểu dữ liệu như các ngôn ngữ lập trình Pascal, C .... Các ngữ nghĩa như là BNF. Các giao thức tầng ứng dụng mô tả các PDU của chúng bằng một cú pháp trừu tượng. Tầng trình diễn tương tác với tầng ứng dụng cũng dựa trên cú pháp trừu tượng này, tầng trình diễn có nhiệm vụ dịch thuật cú pháp trừu tượng của tầng ứng dụng và cú pháp truyền (transfer syntax) mô tả các giá trị dữ liệu dưới dạng nhị phân thích hợp cho việc tương tác với dịch vụ phiên việc dịch thuật này được thực hiện nhờ qui tắc mã hoá chỉ rõ biểu diễn của mỗi giá trị dữ liệu thuộc một kiểu nào đó .

Trước khi sử dụng liên kết của một tầng trình diễn để trao đổi dữ liệu thì hai thực thể trình diễn ở hai đầu phải thoả thuận về cú pháp truyền được xem như là bối cảnh trình diễn (presentation context) được dùng để trao đổi dữ liệu

Cú pháp truyền phải yểm trợ cú pháp trừu tượng tương ứng. Ngoài ra cú pháp truyền có thể có các thuộc tính khác không liên quan gì đến cú pháp trừu tượng mà nó yểm trợ ví dụ một cú pháp trừu tượng có thể yểm trợ bởi bất kì một cú pháp truyền về cơ bản thì giống nhau chỉ khác nhau ở chỗ một cung cấp khả năng mật mã, một chỗ cung cấp cả hai và một không cung cấp khả năng nào.

9.1.1 Phiên dịch dữ liệu

Một mục tiêu quan trọng cần giải quyết khi thiết kế các mạng đó là cho phép kiểu máy tính khác nhau trao đổi dữ liệu. Tuy mục tiêu này ít khi được giải quyết toàn vẹn, nhưng việc vận dụng hiệu quả các kỹ thuật phiên dịch dữ liệu có thể giúp nhiều kiểu máy tính truyền thông với nhau. Có bốn dạng phiên dịch dữ liệu, thứ tự bít, thứ tự byte, mã ký tự, và cú pháp tập tin như sau :

- Thứ tự bit : Khi số nhị phân được truyền qua một mạng, chúng gởi đi theo từng bit, thứ tự byte, mã ký tự, và cú pháp tập tin.
- Phiên dịch thứ tự Byte : Các giá trị phức tạp thường phải được biểu thị bằng nhiều byte, nhưng các máy tính khác nhau thường dùng quy ước khác nhau về việc sẽ truyền byte nào trước. Các bộ vi xử lý Intel bắt đầu bằng byte ít quan trọng nhất. Do chúng bắt đầu tại đầu nhỏ, nên được gọi là kết đầu nhỏ. Các bộ vi xử lý Motorola bắt đầu bằng byte quan trọng nhất. Để hoà hợp những khác biệt này, ta cần phải có tính năng phiên dịch thứ tự byte.
- Phiên dịch mã ký tự : Hầu hết các máy tính đều dùng một trong các bảng mã đánh số nhị phân dưới đây để biểu thị các bộ ký tự : Bảng mã ASCII được dùng để biểu thị các ký tự tiếng Anh trên tất cả máy tính và hầu hết các máy tính mini. EBCDIC (Extended Binary Coded Decimal Interchange Code = Mã hoán đổi thập phân mã hoá nhị phân mở rộng) được dùng để biểu thị cho các ký tự tiếng Anh trên máy tính lớn nhất.
- Phiên dịch cú pháp tập tin : Khi các dạng thức tập tin khác nhau giữa các máy tính, các dạng đó đòi hỏi phải phiên dịch.

9.2 Dịch vụ OSI cho tầng trình diễn

Dịch vụ OSI cho tầng trình diễn có 2 loại : một loại bao gồm các dịch vụ liên quan đến biểu diễn của dữ liệu người sử dụng để đảm bảo cho hai thực thể ứng dụng có thể trao đổi dữ liệu thành công ngay khi chúng dùng các biểu diễn cục bộ khác nhau cho dữ liệu đó, loại thứ hai bao gồm các dịch vụ cho phép các thực thể ứng dụng có thể sử dụng các dịch vụ tầng phiên để quản lý hội thoại.

Để cung cấp loại dịch vụ thứ nhất tầng trình diễn thực hiện hai nhiệm vụ sau :

- Thương lượng về cú pháp truyền : với mỗi kiểu dữ liệu người sử dụng cho trước một cú pháp truyền được thương lượng.
- Chuyển đổi : dữ liệu cung cấp bởi người sử dụng được chuyển đổi thành biểu diễn theo cú pháp truyền để truyền đi , ngược lại dữ liệu nhận được để giao cho người sử dụng sẽ chuyển đổi từ biểu diễn theo cú pháp truyền sang biểu diễn của người sử dụng.

ở thời điểm bất kì trong vòng đời của một liên kết trình diễn dịch vụ trình diễn dịch vụ trình diễn có liên quan đến một hoặc nhiều bối cảnh trình diễn (presentation context). Mối bối cảnh chỉ rõ cú pháp trừu tượng của dữ liệu đó. Có hai loại bối cảnh được sử dụng :

- Defined context set : bao gồm các bối cảnh đã được xác định thông qua sự thoả thuận giữa người sử dụng dịch vụ trình diễn (presentation service user) vaf người cung cấp dịch vụ trình diễn (presentation service provider).
- Default context : là một bối cảnh trình diễn mà người cung cấp dịch vụ trình diễn luôn luôn biết rõ và người sử dụng khi vắng mặt

Ở tầng phiên do kiến trúc phân tầng của ISO các thực thể ứng dụng không thể truy cập trực tiếp tới các dịch vụ tầng phiên, do vậy các yêu cầu dịch vụ liên quan đến tầng phiên phải được chuyển qua tầng trình diễn đến các dịch vụ tầng phiên.

9.3 Giao thức chuẩn tầng trình diễn

Giao thức chuẩn của ISO/CCITT cho tầng Trình diễn đặc tả những nội dung chính sau đây:

- Cấu trúc và mã hoá các đơn vị dữ liệu của giao thức trình diễn (PPDU) dùng để truyền dữ liệu và thông tin điều khiển .
- Các thủ tục để truyền dữ liệu và thông tin điều khiển giữa các thực thể trình diễn của hai hệ thống mở.
- Liên kết giữa giao thức trình diễn với dịch vụ trình diễn và với dịch vụ phiên .

Cũng như các PDU ở các tầng khác ,các PPDU cũng có khuôn dạng tổng quát bao gồm một phần đầu (header) chứa các thông tin điều khiển và có thể thêm mmột phần chứa dữ liệu được truyền từ trên xuống hoặc được truyền lên cho tầng trên. Giao thức trình diễn sử dụng 14 PPDU được liệt kê trong bảng 2-17 cùng với các tham số của chúng .

Qua bảng trên ta thấy số lượng PPDU không nhiều như số lượng SPDU (ở tầng Phiên) và nhiều tham số (có đánh dấu *) là giống với các tham số của các SPDU. Như vậy cả về phương diện dịch vụ và giao thức, tầng trình diễn và tầng Phiên có một mối liên kết rất chặt chẽ .

Qua xem xét các tầng dưới từ tầng phiên trở xuống, chúng ta thấy có 2 nguyên lý sau đây luôn được tuân thủ :

- Mỗi dịch vụ tầng n được cài đặt nhờ trao đổi các nPDU;
- Mỗi nPDU trở thành User data và được “nhét” vào trong một (n-1) PDU;

Tuy nhiên ở tầng trình diễn (và cả ở tầng ứng dụng mà ta sẽ thấy), các nguyên lý đó không còn luôn luôn được áp dụng .Thực tế là không phải mọi dịch vụ trình diến đều yêu cầu các PPDU và một số tham số của một số PPDU không được chuyển thành User data trong một SPDU. Để giải thích động cơ củ a sự khác biệt đó, ta xem xét hai dịch vụ trình diễn : thiết lập liên kết (connection etablishment) và chuyển thẻ bài (token passing).

Khi phát triển các giao thức cho 3 tầng cao của Mô hình óI,người ta thấy rõ ràng nên thương lượng và thiết lập đồng thời các liên kết Phiên ,trình diễn và ứng dụng,mặc dù điều đó đòi hỏi một quan hệ 1-1 chặt chẽ (không có dồn kênh) với cùng vòng đời cho cả ba loại liên kết .Quá trình thiết lập đồng thời các liên kết đó được gọi là quá trình nhúng (embeding), vì các PDU CONNECT.request và CONNECT.response cho cả ba tầng cao đó ,cái này được nhúng vào trong cái kia.

Khuôn dạng của các PDDU header được đặc tả theo cú pháp trừu tượng chuẩn.

9.3.1 Các chuẩn khác cho tầng trình diễn

Ngoài các chuẩn về dịch vụ và giao thức cho tầng Trình diễn như đã trình bày ở trên, ISO và CCITT đã phát triển các chuẩn liên quan đến cú pháp trừu tượng (Abstract Syntas) và quy tắc mã hoá (Encoding Rules) mà chúng ta đã nói đến khi trình bày vai trò và chức năng của tầng Trình diễn

Các chuẩn của ISO gồm có :

- ISO 8824:Abstract Syntax Notation One(viết tắt là ASNI.1)
- ISO 8825:Báic Encoding Rules(Viết tắt là BER)
- Tương ứng CCITT có các khuyến nghị X208 (ANSI.1) và X.209 (BER).

Khái niệm cú pháp trừu tượng mà ISO và CCITT định nghĩa được dựa trên khái niệm kiểu dữ liệu (data type) mà chúng ta đã quen thuộc trong các ngôn ngữ lập trình phổ biến. Thông thường các ngôn ngữ này định nghĩa trước các kiểu dữ liệu đơn giản như integer và boolean, cùng với các phương thức tổ hợp các kiểu đơn giản đó để có các cấu trúc dữ liệu phức tạp hơn. Hơn nữa, các phương pháp tổ hợp có thể thực hiện một cách đệ quy cho phép tạo ra các kiểu phức tạp tuỳ ý.

Chương 10

TẦNG ỨNG DING

Tầng ứng dụng giao tiếp trực tiếp với người sử dụng. Nhiệm vụ của tầng ứng dụng là hiển thị các thông tin nhận được và gửi các thông tin mới của người sử dụng cho các tầng thấp hơn.

Tầng ứng dụng liên quan đến tiến trình cung cấp các dịch vụ trên mạng, các dịch vụ này bao gồm : dịch vụ tập tin, dịch vụ in, dịch vụ cơ sở dữ liệu, và các dịch vụ khác.

Chúng ta sẽ xem xét các vấn đề trước khi bắt đầu với các ứng dụng. Đó là sự an toàn mạng, dịch vụ tên miền DNS dùng để điều khiển đặt tên trong Internet, giao thức hỗ trợ quản trị mạng, phần còn lại là các ứng dụng thực như thư điện tử, UserNet, FTP, Telnet, WWW ...

10.1 An toàn thông tin trên mạng

Việc kết nối mạng máy tính nhằm sử dụng và chia sẻ tài nguyên của các đối tượng trong hệ thống mạng cho dù họ có thể cách xa nhau về mặt địa lý. Tài nguyên hệ thống ở đây chủ yếu là là thông tin. Tuy nhiên đây là loại tài nguyên dễ bị xâm phạm, bị đánh cắp, bị tráo đổi nhất, đặc biệt là nó đang được trong lưu giữ trong môi trường mạng đầy phức tạp và phải chia xẻ cho nhiều người dùng khác nhau ở những vị trí khác nhau.

Vấn đề an toàn thông tin trên mạng đòi hỏi phải sử dụng nhiều biện pháp khác nhau từ cơ bản đến phức tạp, tuỳ theo lượng thông tin cần bảo vệ và khả năng cho phép của từng hệ thống cụ thể.

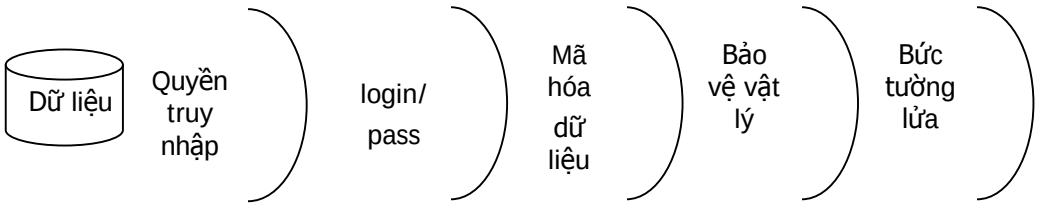
10.1.1 Các chiến lược an toàn hệ thống

1. Quyền hạn tối thiểu : Đây là chiến lược nền tảng nhất. Theo nguyên tắc này bất kì đối tượng nào cũng chỉ có những quyền hạn nhất định đối với những tài nguyên mạng nhất định khi thâm nhập vào mạng.
2. Bảo vệ theo chiều sâu : Tạo nhiều cơ chế an toàn cho hệ thống để chúng hỗ trợ cho nhau.
3. Cơ chế nút thắt : Tạo ra một “cửa khẩu” hẹp và chỉ cho phép thông tin đi vào hệ thông của mình bằng duy nhất con đường này. Đồng thời phải tổ chức một cơ chế kiểm soát và điều khiển các luồng thông tin đi qua cửa khẩu này.
4. Tính toàn cục : Các hệ thống an toàn đòi hỏi phải có tính toàn cục của các hệ thống cục bộ. Nếu có kẻ nào đó có thể bẻ gãy một cơ chế an toàn thì chúng có thể thành công bằng cách tấn công hệ thống nội bộ từ bên trong.

5. Tính đa dạng của việc bảo vệ : Cần phải sử dụng nhiều biện pháp khác nhau cho những hệ thống khác nhau. Nếu không, kẻ nào đó tấn công được hệ thống này thì cũng có thể tấn công vào hệ thống khác.

- Các mức bảo vệ thông tin trên mạng:

Vì không có một giải pháp bảo vệ nào an toàn tuyệt đối nên người ta thường sử dụng nhiều mức bảo vệ khác nhau tạo thành nhiều lớp rào chắn cho hệ thống. Mô hình như sau :



Hình 10-1. Các mức bảo vệ thông tin trên mạng.

10.1.2 An toàn thông tin bằng mã hóa

Để bảo vệ thông tin trên đường truyền, người ta chuyển đổi nó từ dạng nhận thức được sang dạng không nhận thức được trước khi truyền đi trên mạng nhằm bảo đảm tính bí mật cần thiết. Quá trình này diễn ra ở trạm phát được gọi là mã hoá thông tin (encrypting), ở trạm nhận phải thực hiện quá trình ngược lại, tức là biến đổi thông tin từ dạng không nhận thức được (đã mã hoá) sang dạng nhận thức được (dạng gốc), quá trình này gọi là giải mã (decrypting). Đây là một lớp bảo vệ thông tin rất quan trọng và được ứng dụng trong hầu hết các hệ thống mạng.

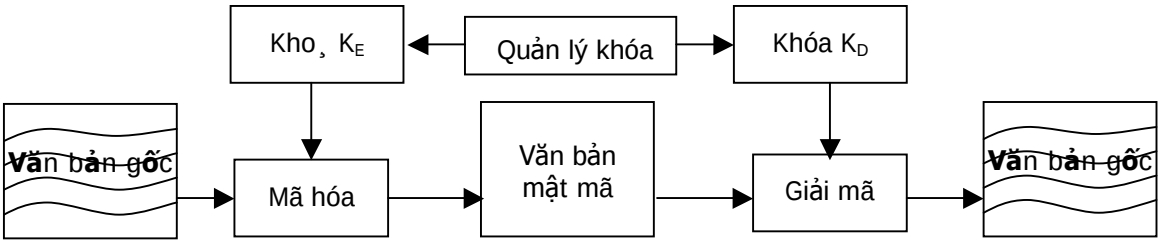
Để bảo vệ thông tin bằng mật mã, người ta thường tiếp cận theo hai hướng:

- Từ nút đến nút (end_to_end)
- Theo đường truyền (link_oriented security)

Theo cách thứ nhất, thông tin được mã hoá để bảo vệ trên đường truyền giữa hai nút mà không quan tâm đến nguồn và đích của thông tin đó. Ớ đây ta chú ý rằng thông tin chỉ được bảo vệ trên đường truyền, tức là ở mỗi nút đều có quá trình giải mã để sau đó thông tin được chuyển đi tiếp, do đó các nút cần được bảo vệ tốt.

Ngược lại theo cách thứ hai, thông tin trên mạng được bảo vệ trên toàn đường truyền từ nguồn đến đích. Thông tin sẽ được mã hoá ngay sau khi mới tạo ra và chỉ được giải mã khi đã về đến đích. Cách này có nhược điểm là chỉ có dữ kiệu người dùng mới được mã hoá còn các thông tin điều khiển thì phải giữ nguyên để có thể xử lí tại các nút.

Quá trình mã hoá và giải mã được mô tả như sau :



Hình 10-2. Sơ đồ quá trình mã hóa.

+ Văn bản gốc (plaintext) là văn bản chưa được mã hoá.

+ Khoá (key) : gồm một số hữu hạn các bit thường được biểu thị dưới dạng các xâu kí tự chữ số, số thập phân hoặc thập lục phân. Trong thực tế thường dùng các khoá có 8 kí tự.

Nếu gọi : M là văn bản gốc

C là văn bản mật mã (Ciphertext)

E là hàm mã hoá (Encryption Function)

D là hàm giải mã (Descryption Function)

Ta có hàm biểu diễn sự phụ thuộc giữa văn bản gốc và văn bản mã như sau:

$C = E(M)$

$M = D(C) = D(E(M))$

Khoá KE được dùng để mã hoá, khoá KD được dùng để giải mã .

Có rất nhiều phương pháp mã hoá nhưng tất cả đều qui về 2 phương pháp chung tuỳ theo việc sử dụng cặp khoá KD và KE:

- Khoá KD trùng với khoá KE : phương pháp này gọi là mã hoá khoá đối xứng, với phương pháp này yêu cầu khoá phải được giữ bí mật tuyệt đối, vì khoá dùng để mã hoá cũng được dùng giải mã.
- Khoá KD khác với khoá KE : phương pháp này gọi là mã hoá khoá công khai. Trong đó, có thể chuyển đổi vai trò giữa 2 khoá và rất khó để suy ra khoá này từ khoá kia. Khoá mã hoá (KE) có thể đưa ra công khai nhưng khoá dùng giải mã (KD) phải được giữ bí mật tuyệt đối.

Người ta còn phân biệt 2 loại khóa:

- Các khoá dùng trong thời gian dài gọi là khoá chính (primary) hay khoá mã hoá (key encryption).
- Các khoá được dùng trong khuôn khổ một cuộc truyền thông gọi là khoá làm việc (working) hay khoá mã hoá dữ liệu (data encryption).

10.2 CáC phương pháp mã hóa dữ liệu

10.2.1 Phương pháp hoán vị

Phương pháp này sắp xếp lại các kí tự trong văn bản gốc để tạo ra văn bản mật mã. Phương pháp này có một số kỹ thuật sau :

1. Đảo ngược toàn bộ văn bản gốc

Từ văn bản gốc, ta mã hoá bằng cách viết theo thứ tự ngược lại. Ví dụ DHKTDN được mã hoá thành NDTKHD. Đây là một trong những phương pháp mã hoá đơn giản nhất và chỉ mang tính tham khảo vì không an toàn.

2. *Mã hoá theo mẫu hình học*

Sắp xếp lại văn bản gốc theo mẫu hình học nào đó (thường là ma trận 2 chiều) để tạo văn bản mật mã.

Ví dụ : ĐAIHOCĐANANG được viết thành ma trận 3 x 4:

Đ	A	I	H
O	C	Đ	A
N	A	N	G

Nếu ta lấy các kí tự ra theo thứ tự các hàng là 3,1,2 ta sẽ có văn bản mật mã là N A N G O C Đ A Đ A I H. Phương pháp cũng kém an toàn, có thể dựa vào tần số xuất hiện của các kí tự trong bản mã để suy ra văn bản gốc.

3. *Đổi chổ cột*

Sắp xếp lại văn bản gốc thành dạng hình chữ nhật theo các cột, sau đó các cột được sắp xếp lại và lấy các kí tự theo chiều ngang.

Ví dụ : văn bản TRUONGDAIHOCKYTHUATDANANG được viết thành ma trận 5 x 5 :

Cột	1	2	3	4	5
Văn bản	T	R	U	O	N
	G	D	A	I	H
	O	C	K	I	T
	H	U	A	T	D
	A	N	A	N	G

Vì có 5 cột nên có thể sắp xếp lại theo 5!= 120 cách khác nhau. Nếu ta chuyển vị các cột theo thứ tự 2,3,4,1,5 rồi lấy các kí tự theo hàng ta sẽ có văn bản mã như sau: RUOTN DAIGH CKYOT UATHD NANAG.

Ta thấy rằng, với một văn bản càng lớn (nhiều kí tự) số cách sắp xếp có thể sẽ rất lớn làm tăng khả năng an toàn. Hạn chế của phương pháp này là toàn bộ ma trận kí tự phải được sinh để mã hoá và giải mã và cũng dễ nhầm lẫn trong việc giải mã.

4. Hoán vị các kí tự của văn bản gốc theo chu kì cố định T

Cho hàm f là hoán vị của một khối gồm T kí tự thì khoá mã hoá được biểu diễn bởi hàm K(T,f). Do vậy, văn bản gốc :

$M = m_1\ m_2\ md\ md_{+1} \ldots m_{2d}$

Trong đó m_i là các kí tự riêng lẻ sẽ được mã hoá thành :

$Ek(M) = mf_{(1)}\ mf_{(2)} \ldots mf_{(d)}\ md_{+f(1)} \ldots md_{+f(d)}$

Với $mf_{(1)}\ mf_{(2)} \ldots mf_{(d)}$ là một hoán vị của $m_1\ m_2 \ldots md$

Ví dụ : giả sử T=7 và f hoán vị dãy i = 12345 thành f(i)=23415, chẳng hạn từ gốc STUDY được biểu diễn như sau :

Vị trí đầu	*Vị trí hoán vị*	*từ*	*Mã hoá*
1	2	S	T
2	3	T	U
3	4	U	D
4	1	D	S
5	5	Y	Y

Bằng cách đó văn bản gốc TRUONGDAIHOCKYTHUATDANANG được mã hoá thành RUOTN DAIGH CKYOT UATHD NANAG

10.2.2 Phương pháp thay thế

Phương pháp này mã hoá văn bản bằng cách thay thế mỗi kí tự trongvăn bản bằng một kí tự khác nào đó (có thể là chữ cái, chữ số hoặc kí hiệu), có thể dùng một trong các phương pháp thay thế sau :

1. Thay thế đơn giản

Mỗi kí tự trong văn bản gốc được thay thế bằng một kí tự tương ứng trong văn bản mật mã. Một ánh xạ 1 – 1 được dùng để mã hoá và giải mã thông điệp.

2. Thay thế đồng âm

Mỗi kí tự trong văn bản gốc được mã hoá với một số kí tự của văn bản mật mã (ánh xạ 1 - n). Ngoài ra còn một số phương pháp thay thế khác như thay thế đa mẫu tự, thay thế theo sơ đồ...

Một trong những mật mã thay thế đơn giản được biết đến nhiều nhất là mã Morse, trong đó các chữ cái được thay thế bằng các kí tự gạch và chấm. Bảng mã ASCII ta thường dùng cũng là một dạng mật mã thay thế đơn giản. Trong đó, chữ A

được biểu diễn bằng chuỗi số nhị phân 1000001 hoặc số thập phân 65, chữ B được biểu diễn bởi 1000010 hoặc 66,v.v...

Một dạng mật mã khác cũng được biết đến nhiều đó là bảng chữ cái dịch chuyển. ở đây, các chữ cái trong bảng được dịch chuyển sang phải k vị trí, k gọi là khoá. Ta có hàm dịch chuyển : f(a) = (a + k) mod n với a là một chữ cái trong bảng mã, n là số chữ cái (n=26 nếu là bảng chữ tiếng Anh chuẩn).

Ví dụ : với k = 5 ta có :

Bảng chữ gốc : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Bảng chữ mật mã : F G H I J K LMN O P Q R S T U V W X Y Z A B C D E

Để giải mã, người nhận phải biết khoá k và phục hồi lại văn bản gốc bằng cách biến đổi ngược.

Như vậy nếu văn bản gốc là TRUONG DAI HOC KY THUAT DA NANG thì văn bản mã sẽ là YWZTSL IFN MTH PN YMZFY IF SFSL. Phương pháp này kém an toàn vì chỉ cần thử lần lượt 26 khả năng là ta đã giải mã được.

Một phương pháp thay thế khác tốt hơn là dùng từ khoá theo sơ đồ Vignère. Theo sơ đồ này, từ khoá được cộng vào liên tiếp theo từng kí tự một cho văn bản gốc, mỗi kí tự được biểu diễn bởi một vị trí của nó trong bảng kí tự và phép cộng được thực hiện theo môđun 26. Ví dụ, giả sử ta có bằng :

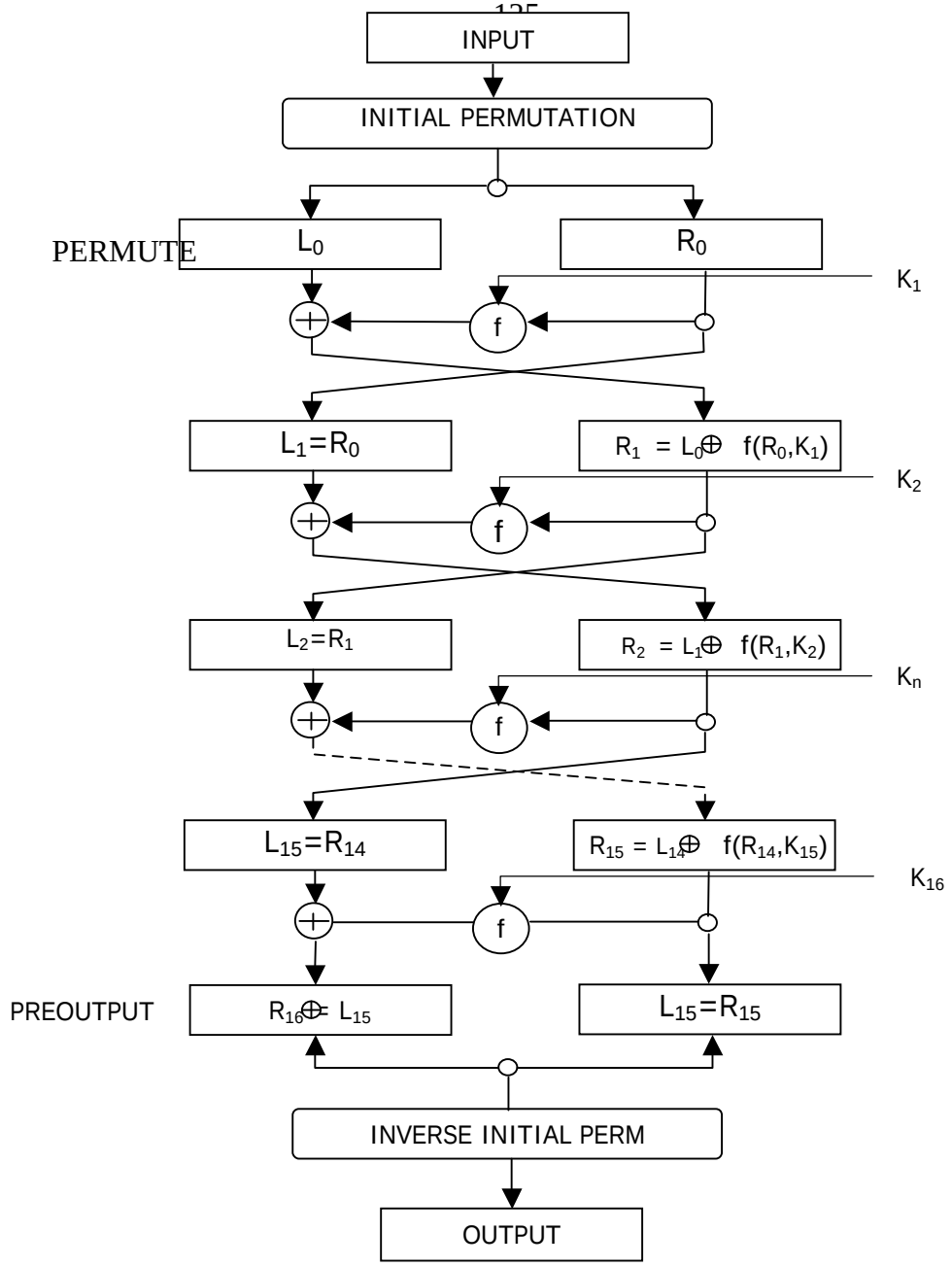
Vị trí	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ký tự	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S

19	20	21	22	23	24	25
T	U	V	W	X	Y	Z

Cho từ khoá là ABCDEF, văn bản DAIHOC được mã hoá thành DBKKSH. Lưu ý rằng ở trên các từ được viết tách ra cho dễ đọc, trên thực tế dữ liệu được truyền thành từng dòng liên tục để sau đó bí mật tính chu kì.

10.2.3 Phương pháp mã hóa chuẩn DES

Những thuật toán hiện đại hơn dùng kết hợp cả hai kỹ thuật thay thế và đổi chỗ. Trong đó phương pháp được biết đến nhiều nhất là chuẩn mã hoá dữ liệu DES (Data Encryption Standard) được Mỹ và hãng IBM phát triển trong những năm 70. Lưu đồ mã hoá DES được mô tả như hình sau.

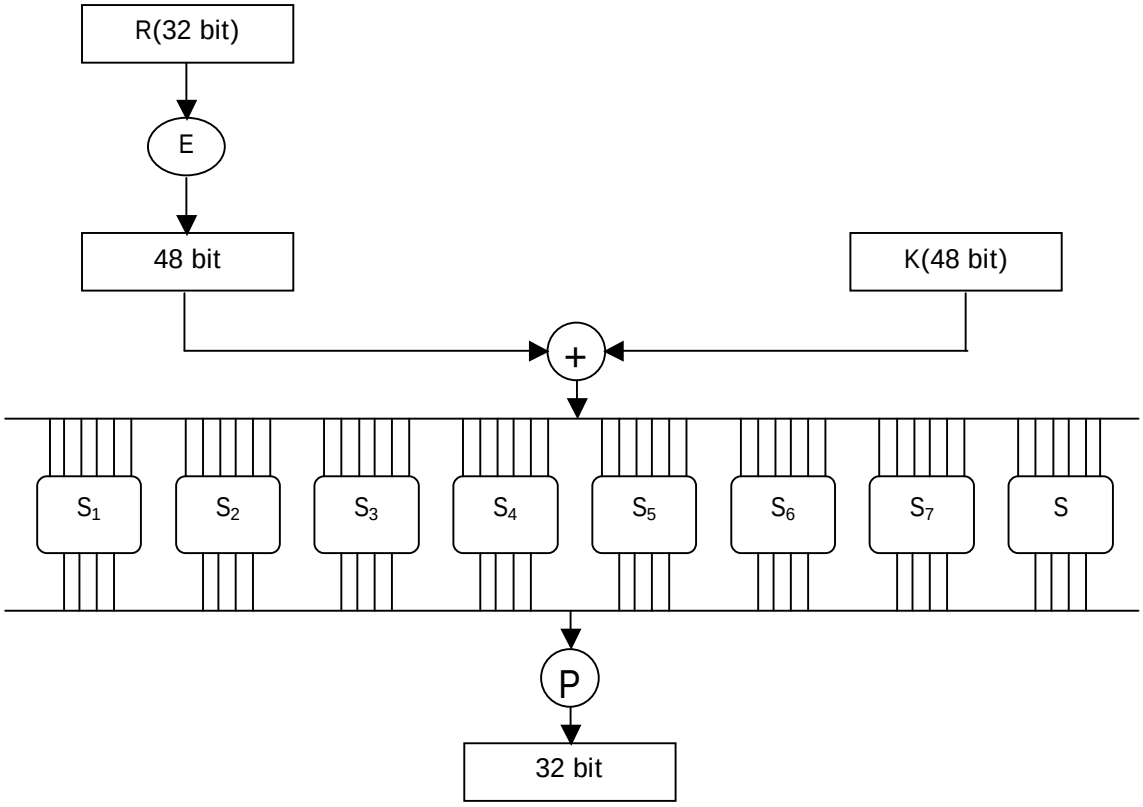


Hình 10-3. Sơ đồ mã hoá DES.

Đầu vào là một dãy 64 bit biểu diễn một khối các kí tự trong văn bản gốc và đầu ra là một dãy 64 bit biểu diễn văn bản mã. Quá trình mã hoá được chia làm 3 giai đoạn :

Đầu tiên văn bản gốc được chuyển qua bộ hoán vị khởi đầu (initial permutation-IP) để tạo ra 64 bit đã hoán vị . Sau đó thực hiện 16 phép lặp của một hàm chữ số (cipher function), kí hiệu là f(R,K) là tổ hợp cả kĩ thuật hoán vị lẫn kĩ thuật thay thế. Trong đó R là dãy con phải (32 bit) của văn bản gốc, khoá K có độ dài 56 bit. 64 bit đầu ra được làm đầu vào cho hoán vị ngược với hoán vị khởi đầu IP^{-1} để tạo ra 64 bit văn bản gốc.

Chi tiết của hàm f(R,K) được mô tả như sau :



Hình 10-4. Hàm f(R,K).

Phép toán của f(R,K) :

Giả sử, bit đầu tiên trong kết quả hoán vị là bit 58 trong dãy ban đầu, bit thứ 2 trong kết quả là bit thứ 50 trong dãy ban đầu,v.v... Dãy hoán vị được chia làm 2 dãy con 32 bit : dãy con trái, kí hiệu là L_0 trong sơ đồ, và dãy con phải kí hiệu là R_0. Hàm f(R,K) dùng các phép toán thay thế và một khoá K_1 để chuyển R_0 thành một dãy 32 bit mới, kí hiệu $f(R_0,K_1)$. Dãy bit này được cộng vào L_0 từng bit một theo mođun 2(phép toán cộng loại trừ) để tạo ra dãy con phải ở giai đoạn tiếp theo. Dãy R_0 ban đầu trở thành dãy con trái L_1.

Phép hoán vị ban đầu IP được cho như bảng dưới đây :

IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Chuỗi các phép toán được thực hiện 16 lần với 16 khoá khác nhau K_1, K_2,..., K_{16}, ngoại trừ một điều là không có "phép chuyển qua" ở giai đoạn cuối cùng. Những phép toán này tạo ra dãy 64 bit $R_{16}L_{16}$, được đánh dấu PREOUTPUT trong sơ đồ. Phép toán ngược IP^{-1} của phép hoán vị IP được dùng để biến đổi dãy PREOUTPUT để tạo ra bản mã cuối cùng.

IP^{-1}							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Dãy con phải được kí hiệu bởi R trước hết được mở rộng thành một dãy số 48 bit dùng bảng chọn bit E sau đây :

E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Như vậy, khối 6 bit đầu tiên gồm các bit 32,1,2,3,4,5 của R; khối thứ hai gồm các bit 4,5,6,7,8,9, ... Sau đó một phép toán thay thế được áp dụng cho dãy 48 bit này bằng cách cộng nó (theo phép cộng loại trừ) với khoá 48 bit. Một phép thay thế khác được sử dụng cho các khối 6 bit để tạo ra các khối 4 bit để kết quả cuối cùng là dãy 32 bit. Ví dụ bảng thay thế cho S_1 là :

S_1																
Số hàng	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	5	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	12	11	15	12	9	7	3	10	5	0
3	15	12	8	2	14	9	1	7	5	11	3	14	10	0	6	13

Để minh hoạ cách sử dụng, giả sử rằng khối 6 bit đầu tiên là 101000. Số nhị phân 10 tạo bởi bit đầu tiên và bit cuối cùng xác định một hàng trong bảng, cụ thể là hàng 2, 4 bit giữa 0100 xác định cột trong bảng, cụ thể là cột 4. Biểu diễn nhị phân 4 bit 1101 của phần tử 13 ở hàng 2 cột 4 trong bảng là giá trị thay thế cho 6 bit này. các phép toán tương tự S_2, S_3,..., S_8 được dùng để chuyển đổi cho các khối 6 bit khác.

Phép hoán vị cuối cùng P được áp dụng cho dãy 32 bit để tạo ra f(R,K):

P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Mười sáu khoá khác nhau dùng trong DES được lấy ra theo một qui định chặt chẽ từ một khoá 64 bit duy nhất. Như vậy người dùng chỉ cần giữ một khoá để mã hoá và giải mã hơn là giữ 16 khoá khác nhau. Thuật toán giải mã cũng tương tự như khi mã hoá, chỉ khác một diều là 16 khoá được dùng theo thứ tự ngược lại.

Việc giải mã được thực hiện ngược lại với 64 bit văn bản mã làm đầu vào cho hoán vị ngược với hoán vị khởi đầu IP^{-1} để tạo ra 64 bit văn bản gốc.

Phương pháp DES được Uỷ ban tiêu chuẩn quốc gia (National Bureau of Standards) Hoa Kỳ đề nghị như là một sơ đồ mã hoá “chuẩn “. Tuy nhiên, người ta còn đang tranh luận liệu khoá 48 bit có đủ dài hay chưa và các phép toán thay thế có đủ độ bảo mật cần thiết hay chưa.

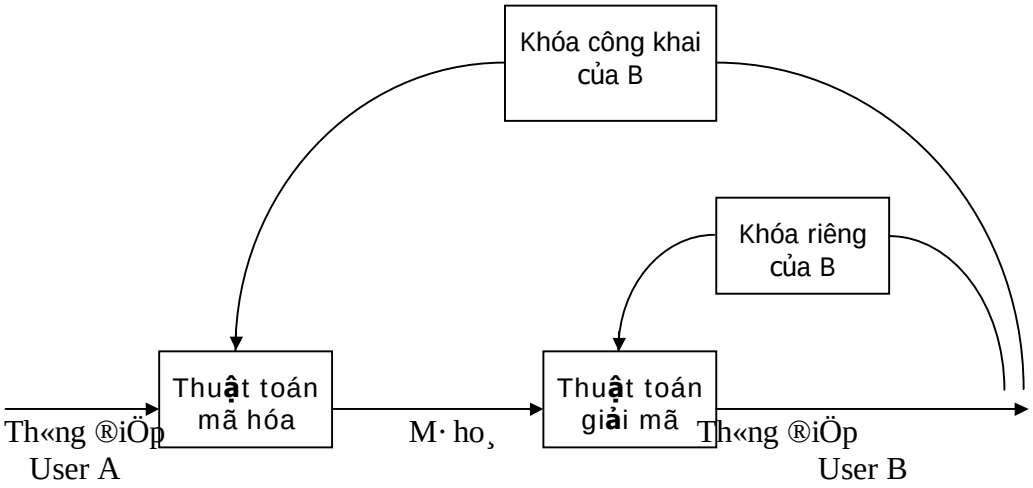
10.2.4 Phương pháp mã hoá khoá công khai

10.2.4.1 Nguyên lí mã hóa công khai

Trong khi thuật toán mã hoá cổ điển dùng một khoá chung cho mã hoá và giải mã thì phương pháp mã hoá bằng khoá công khai sử dụng hai khoá có quan hệ với nhau trong thuật toán để ứng dụng trong mã hoá/giải mã. Các thuật toán này có đặc trưng quan trọng là khó có thể tính toán bằng máy để tìm ra được khoá giải mã nếu chỉ biết được khoá mã hoá và phương pháp mã hoá.

Một số các thuật toán mã hóa công khai (như RSA chẳng hạn) còn có một đặc trưng nữa là khả năng hoán đổi vai trò giữa cặp khoá. Có nghĩa là khi khoá này dùng để mã hoá thì khoá kia dùng để giải mã và ngược lại.

Hình sau mô tả nguyên lí quá trình mã hoá/giải mã bằng khoá công khai :



Hình 10-5. Quá trình mã hoá/giải mã bằng khoá công khai.

Quá trình mã hoá/giải mã như sau:

b. Mỗi hệ thống cuối trong một mạng tạo ra một cặp khoá dùng để mã hoá và giải mã thông tin khi nhận được chúng.

c. Mỗi hệ thống phải có 2 khóa, khóa công khai và khóa bí mật, khoá công khai được công bố lên mạng tại nơi cho phép đăng kí công cộng hoặc đưa vào file. Khoá còn lại phải được giữ bí mật tuyệt đối.

d. Nếu A muốn gửi thông điệp cho B, A sẽ dùng khoá công khai của B trên mạng để mã hoá nó rồi gởi.

e. Khi B nhận được thông điệp của A, B sẽ dùng khoá riêng của mình để giải mã thông điệp nhận được. Không ai có thể giải mã thông điệp được vì chỉ có một mình B biết khoá giải mã.

Thông tin về khoá phải được giữ an toàn tuyệt đối và có thể cập nhật hoặc thay đổi lại khoá cũ. Việc tạo ra các hệ thống bảo vệ và quản lí khoá cũng cần hết sức chặt chẽ.

10.2.4.2 Phương pháp mã hóa RSA

Bản thuyết trình đầu tiên của Diffe và Hellman đưa ra năm 1976 tại hội nghị MIT và gần như ngay lập tức, sự thách thức về vấn đề mã hoá đã tìm được câu trả lời bởi hệ thống mã hoá công khai. Một trong những câu trả lời đầu tiên đưa ra vào năm 1977 bởi Ron Rivest, Adi Shamir và Len Adltôian được công bố vào năm 1978

(gọi tắt là rsa). ý tưởng RSA trở thành gần như độc tôn và được sử dụng rộng rãi trong phương pháp mã hoá bằng khoá công khai.

Giả sử ta có :

Văn bản gốc : $M = M_1 M_2 ... Mk$

Văn bản mã hóa : $C = C_1 C_2 ... Ck$, trong đó $Ci = M_i^E \bmod n$, n là tích 2 số nguyên tố bất kì p và q.

Thuật toán RSA dùng thuyết số để phát triển phương pháp phát sinh một cặp các số nguyên tố - các khoá, thuật toán dựa trên nhận xét: *Có thể dễ dàng sinh ra 2 số nguyên tố lớn và khi nhân chúng với nhau thì rất khó khi muốn phân tích tích của chúng thành thừa số và khó có thể tìm được số còn lại từ số kia*.

Theo một hệ quả của định lí Euler đưa ra: *Cho 2 số nguyên tố p và q và hai số nguyên n và m để n=p.q và 0<m<n, tồn tại một số nguyên duy nhất k sao cho:*

$$(mk^{\phi(n)+1} = mk^{(p-1)(q-1)+1}) \bmod m = n$$

trong đó $\phi(n)$ là hàm Euler với giá trị số nhỏ hơn n và có quan hệ nguyên tố với n, $\phi(n)=(p-1)(q-1)$.

Do đó ta có thể đạt được kết quả mong muốn nếu: $ED = k\phi(n) + 1$

Điều này tương đương với: $ED \bmod \phi(n) = 1$.

Thuật toán RSA được mô tả như sau:

1. Chọn 2 số nguyên tố p, q.
2. Tính tích n = p*q
3. Tính $\phi(n) = (p-1)(q-1)$
4. Chọn E thỏa $USCLN(\phi(n), E) = 1$; *với* $1 < E < \phi(n)$
5. Tìm D thỏa $DE \bmod \phi(n) = 1$.

Khoá công khai là KE = {E,n}, khoá riêng là KD = {D,n} hoặc ngược lại.

Giả sử rằng user A công bố khoá công khai KE lên mạng và user B muốn gửi thông điệp cho user A :

- B sẽ dùng khóa công khai của user A để mã hoá thông điệp của mình bằng công thức $C = ME \bmod n$, rồi gởi nó đi.
- User A sẽ nhận được thông điệp đã mã hoá và giải mã nó bằng khoá riêng của mình bằng công thức $M = CD \bmod n$

Ví dụ: Chọn p =7, q =17

Tính n = p*q = 7*17 = 119

$\phi(n) = (p-1)*(q-1) = 96$

Chọn E thỏa : USCLN(E, 96) = 1. Ta chọn E = 5.

Tìm D thỏa : D*E mod 96 = 1 và D < 96 , suy ra D =77.

Ta được KE = {5,119} , KD = {77,119}.

Giả sử M = 19. Quá trình mã hoá: $C = 19^5 \bmod 119 = 66$.

Quá trình giải mã: $M = 66^{77} \bmod 119 = 19$.

10.2.4.3 Các vấn đề nảy sinh trong thuật toán

1. Vấn đề phức tạp trong tính toán.

Trong quá trình mã hoá và giải mã, thuật toán RSA phát sinh ra các số nguyên rất lớn, cho dù có phép chia modulo n. Rivest, Shamir và Adltôian đề nghị rằng các số p và q phải có độ dài trên 100 chữ số để đảm bảo an toàn gần như tuyệt đối. Như vậy sự luỹ thừa quá lớn và sau đó cho dù có chia modulo n thì kết quả trung gian cũng sẽ khổng lồ và rất dễ dẫn đến tràn số. Ta có thể ứng dụng tính chất của phép chia modulo sau:

((a mod n) * (b mod n)) mod n = (a * b) mod n

Do đó, chúng ta có thể làm giảm kết quả trung gian trong phép chia này đi. Điều này làm cho các phép toán trở nên khả thi hơn.

2. Vấn đề bẻ khoá

Với thuật toán thay thế và hoán vị, về mặt lí thuyết khi độ dài của khoá càng lớn thì mức độ an toàn càng cao, nhưng những người giải mã giàu kinh nghiệm vẫn có thể phân tích tần số xuất hiện của một số kí tự xác định hay tổ hợp của chúng để từ đó suy ra khoá và thực hiện giải mã. Trong thuật toán RSA khoá KE(E,n) là khoá công khai nên ta không cần giữ bí mật, ta chỉ giữ bí mật cho khoá riêng KD(D,n). Vì vậy, để bẻ khóa phải xác định được D từ các giá trị E và n. Theo như cách chọn các số E và D, điều này có thể làm được nếu có thể phân tích n thành tích của hai số nguyên tố. Như vậy tính an toàn của thuật toán RSA phụ thuộc vào sự khó khăn của việc xác định các thừa số nguyên tố của một số nguyên tố lớn. Hiện nay nếu sử dụng thuật toán phân tích thừa số nhanh nhất của Schroeppel thì cũng cần đến : $S = \exp[(\ln n)\ln(\ln n)]^{1/2}$ bước tính toán để phân tích n thành p và q.

Bảng dưới đây hiển thị các thời gian dự đoán của các nhà phân tích, giả sử rằng mỗi phép toán được thực hiện trong 1 micro giây :

Độ dài của khóa	Thời gian
50	4 giờ

75	104 ngày
100	74 năm
200	4.000.000 năm
300	5×10^{15} năm
500	4×10^{25} năm

Phương pháp mã hoá với khoá công khai xem như được bảo đảm vì hiện nay vẫn chưa tìm ra một thuật toán phân tích thừa số nguyên tố có hiệu quả.

10.2.4.4 ứng dụng của mã hoá dữ liệu

Mã hoá dữ liệu có các ưu điểm là an toàn vì ít phụ thuộc vào cấu trúc hệ thống mạng. Ngoài ra mã hoá dữ liệu có tính bảo mật do dữ liệu được mã hoá rồi thì chỉ có những người có quyền mới có thể giải mã để nhận lại được dữ liệu ban đầu. Các phương pháp mã hoá trên có thể áp dụng trong những tình huống sau :

- Phương pháp mã hoá thay thế kết hợp với phương pháp mã hoá hoán vị dùng tạo ra phương pháp mã hoá DES.
- Các dịch vụ e-mail trên mạng Internet hay các mạng cục bộ có thể sử dụng thuật toán RSA để tạo ra một mặt nạ nhận dạng (authentication mask) các thông điệp giữa các cá nhân với nhau. Có nghĩa là chỉ những người nhận được thư gởi cho mình bằng khoá mã hoá của mình thì mới giải mã được thông điệp đó và hoàn toàn không thể (nói theo nguyên tắc) đọc được các thư không phải gởi cho mình.
- Kỹ thuật mã hoá chữ kí số (digital signature) có thể dùng để tạo ra một chữ kí mã hoá dùng để xác định, nhận dạng một đối tượng trong các dịch vụ thương mại, ví dụ như các thẻ tín dụng hoặc các loại visa, cardphone chẳng hạn....
- Thư điện tử e-mail cũng có thể kết hợp thuật toán này với các thuật toán mã hoá khác như DES theo mô hình có thể là:

- Nội dung thư được mã hoá bằng phương pháp DES
- Tạo một chữ ký số và mã hoá bằng khoá RSA
- Khoá DES dùng để giải mã có thể được mã hoá bằng RSA và gửi kèm trong thư luôn mà không cần phải bí mật. Người nhận sẽ dùng khóa riêng của mình để giải mã khoá DES, sau đó giải mã thư nhận được.

10.3 Cơ chế bảo vệ bằng firewall

Vấn đề quan trọng trong việc quản lý các tài nguyên thông tin là cơ chế bảo vệ chống việc truy cập bất hợp pháp trong khi vẫn cho phép người được ủy nhiệm sử dụng những nguồn thông tin mà họ được cấp quyền, và phương pháp chống thất

thoát thông tin được truyền tải trên các mạng truyền dữ liệu công cộng (Public Data Communication Network). Đó chính là yêu cầu của một giải pháp hoặc hệ thống an ninh cho hệ thống mạng hay còn gọi là hệ thống an ninh dữ liệu (Data Security System).

Nhu cầu an ninh hệ thống ngày càng trở nên quan trọng vì nhiều nguyên nhân như các đối thủ luôn tìm cách để nắm được mọi thông tin liên quan, ngày càng nhiều hacker truy cập thông tin từ các mạng nội bộ theo nhiều mục đích khác nhau.

Một giải pháp an ninh cho hệ thống mạng được ứng dụng nhiều đó là bức tường lửa (firewall). Thuật ngữ firewall có nguồn gốc từ một kỹ thuật thiết kế trong xây dựng để ngăn chặn, hạn chế hỏa hoạn. Trong công nghệ mạng thông tin, firewall là một kỹ thuật được tích hợp vào hệ thống mạng để chống lại việc truy cập trái phép nhằm bảo vệ các nguồn thông tin nội bộ cũng như hạn chế sự xâm nhập vào hệ thống của một số thông tin khác không mong muốn.

Về mặt chức năng hệ thống, firewall là một thành phần được đặt giữa hai mạng để kiểm soát tất cả các việc lưu thông và truy cập giữa chúng với nhau, bao gồm:

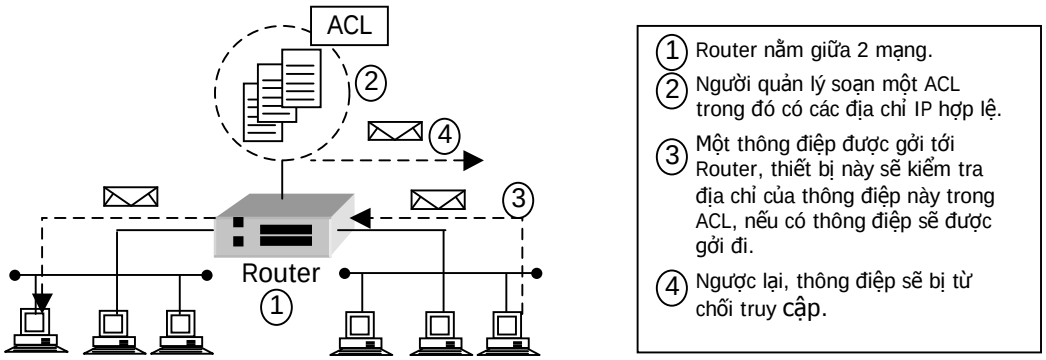
1. Tất cả các trao đổi dữ liệu từ trong ra ngoài và ngược lại phải thực hiện thông qua firewall.
2. Chỉ có những trao đổi nào được phép bởi chế độ an ninh của hệ thống mạng nội bộ (trusted network) mới được quyền lưu thông qua firewall.

Về mặt vật lý, firewall bao gồm:

1. Một hoặc nhiều hệ thống máy chủ kết nối với các bộ định tuyến (router) hoặc có chức năng router.
2. Các phần mềm quản lý an ninh chạy trên các hệ thống máy chủ. Thông thường là các hệ quản trị xác thực (Authentication), cấp quyền (Authorization) và kế toán (Accounting).

Firewall bao gồm phần cứng và/hoặc phần mềm nằm giữa 2 mạng (như mạng nội bộ và mạng Internet), bảo vệ mạng nội bộ bằng cách cấm các người sử dụng truy cập trái phép đến và đồng thời ngăn chặn những thông điệp không được phép gửi đi cho người nhận bên ngoài mạng. Firewall có thể nằm trên bộ dẫn đường hay trên Server. Cơ chế làm việc của Firewall dựa trên việc kiểm tra các gói dữ liệu IP lưu chuyển giữa hai mạng tùy thuộc vào các qui tắc mà người quản trị hệ thống đã xác lập.

Khái quát phương thức làm việc của Firewall như trong hình vẽ sau:



Hình 10-6. Cơ chế hoạt động của Firewall.

10.3.1 Các loại firewall và cơ chế hoạt động

Khi nói đến việc lưu thông dữ liệu giữa các mạng với nhau thông qua firewall thì điều đó có nghĩa rằng firewall hoạt động kết hợp chặt chẽ với giao thức TCP/IP. Vì giao thức này làm việc theo thuật toán chia nhỏ các dữ liệu nhận được từ các ứng dụng trên mạng, hay chính xác hơn là các dịch vụ chạy trên các giao thức (Telnet, SMTP, DSN, SMNP, NFS,...) thành các gói dữ liệu rồi gán cho các gói này những địa chỉ để có thể nhận dạng tái lập lại ở đích cần gửi đến. Do đó các loại firewall cũng liên quan rất nhiều đến các packet và các địa chỉ của chúng.

10.3.1.1 Bộ lọc packet (Packet filtering)

Loại firewall này thực hiện việc kiểm tra số nhận dạng địa chỉ của các packet để cho phép chúng có thể lưu thông qua lại hay không. Các thông số có thể lọc được của một packet như sau:

1. Địa chỉ IP nơi xuất phát (source IP address).
2. Địa chỉ IP nơi nhận (destination IP address).
3. Cổng TCP nơi xuất phát (TCP source port).
4. Cổng TCP nơi nhận (TCP destination port).

Nhờ đó firewall có thể ngăn cản được các kết nối vào những máy chủ hoặc mạng nào đó được xác định, hoặc khóa việc truy cập vào hệ thống nội bộ từ những địa chỉ không cho phép.

Hơn nữa việc kiểm soát các cổng làm cho firewall có khả năng chỉ cho phép một số loại kết nối nhất định vào máy chủ nào đó, hoặc chỉ có những dịch vụ nào đó (Telnet, SMTP, FTP,...) được phép mới chạy được trên hệ thống mạng nội bộ.

10.3.1.2 Cổng ứng dụng (Application gateway)

Đây là một loại firewall được thiết kế để tăng cường chức năng kiểm soát các loại dịch vụ, giao thức được cho phép truy cập vào hệ thống mạng. Cơ chế hoạt

động của nó dựa trên cách thức gọi là Proxy Service (dịch vụ đại diện): một ứng dụng nào đó được quy chiếu đến (hay đại diện bởi) một Proxy Service trong khi các Proxy Service chạy trên các hệ thống máy chủ thì được quy chiếu đến application gateway của firewall. Cơ chế lọc của packet filtering phối hợp kiểm soát với cơ chế "đại diện" của application gateway cung cấp một khả năng an toàn và uyển chuyển hơn.

Ví dụ một hệ thống mạng có chức năng lọc các gói tin ngăn các kết nối bằng Telnet vào hệ thống chỉ trừ một chủ duy nhất -Telnet application gateway là được phép. Một người sử dụng dịch vụ Telnet muốn kết nối vào hệ thống phải thực hiện các bước sau :

1. Thực hiện dịch vụ TELNET đến Telnet application gateway rồi cho biết tên của máy chủ bên trong cần truy cập.
2. Gateway kiểm tra địa chỉ IP nơi xuất phát của người truy cập rồi cho phép hoặc từ chối tùy theo chế độ an ninh của hệ thống.
3. Người truy cập phải vượt qua được hệ thống kiểm tra xác thực.
4. Proxy Service tạo một kết nối Telnet giữa gateway và máy chủ cần truy cập.
5. Proxy Service liên kết lưu thông giữa người truy cập và máy chủ.

Cơ chế hoạt động này có ý nghĩa quan trọng trong việc thiết kế an ninh hệ thống ví dụ như:

1. Che giấu các thông tin: người dùng chỉ có thể nhìn thấy trực tiếp các gateway được phép.
2. Tăng cường kiểm tra truy cập bằng các dịch vụ xác thực (Authentication).
3. Giảm đáng kể giá thành cho việc phát triển các hệ quản trị xác thực vì các hệ thống này được thiết kế chỉ quy chiếu đến application gateway.
4. Giảm thiểu các quy tắc kiểm soát của bộ lọc (Packet filtering). Điều này làm tăng tốc độ hoạt động của firewall.

10.3.1.3 Bộ lọc session thông minh (Smart session filtering)

Cơ chế hoạt động phối hợp giữa bộ lọc packet và cổng ứng dụng như trên cung cấp một chế độ an ninh cao tuy nhiên nó cũng bị vài hạn chế. Vấn đề chính hiện nay là làm sao để cung cấp đủ Proxy Service cho rất nhiều ứng dụng khác nhau đang phát triển ồ ạt. Điều này có nghĩa là nguy cơ, áp lực đối với việc đánh lừa firewall gia tăng lên rất lớn nếu các proxy không kịp đáp ứng.

Trong khi giám sát các packet ở những mức phía trên, nếu như lớp network đòi hỏi nhiều công sức hơn đối với việc lọc các packet đơn giản, thì việc giám sát

các giao dịch lưu thông ở mức mạng (Session) đòi hỏi ít công việc hơn. Cách này cũng loại bỏ được các dịch vụ đặc thù cho từng loại ứng dụng khác nhau.

Nếu kết hợp khả năng ghi nhận thông tin về các session và sử dụng nó để tạo các quy tắc cho bộ lọc thì sẽ có được một bộ lọc thông minh hơn. Đó chính là cơ chế hoạt động của bộ lọc session thông minh.

Vì một session ở mức network được tạo bởi 2 packet lưu thông theo 2 chiều, cho nên nếu thiết kế 2 quy tắc lọc cho 2 chiều này: một để kiểm soát các packet lưu thông từ host phát sinh ra nó đến máy chủ cần tới, một để kiểm soát packet trở về từ máy chủ phát sinh. Một bộ lọc thông minh sẽ nhận biết được rằng packet trở về theo chiều ngược lại nên quy tắc thứ 2 là không cần thiết. Do vậy, cách để tiếp nhận các packet không mong muốn sinh ra từ bên ngoài firewall sẽ khác biệt rất rõ với cách tiếp cận cho các packet do những kết nối được phép (ra bên ngoài). Và như vậy dễ dàng nhận dạng các packet "bất hợp pháp".

10.3.1.4 Firewall hỗn hợp (Hybrid firewall)

Trong thực tế các firewall được sử dụng là sự kết hợp của nhiều kỹ thuật để tạo ra hiệu quả an ninh tối đa. Ví dụ việc để lọt lưới tại các kiểm soát của bộ lọc packet có thể được thực hiện tại bộ lọc session thông minh ở mức ứng dụng. Các giám sát của bộ lọc lại được bọc lót chặt chẽ bởi các dịch vụ proxy của application gateway.

10.3.1.5 Một vài ứng dụng của Firewall

Từ các chế độ hoạt động trên, firewall được ứng dụng nhiều vào hệ thống an ninh dữ liệu. Có 3 yêu cầu chính cho vấn đề an ninh hệ thống theo tiêu chuẩn ISO cho mô hình mạng OSI :

- Quản lý xác thực (Authentication)
- Quản lý cấp quyền (Authorization)
- Quản lý kế toán (Accounting management)
-

f. Ưu điểm của Firewall

Firewall là điểm kiểm tra các kết nối giữa mạng nội bộ và mạng Internet bên ngoài, mọi kết nối đều phải đi qua cửa khẩu này. Đây chính là một bộ lọc an toàn bởi vì có rất nhiều dịch vụ đang hoạt động trên Internet, nếu chúng ta không có một cơ chế kiểm soát chặt chẽ thì các dịch vụ này sẽ tự do mang thông tin tràn vào mạng của chúng ta và ngược lại.

Firewall có thể được sử dụng để ghi nhận lại các hoạt động kết nối với Internet. Bởi vì, mọi hoạt động như vậy đều phải thông qua Firewall nên nó có thể cung cấp thêm chức năng thu thập mọi thông tin về các kết nối xảy ra giữa mạng nội bộ và mạng Internet bên ngoài.

Ta cũng có thể sử dụng Firewall để bảo vệ một máy đơn của người sử dụng.

g. Hạn chế của Firewall

Bên cạnh những mặt tích cực của Firewall kể trên, nó còn có những hạn chế và những việc mà nó không thể thực hiện được như sau:

1. Bên cạnh việc ngăn chặn các người dùng trong mạng nội bộ kết nối ra ngoài khi không được phép thì nó cũng ngăn cản các việc làm tốt của họ.
2. Firewall không thể chống lại các mối nguy hiểm mới, bởi vì chúng nằm ngoài sự kiểm soát của Firewall.
3. Do không kiểm tra trên nội dung của các gói tin, nên Firewall không sử dụng để ngăn ngừa các thông tin xấu trên một dịch vụ đã được cho phép và cũng không thể nhận biết các đoạm mã virus trong các tập tin truyền đi.

10.4 Hệ thống tên miền DNS (Domain Name System)

Địa chỉ Internet 32 bit thỏa mãn yêu cầu kỹ thuật, nhưng phức tạp và khó nhớ đối với người dùng. Giải pháp đưa ra ở đây là dùng những tên gợi nhớ thay cho địa chỉ số là tự nhiên và dễ nhớ đối với người sử dụng. Hơn nữa, dùng tên tin cậy hơn địa chỉ số vì địa chỉ số có thể thay đổi những tên luôn luôn dùng lại được. Do đó nảy sinh vấn đề cách đặt tên và ánh xạ địa chỉ IP với tên.

Trước đây trung tâm thông tin Internet NIC chịu trách nhiệm cấp phát và quản lý tên. Người ta dùng một file có tên host.txt trên Windows hoặc /etc/hosts trên Unix, tập tin này chứa tên của tất cả các mạng, router, host và địa chỉ IP tương ứng với chúng. Các tên được cấp phát không có mối liên hệ gì với nhau. Khi Internet phát triển, giải pháp này trở nên phức tạp không chấp nhận được về mặt quản lý.

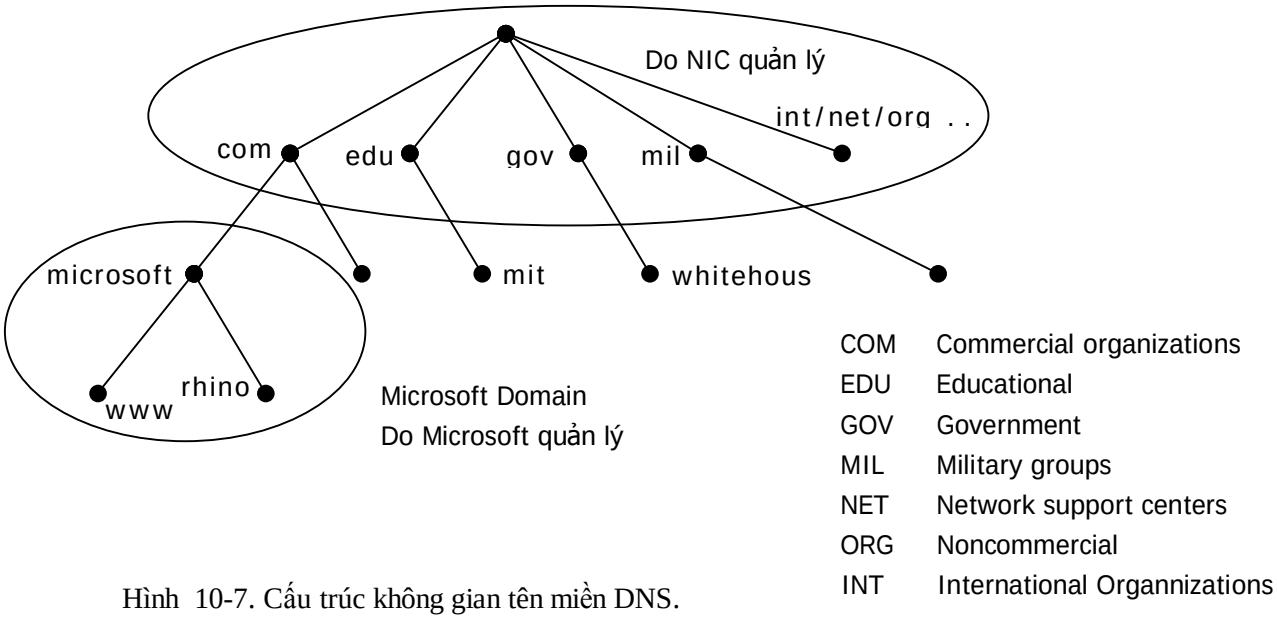
Theo Paul Mockepetris, người thiết kế chính DNS, mục tiêu thiết kế bắt đầu của DNS là để thay thế các tập tin host phức tạp bằng một cơ sở dữ liệu phân tán nhẹ hơn có khả năng cung cấp một *không gian tên thứ bậc, sự quản lý phân tán, có bộ đệm cục bộ (caching), các kiểu dữ liệu mở rộng, kích thước cơ sở dữ liệu không giới hạn và có hiệu năng.*

DNS tương ứng với tầng 7 của mô hình OSI và dùng giao thức UDP hay TCP ở tầng dưới. Việc truy cập DNS thực hiện theo mô hình Client/Server. Hầu hết các hệ thống kết nối Internet đều hỗ trợ DNS. Các đặc tả chính của DNS được định

nghĩa trong các tài liệu RFC 974, 1034, 1035. Dịch vụ cài đặt giao thức DNS phổ biến nhất là BIND (Berkeley Internet Name Domain), được phát triển đầu tiên tại Berkeyley cho hệ điều hành Unix.

DNS gồm 3 thành phần : ***Namespace, các NameServer và Resolver.***

10.4.1 Không gian tên miền DNS



Hình 10-7. Cấu trúc không gian tên miền DNS.

DNS tổ chức không gian tên miền theo cấu trúc cây, trên cùng là gốc, rồi đến các nút cha, nút con... và cuối cùng là các nút lá.

Một máy tính trong mạng sẽ ứng với một nút của cây. Như ở cây trên, máy ở lá www sẽ có địa chỉ hoàn chỉnh là www.microsoft.com. Mỗi nút trên cây biểu diễn một miền (domain) trong hệ thống DNS; mỗi miền lại có một hay nhiều miền con. Tại mỗi miền này đều phải có máy chủ DNS tương ứng quản lý hệ thống tên trong miền đó.

Nút trên cây : Mỗi nút có một tên tương ứng dài từ - đến 63 ký tự dưới 128 trong bảng mã ASCII. Các nút kề nhau không được có cùng tên. Mỗi nút có một tập (có thể rỗng) các *bản ghi tài nguyên* (Resource Record - RR) chứa thông tin đi kèm nút đó. Nhãn rỗng dành riêng cho nút gốc, ký hiệu bằng dấu chấm (.).

Miền con : Được tạo thành từ mỗi nút của không gian tên và các nút bên dưới có thể đi đến được các nút đó.

Vùng : là một phần cây con của cây DNS được quản lý như một thực thể riêng. Vùng có thể bao gồm một miền hay một miền với một số miền con. Các miền con mức thấp hơn của một vùng lại có thể chia thành các vùng rời nhau.

Tên miền của một nút : là dãy các nhãn từ một nút trên cây đến gốc của cây. Các nhãn trong tên miền cách nhau bằng dấu chấm (.). *Tên miền tuyệt đối* kết thúc bằng dấu chấm. Ví dụ "poneria.ISI.EDU.". *Tên miền tương đối* không kết thúc bằng dấu chấm và sẽ được phần mềm cục bộ ghép đầy đủ khi xử lý. Để đơn giản việc cài đặt, độ dài tên miền được giới hạn dưới 255. Một miền là miền con của miền khác nếu tên miền đó chứa tên miền kia. Ví dụ A.B.C.D là miền con của các miền con của các miền B.C.D, C.D, D và miền gốc.

Tên miền đầy đủ là tên các nút từ gốc đến lá của cây nối với nhau và phân cách bằng dấu chấm. Ví dụ : mrp2.widgets.mfg.universal.co.uk

Các miền mức đỉnh : Miền gốc và các miền mức đỉnh của cây DNS do NIC quản lý. Các tên miền mức đỉnh có thể chia ba loại :

- Các miền tổ chức (tên 3 ký tự) : com, edu, gov, . . .
- Các miền địa lý (các mã quốc gia, 2 ký tự) : uk, vn, ca, fr, . . .
- Miền in-addr-arpa : miền đặc biệt dùng để ánh xạ địa chỉ thành tên.

Trách nhiệm quản lý không gian tên DNS dưới mức đỉnh được NIC ủy nhiệm cho các tổ chức khác. Các tổ chức này lại chia không gian tên phía dưới và uỷ nhiệm xuống. Mô hình quản lý phân tán này cho phép DNS được quản lý tự trị bởi các tổ chức tham gia. Cách đặt tên như vậy có tác dụng phân cấp quản lý vùng tên. Các tổ chức có thể tự tạo và quản lý không gian tên riêng của mình trong mạng, không phụ thuộc vào sự cho phép của NIC.

Vấn đề tên và vùng còn được nhiều hãng lớn bổ sung và làm phong phú thêm bằng những giải pháp của riêng họ. Ví dụ Microsoft có WINS - Windows Internet Naming Service, IBM có DDNS - Dynamic Domain Name System.

10.4.1.1 Cú pháp tên miền

Cú pháp cho tên miền sau đây cho phép phù hợp với nhiều ứng dụng như mail, telnet, . . .

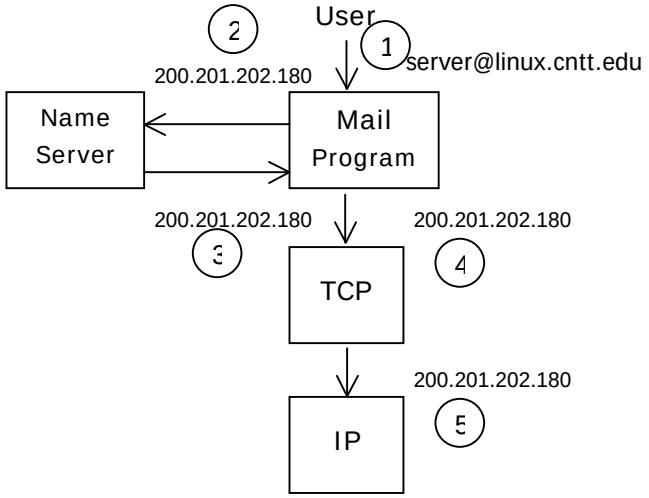
<domain>::=<subdomain>| " "
<subdomain>::=<label>|<subdomain> "." <label>
<label>::=<letter>[[<ldh-str>]<let-dig>]
<ldh-str>::=<let-dig-hyp>|<let-hyp><ldh-str>
<let-dig-hyp>::=<let-dig>| "-"
<let-dig>::=<letter>|<digit>
<letter>::= ký tự từ A-Z, a-z
<digit>::= chữ số 0-9

10.4.2 Máy chủ quản lý tên

Máy chủ quản lý tên (Name Server) là hệ thống chương trình quản lý cấu trúc cây của miền và các tập thông tin đi kèm. Máy chủ tên có thông tin đầy đủ về một số tập con gọi là vùng của không gian tên và các con trỏ đến các nameserver khác để lấy tin về một miền bất kỳ của cây miền. Các máy chủ tên có thông tin đầy đủ về một số phần của cây miền được gọi là có thẩm quyền (authoritative) về các phần đó. Một *vùng* (zone) là một đơn vị thông tin có thẩm quyền của cơ sở dữ liệu DNS. Trong thực tế, các máy chủ tên thường lưu tạm thời trong bộ đệm cấu trúc và thông tin các vùng và thông tin về các vùng khác để tăng hiệu năng. Các máy chủ quản lý tên trong vùng trao đổi thông tin với nhau bằng Zone Transfer Protocol.

10.4.3 Chương trình phân giải tên

Chương trình phân giải tên (Resolver) là các thường trình hệ thống lấy thông tin từ namserver để trả lời yêu cầu của những ứng dụng khách (client). Resolver phải có khả năng truy cập đến ít nhất một nameserver và dùng thông tin từ nameserver đó để trực tiếp trả lời câu hỏi hay để hỏi tiếp đến các nameserver khác. Chương trình người sử dụng có thể truy cập trực tiếp đến resolver, do đó không cần có một giao thức giữa resolver và chương trình người dïng.



Hình 10-8. Quá trình phân giải tên trong thực tế .

10.5 Hệ quản trị mạng

Hệ thống quản trị mạng (Network Management) còn gọi là mô hình Manager/Agent bao gồm các thành phần như sau :

- Hệ quản trị - Manager
- Hệ bị quản trị - Managed system
- Một cơ sở dữ liệu chứa thông tin quản trị và giao thức quản trị mạng.
- Hệ quản trị - Manager

Thực hiện cung cấp giao diện giữa người quản trị mạng và các thiết bị mạng được quản trị, bao gồm các thông tin thể hiện dưới dạng đồ họa, đồ thị, số liệu thống kê, báo cáo. Ví dụ như hiển thị dạng đồ họa bản đồ về topology liên mạng thể hiện các vị trí của các LAN segments, từ đó có thể chọn xem trạng thái hoạt động hiện hành của nó.

10.5.1 Hệ bị quản trị

- Bao gồm tiến trình Agent và các đối thượng quản trị (manager objects).
- Tiến trình Agent thực hiện các thao tác quản trị mạng như đặt các tham số cấu hình và các thống kê hoạt động hiện hành của các router trên một segments cho trước.
- Các đối tượng quản trị bao gồm các trạm làm việc, máy server, hub, kênh truyền.

10.5.2 Cơ sở dữ liệu chứa thông tin quản trị mạng

Được gọi là *cơ sở thông tin quản trị* (Management Information Base - MIB) được lưu trữ tại Server và Client. MIB được tổ chức thành một cấu trúc cây, gọi là SMI (Structure of Management Information). SMI bắt đầu từ gốc root, tiếp theo là các nhánh chứa các đối tượng quản trị được phân loại lôgic.

Kiến trúc quản trị mạng ISO như sau :

1. Quản trị sự cố (Fault Management) : phát hiện, cô lập và khắc phục sự cố.
2. Quản trị kế toán (Accounting Management) : kiểm soát va 2đánh giá việc sử dụng tài nguyên trong mạng
3. Quản trị cấu hình (Configuration Management)
4. Quản trị hiệu năng (Performance Management)
5. Quản trị an toàn (Security Management)

Simple Network Management Protocol (SNMP) được tạo ra ban đầu với mục đích cung cấp phương tiện để điều khiển các router trên mạng. SNMP, mặc dù là một phần trong gia đình giao thức TCP/IP, không phụ thuộc vào IP. SMNP được thiết kế độc lập với giao thức truyền, tuy nhiên phần lớn các hãng đều sản xuất SNMP chạy trên IP.

SNMP thực chất là gồm 3 giao thức cấu tạo thành, tất cả đều được thiết kế để làm việc với mục đích điều hành:

- Management Information Base (MIB): Một cơ sở dữ liệu chứa các thông tin trạng thái.

- Structure and Identification of Management Information (SMI): Một tiêu chuẩn định nghĩa các đầu mục của một MIB.
- Simple Network Management Protocol (SNMP): Phương thức trao đổi thông tin giữa các thiết bị và Server.

10.6 Dịch vụ thư điện tử

Electronic Mail (viết gọn là e-Mail, thư điện tử) là một trong những dịch vụ thông tin phổ biến nhất trên Internet. Dịch vụ e-Mail giúp mọi người có thể trao đổi thông tin với nhau trên mạng Internet. Liên lạc bằng thư điện tử nhanh hơn, thuận tiện hơn và chi phí thấp hơn rất nhiều so với trao đổi thư từ qua đường bưu điện bình thường. Ngoài ra còn cho phép họ gửi cho nhau cả các loại tài liệu như: các văn bản, các báo cáo, các chương trình máy tính, . .. và nhiều thông tin khác nữa.

Mỗi người sử dụng đều có một thư mục lưu trữ thư trên máy Server gọi là Mailbox. Tất cả các địa chỉ mail bao gồm hai phần được ngăn cách nhau bằng 1 ký tự @ (ampersand). Ví dụ : . Tên miền có thể được chia nhiều phần cách nhau bởi dấu chấm (.). Một địa chỉ mail tiêu biểu có các thành phần như sau :

Username @ ServerName. Type of Organization . Country

Cấu trúc của một E-Mail bao gồm các phần như sau :

- ***Phần tiêu đề thư***

Phần này do các MTA (Message Transfer Agent) tạo ra và sử dụng, nó chứa các thông tin để chuyển nhận e-Mail như địa chỉ của nơi nhận, địa chỉ của nơi gửi. Các hệ thống e-Mail cần những thông tin này để chuyển dữ liệu từ máy tính này sang máy tính khác. Cấu tạo phần này gồm nhiều trường (field), mỗi trường là một dòng văn bản ASCII chuẩn 7 bit như sau: <tên trường >: <nội dung của trường>.

Sau đây là một số trường thông tin thông dông:

Trường	Chức năng
DATE	Chỉ ngày giờ nhận mail.
FROM	Chỉ địa chỉ người gởi.
TO	Chỉ địa chỉ người nhận.
CC	Chỉ địa chỉ những người nhận bản copy của mail. Các người nhận thấy được địa chỉ của những người cùng nhận trong nhóm.
BCC	Chỉ địa chỉ những người nhận bản sao chép của bức mail, nhưng từng người không biết những người nào sẽ nhận bức thư này.
REPLY-TO	Chứa các thông tin để người nhận có thể trả lời lại, thường nó chính là địa chỉ người gởi.
MESSAGE-ID	Định danh duy nhất, được sử dụng bởi hệ điều hành.
SUBJECT	Chủ đề của nội dung thư.

Các trường trên là các trường chuẩn do giao thức SMTP quy định, ngoài ra trong phần header cũng có thể có thêm một số trường khác do chương trình e-Mail tạo ra nhằm quản lý các e-Mail riêng. Các trường này được bắt đầu bằng ký tự X- và thông tin theo sau là cũng giống như ta thấy trên một trường chuẩn.

- ***Phần nội dung***

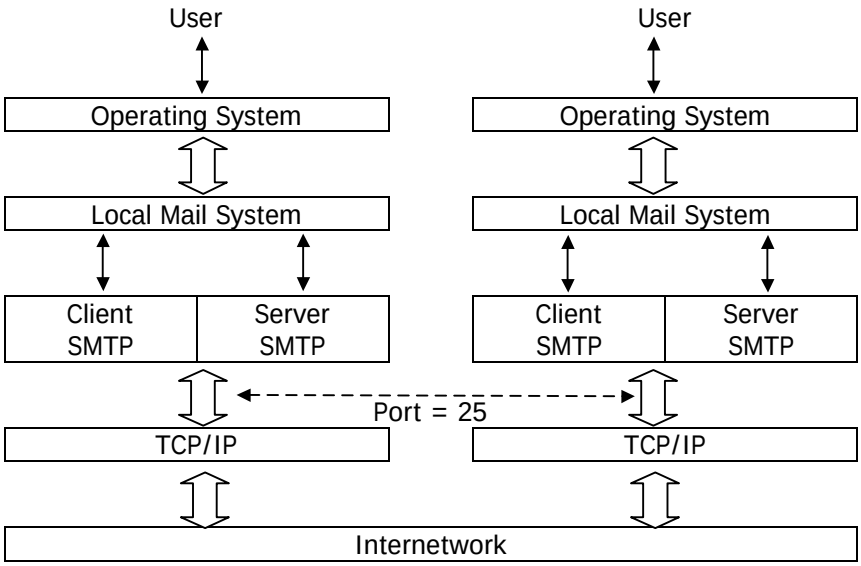
Để phân biệt phần tiêu đề và phần nội dung của e-Mail, người ta qui ước đặt ranh giới là một dòng trắng (chuỗi ký tự "\r\n"). Kết thúc của phần nội dung là chuỗi ký tự "\r\n.\r\n".

Như vậy nội dung bức thư nằm trong khoảng giữa dòng trắng đầu tiên và ký tự kết thúc thư, và trong phần nội dung của bức thư không được phép tồn tại chuỗi ký tự kết thúc thư. Mặt khác do môi trường truyền thông là mạng Internet nên các ký tự cấu thành phần thân của bức thư phải là các ký tự ASCII chuẩn.

10.6.1 Giao thức SMTP

SMTP (Simple Mail Transfer Protocol) là giao thức qui định việc truyền mail chủ yếu dùng trong mạng Internet.

Mối quan hệ giữa SMTP và hệ thống Mail cục bộ như sau:



Hình 10-9. Quan hệ giữa SMTP và hệ thống Mail cục bộ.

Client liên quan đến thư đi, Server liên quan đến nhận thư. Hệ thống thư cục bộ hộp thư (mailbox) cho mỗi user. Mail box có 2 phần: phần cục bộ và phần toàn cục.

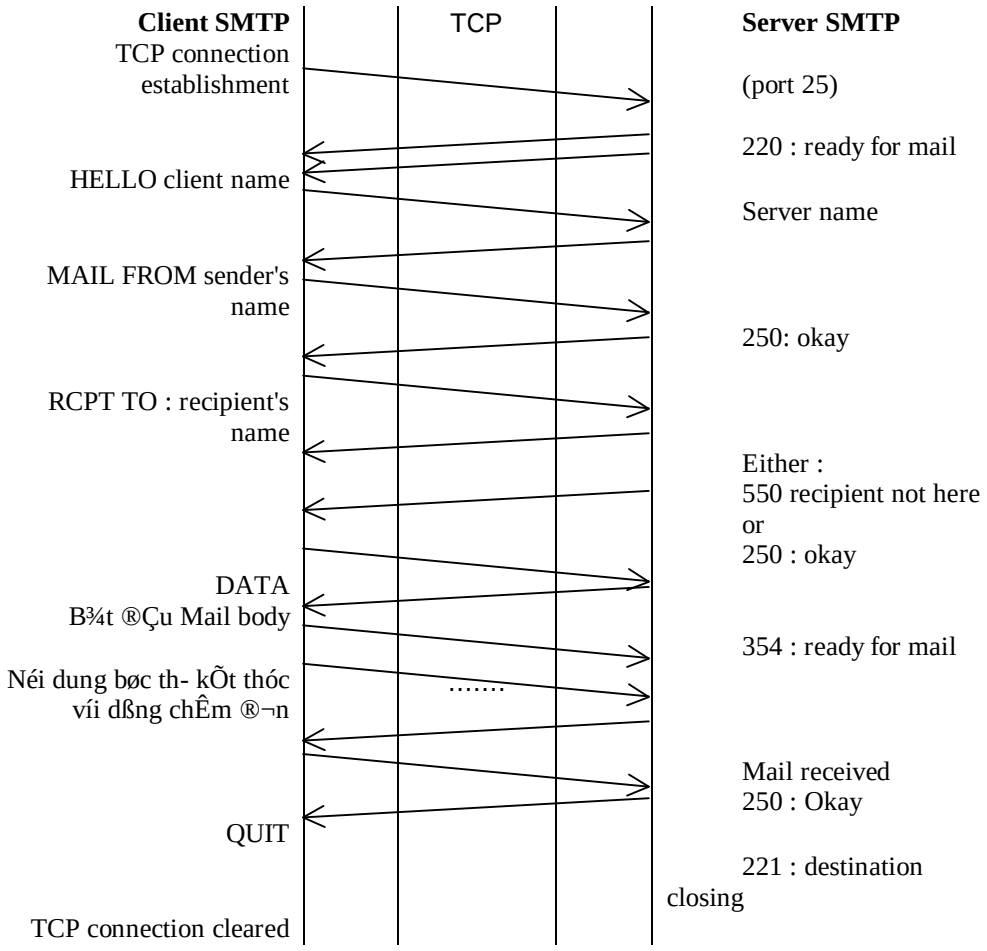
Sau khi tháo bức thư trong khuôn dạng chuẩn, hệ thống mail cục bộ xác định tên người nhận ở hộp thư cục bộ hay phải gửi ra ngoài. để gửi bức thư Client SMTP

phải biết địa chỉ IP của nơi nhận qua DNS và gửi qua cổng địa chỉ SMTP (25) để bắt đầu thiết lập kết nối server SMTP nơi nhận. Khi mối nối đã được thiết lập, Client bắt đầu chuyển bức thư đến Server bởi các lệnh của SMTP. SMTP dùng từ khóa như các lệnh để thực hiện thao tác chuyển giao mail. Một số lệnh chính của SMTP trong phiên làm việc giữa Client MTA và Server MTA như sau :

Lệnh	Tác dụng
HELLO	Xưng danh với SMTP bên nhận, báo cho bên nhận biết bên gởi là ai. SMTP bên gởi gửi lệnh này đầu tiên cho SMTP bên nhận.
MAIL	Khởi động một cuộc giao dịch mail mà mục đích cuối cùng là chuyển giao các mail tới một hay nhiều Mailbox (nơi chứa Mail nhận được) khác nhau.
RCPT	Nói rõ người nhận mail là ai.
DATA	Các dòng sau lệnh DATA là dữ liệu của Mail. Đối với SMTP, chuỗi ký tự "CRLF.CRLF" báo nhận biết kết thúc nội dung bức Mail.
RSET	Bỏ (Reset) cuộc giao dịch hiện tại.
NOOP	Yêu cầu SMTP bên nhận không làm gì ngoài việc trả về câu trả lời OK (dùng để kiểm tra).
QUIT	Yêu cầu SMTP nhận trả lời OK và kết thúc phiên giao dịch hiện tại.
VRFY	Yêu cầu SMTP bên nhận kiểm tra người nhận là đúng, xác nhận các tham số gởi theo dòng lệnh.
SEND	Khởi động một cuộc giao dịch mà mail sẽ được gởi tới một hay nhiều thiết bị đầu cuối chứ không phải mailbox.
SOML	Khởi động một cuộc giao dịch mà mail sẽ được gởi tới một hay nhiều thiết bị đầu cuối hay mailbox.
SAML	Khởi động một cuộc giao dịch mà mail sẽ được gởi tới một hay nhiều thiết bị đầu cuối và mailbox.
HELP	Yêu cầu SMTP bên nhận gửi thông tin giúp đỡ cho SMTP bên phát.
EXPN	Yêu cầu SMTP bên nhận gởi về danh sách những người nhận Mail để có thể mở rộng việc chuyển mail cho các user khác.
TURN	Yêu cầu SMTP bên nhận gưi OK và đổi vai trò trở tránh SMTP gởi.

Bảng 10-1. Các lệnh của giao thức SMTP.

SMTP (trong RFC 821) ban đầu được thiết kế để cho phép các mail server chuyển đổi các mail message. Cơ chế chính được dùng để chuyển đổi các mail là phân đường các message quanh Internet. SMTP hoạt động trên mô hình lưu và truyền trong đó client nắm các message cần để truyền đến server và gửi các lệnh đến server để báo cho server cách xử lí các message. Mail client có thể là một mail server khác, nó có một hay nhiều message phải truyền đến một server khác. Hầu hết các Internet mail client sử dụng SMTP để gửi các message.



Hình 10-10. Cơ chế trao đổi SMTP.

10.6.1.1 Quy tắc làm việc với SMTP

1. Mỗi câu lệnh phân cách tham số theo sau bằng khoảng trắng và kết thúc bằng ký tự CRLF. Mail đi từ SMTP gởi đến một SMTP nhận và đến lượt SMTP nhận trở thành SMTP gởi để gửi mail đi tiếp cho đến khi chúng được giao vào Mailbox của người nhận.
2. Các lệnh SMTP phải diễn ra một cách tuần tự.
3. Việc đánh địa chỉ phải theo cách đánh địa chỉ Internet.

Giao thức SMTP qui định các Server MTA (ở đây là SMTP bên nhận) phải gởi tín hiệu phản hồi ACK sau mỗi lệnh mà nó nhận được từ Client MTA. Mỗi câu trả lời của bên nhận đều mở đầu với một mã số theo sau mới là thông tin dạng text. Mỗi số mở đầu trong mã số có một ý nghĩa khác nhau, nó chỉ ra rằng kết quả thực hiện thao tác là tốt (số 2), thất bại (số 5) hay chưa hoàn thành (số 3).

10.6.1.2 Một số mã phản hồi thông dụng của SMTP

220 Dịch vụ đã sẵn sàng.

221 Đóng kết nối đã được thiết lập.

250 Thao tác do Client MTA yêu cầu đã được hoàn thành.

354 Sẵn sàng nhận nội dung của mail.

550 Thao tác yêu cầu không thực hiện được do không có mailbox trên máy.

.v.v...

10.6.1.3 Phiên giao dịch SMTP

Để hiểu cách dùng một số lệnh chúng ta xem xét qua ví dụ sau: Bên gởi tên Thuận ở máy Sample1 muốn gởi cho Tín , Thức ở máy Sample2, giả sử Thức không có Mailbox tại Sample2.

Bên gởi thực hiện một kết nối đến SMTP Server.

RECEIVER : 220 sample2 Simple Mail Transfer Service Ready
Khi được kết nối qua giao thức TCP/IP, máy nhận trả lời với mã 220
để báo cho máy gởi biết dịch vụ SMTP đã sẵn sàng.
SENDER : HELO sample1
Bên nhận đã sẵn sàng, bên gởi gởi HELLO và xưng tên người gởi.
RECEIVER : 250 sample2
Trả với mã 250 báo cho biết bên nhận đã sẵn sàng.
SENDER : MAIL FROM: <>
Bên gởi dùng lệnh MAIL để khởi động phiên giao dịch. Cú pháp trên cho bên nhận biết
địa chỉ bên gởi (mailbox của bên gởi) để bên nhận gởi thông báo lỗi nếu có về bên gởi.
RECEIVER : 250 OK
Trả lời với mã 250 cho biết đã chấp nhận.
SENDER: RCPT TO: <>
Bên gởi cho biết e-Mail đích
RECEIVER: 250 OK
Trả lời với mã 250 cho biết đã chấp nhận
SENDER : RCPT TO: <>
Muốn gởi cho bao nhiêu người dùng bấy nhiêu lệnh RCPT kèm theo địa chỉ nhận,
bên nhận nếu đúng sẽ trả về mã 250 kèm theo OK.
RECEIVER : 550 No such user here
Báo kèm theo mã 550 cho biết không có mailbox trên địa chỉ trên đối với nơi nhận.
SENDER : DATA
Báo cho bên nhận biết dữ liệu bắt đầu từ sau từ DATA.
RECEIVER : 354 Start mail input; end with <CRLF>.<CRLF>
Mã 354 báo cho biết đã sẵn sàng nhận mail, kết thúc mail với ký tự "CRLF.CRLF".
SENDER : Bắt đầu thân của mail
SENDER : . . .
SENDER : (đến khi kết thúc gởi CRLF.CRLF)
RECEIVER : 250 OK
E-Mail đã được chấp nhận.

SENDER : QUIT

Phát lệnh báo kết thúc phiên giao dịch.

RECEIVER : 221 sample2 Service closing transmission channel

Mã 221 đóng kết nối đã thiết lập

10.6.1.4 Giao thức mở rộng ESMTP

SMTP có một hạn chế gây khó khăn lớn trong việc truyền nhận mail là giới hạn tối đa kích thước nội dung một bức mail chỉ là 128KB. Do vậy người ta đã cải tiến chuẩn SMTP thành một chuẩn mở rộng mới gọi là ESMTP, cho phép tăng giới hạn kích thước của mail lên trên 1MB.

Để biết xem Server MTA có theo chuẩn ESMTP hay không, thay vì dùng lệnh HELO ở đầu một cuộc giao dịch, Client MTA dùng lệnh mới EHLO, nếu Server MTA có trang bị, nó sẽ trả về mã thành công là 250. Ngày nay chuẩn ESMTP đã thay thế chuẩn SMTP ở đa số các hệ thống.

Chẳng hạn để khởi động cuộc giao dịch với kích thước mail lên tới 1MB, sử dụng dòng lệnh sau:

MAIL FROM :<thuan@sample1> SIZE=1000000

10.6.2 MIME

Từ khi MIME (Multipurpose Internet Mail Extention) được đưa ra, kiểu dữ liệu mà user có thể gửi thông qua e-Mail được mở rộng. Ban đầu dữ liệu chỉ ở dạng text. Ngày nay, ta có thể gửi các tài liệu (file *.doc), các file ảnh hay các file âm thanh.

Để có thể phân phát các kiểu dữ liệu này, khuôn dạng các message trên Internet nên được mở rộng. MIME được phát triển cho mục đích này.

10.6.2.1 Cấu trúc message của MIME

MIME không phải cho các ứng dụng e-Mail mới, nhưng cho phép mở rộng khả năng e-Mail trên Internet trong khi vẫn giữ các ứng dụng giao vận và nền tảng hiện tại. Khuôn dạng MIME duy trì các cấu trúc message cơ bản với các phần Header và phần body (tham khảo RFC 822). Ví dụ về khuôn dạng của một tài liệu MIME như sau :

{Dòng này xác định MIME message}

MIME-Version: 1.0

To:

Subject: Book CD

{Dòng này xác định đây là một kiểu messsage hỗn hợp và các phần được phân tách nhau bởi dấu biên}

Content-Type: multipart/mixed; boundary="--------6B9767D111AE"

X-Mozilla-Status: 0001

{Kết thúc phần header}
{Biên đầu tiên, thể hiện phần đầu của message}

--------6B9767D111AE
{Đây là đoạn text, thể hiện các kí tự dạng US-ASCII}
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
{Kết thúc phần header}

Davis,
I am .....
Thanhs,
Davis
{Phần sau là phần đánh dấu biên}

--------6B9767D111AE
{Phần tiếp sau là một file nhị phân}
Content-Type: application/octet-stream
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="Sublic2.doc"
{Phần dưới đây là nội dung file}

0M8.....
{Phần sau đây là biên kết thúc file}

--------6B9767D111AE

10.6.2.2 MIME version header

MIME version header định danh một message như một message MIME, và xác định version của MIME chuẩn để dịch message. Nếu không tìm thấy header, client sẽ đối xử với message theo khuôn dạng chuẩn trong RFC. Phiên bản hiện tại của MIME là 1.0. Cú pháp của MIME header version như sau:

MIME-Version: 1.0

1. Content Type header

Content Type header xác định khuôn dạng file được gán vào trong một đối tượng. Header báo cho MIME cách hiển thị hay thao tác trên thân của message. Content Type Header bao gồm tên của header, theo sau bởi kiểu MIME. Kiểu MIME theo sau hai tên và được cách biệt nhau bởi kí tự slash (/). Tên đầu tiên là tên kiểu và tên thứ hai là một tên phụ. Sau đây là các ví dụ của Content type header:

Content-Type: image/jpeg
Content-Type: image/gif
Content-Type: image/bmp
Content-Type: image/mpeg

Content-Type: application/octet-stream

Ba ví dụ đầu tiên trong phần này, đối tượng là kiểu ảnh (cũng là kiểu nhị phân), kiểu con của nó là jpeg, gif, và bmp. Các file ảnh này được nhúng vào trong các message. Dòng thứ tư trong các ví dụ này đó là một file chương trình.

Các kiểu và kiểu con có thể được thiết lập bởi các tham số. Mỗi tham số bao gồm một tên tham số, theo sau bởi dấu bằng (=) và tiếp theo là giá trị tham số. Các tham số này được tách biệt giữa kiểu và kiểu con, cũng như các tham số khác và được tách biệt nhau bởi dấu chấm phẩy. Ví dụ sau đây thể hiện một tập các tham số:

Content-Type: text/plain; charset=us-ascii

Kiểu đối tượng này báo cho người đọc message rằng các phần theo sau là dạng text và sử dụng các kí tự theo kiểu text.

Header này có thể hoàn toàn tuỳ chọn. Nếu nó không được cung cấp thì message được đối xử như một chuỗi các kí tự ASCII.

2. *Content Transfer Encoding Header*

Content Transfer Encoding Header xác định mô hình mã hoá được sử dụng để nhúng đối tượng vào trong thân của message. Để nhúng một đối tượng nhị phân vào trong một thư điện tử, cần phải chuyển nó sang kiểu dạng ASCII, do vậy nó được biên dịch theo khuôn dạng RFC 822. Ví dụ một cú pháp header dùng để mã hoá nội dung khi truyền là Content-Transfer-Encoding Base64.

Tài liệu MIME định nghĩa 5 kiểu mã hoá, nhưng 3 kiểu mã hoá thể hiện đối tượng không được mã hoá. Mã hoá 7 bit thường được dùng cho các vùng text theo khuôn dạng MIME. Hai kiểu kia mã hoá theo kiểu 8 bit và nhị phân, chỉ được sử dụng khi chuyển thư không phải SMTP, do SMTP chỉ cho phép các kí tự ASCII theo kiểu mã hoá 7 bit. Hai mô hình mã hoá còn lại đó là quoted-printable và base64 để chuyển các các đối tượng từ dạng nhị phân sang kiểu ASCII.

10.6.2.3 Cấu trúc message MIME đa phần

Một trong số các khả năng phổ biến của MIME đó là có một message đa phần. Bằng cách sử dụng message đa phần, ta có thể nhúng cả hình ảnh và âm thanh vào các message text hay xây dựng một ứng dụng về một đối tượng hoạt hình, nó bao gồm một số file cần thiết để chạy ứng dụng.

Cấu trúc message đa phần bao gồm nhiều message kết hợp vào trong thân của một message, mỗi message với thông tin header của nó thể hiện kiểu nội dung mà mô hình mã hoá. Các phần này được tách biệt bởi các dấu biên mà message chính định ra. Để hiểu chi tiết về cấu trúc của một message đa phần, xem RFC 1521.

10.6.2.4 Mã hóa BASE64

Thuật toán mã hoá Base64 được thiết kế để mô tả một chuỗi tuỳ ý các giá trị 8bit mà con người không có khả năng đọc được thành các kí tự ASCII. Thuật toán mã hoá và giải mã đơn giản nhưng dữ liệu mã hoá sẽ lớn hơn dữ liệu nguồn 33%.

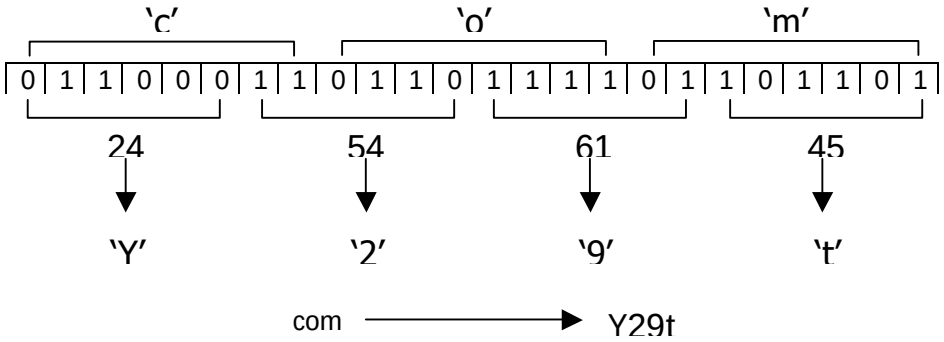
Một tập 65 kí tự US-ASCII được dùng, cho phép 6bits biểu diễn cho các kí tự có thể in được. (Kí tự thứ 65, "=", là một kí tự xử lý đặc biệt)

Tiến trình mã hoá biểu diễn nhóm 24 bits dữ liệu nhập thành 4 kí tự mã hoá ở đầu ra. Tiến trình thực hiện từ trái sang phải, một nhóm 24 bit nhập được kết hợp từ nhóm 3 kí tự 8bits. 24 bits đó được chia làm 4 nhóm kí tự 6bits, mỗi nhóm được dịch thành một kí tự đơn dựa vào bảng mã Base64.

Bảng mã Base64

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
0	A	17	R	34	i	51	z
1	B	18	S	35	j	52	0
2	C	19	T	36	k	53	1
3	D	20	U	37	l	54	2
4	E	21	V	38	m	55	3
5	F	22	W	39	n	56	4
6	G	23	X	40	o	57	5
7	H	24	Y	41	p	58	6
8	I	25	Z	42	q	59	7
9	J	26	a	43	r	60	8
10	K	27	b	44	s	61	9
11	L	28	c	45	t	62	+
12	M	29	d	46	u	63	/
13	N	30	e	47	v		
14	O	31	f	48	w	(pad)	=
15	P	32	g	49	x		
16	Q	33	h	50	y		

Ví dụ sau mô tả tiến trình mã hoá 3 kí tự nhập là "com":



Luồng dữ liệu được mã hoá đầu ra phải được biểu diễn bằng các dòng có độ dài không lớn hơn 76 kí tự. Tất cả các kí tự xuống dòng hay các kí tự khác không có trong bảng mã Base64 đều được phần mềm giải mã bỏ qua.

Khi nhóm bit dòng nhập ít hơn 24 bits (nghĩa là đến cuối của dữ liệu cần mã hoá) thì cần có xử lý đặc biệt. Khi có ít hơn 24 bits dòng nhập thì các bits 0 được thêm vào phía bên phải nhóm bit để được đủ số 24 bits. Khi dòng nhập đã đủ 24bits thì có các khả năng có thể xảy ra:

1. Phần cuối cùng của dữ liệu cần mã hoá là 24 bits thì dữ liệu đầu ra cuối cùng sẽ là 4 kí tự đã mã hoá mà không có kí tự đệm "=".
2. Phần cuối cùng của dữ liệu cần mã hoá chính xác là 8 bits thì dữ liệu đầu ra cuối cùng sẽ là 2 kí tự đã mã hoá kèm theo với 2 kí tự đệm “=” ở cuối.

- Nếu phần cuối cùng của dữ liệu cần mã hoá chính xác là 16 bits thì dữ liệu đầu ra cuối cùng sẽ gồm 3 kí tự đã mã hoá kèm theo với 1 kí tự đệm “=” ở cuối.

Bởi vì các kí tự đệm chỉ được thêm vào cuối của dữ liệu nên khi gặp bất kì một kí tự “=” nào thì hiển nhiên là đã đến vị trí kết thúc của dữ liệu.

10.6.3 Giao thức POP

Người sử dụng có thể gửi thư bằng cách sử dụng SMTP, và có thể nhúng bất kỳ đối tượng nào vào trong message thông qua việc sử dụng khuôn dạng MIME. Tuy nhiên, với SMTP, server để nhận được các message thư phải nối đến client và gửi tất cả các message được phân phát cho client. Do đó, người sử dụng phải đăng ký tên máy dưới dạng tên địa chỉ Internet của người nhận.

SMTP được thiết kế trong trường hợp nhiều user sử dụng tất cả thời gian của họ kết nối đến một vài host và chạy một phiên đầu cuối. Giao thức không được thiết kế cho các tình huống thông dụng hiện nay, trong đó, hầu hết tất cả các user sử dụng e-mail kết nối hạn chế đến mail server đang giữ hộp thư. Người sử dụng phải duy trì các message thư trên server và chuyển nó đến cho client khi client yêu cầu. Đây là một mục đích trong thiết kế của POP.

POP (Post office Protocol) được thiết kế để bù đắp cho SMTP trong phần nhận các message. Những người thiết kế POP không gộp các chức năng gửi message và cho rằng SMTP tiếp tục được sử dụng để thực hiện các chức năng đó. Với giao thức POP, máy tính nhận khởi tạo kết nối. Máy nhận kết nối đến mail server, login và nhận bất kỳ một message nào đang chờ. Do vậy mà máy gửi không cần biết gì về máy nhận trừ khi nó sử dụng login và password để đăng nhập. Ngày nay, hầu hết tất cả các mail client trên Internet mà bạn có thể sử dụng để kết hợp cả SMTP và POP.

10.6.3.1 Mô hình thông tin POP

Trong mô hình lưu và phát, server mail cục bộ lưu các message đến khi các client nhận nó. POP client kết nối với server trên cổng 110 của TCP. Để đăng nhập vào server, user sử dụng định danh (ID) và password. Sau khi đăng nhập thành công vào server, client có thể yêu cầu server về các message mới đang sẵn sàng, lấy bất kỳ message nào mà server đang gửi hay xoá đi một message nào đó trên server.

Mô hình thông tin POP sử dụng 3 trạng thái giao tác để cung cấp chức năng này đến POP client:

- Trạng thái đặc quyền : Server kiểm tra quyền truy nhập của client (ID và password).
- Trạng thái giao tác : Client có thể nhận hay xoá các message.
- Trạng thái cập nhật : Trạng thái này được chuyển đến ngay sau khi client tạo ra lệnh QUIT.

Trạng thái cập nhật là trạng thái cho phép thao tác trên các message. Khi client đang ở trên trạng thái giao tác, bạn có thể tạo ra lệnh reset để huỷ bỏ tất cả các thao tác xóa trước đó (undo).

10.6.3.2 Chuẩn POP3

Giao thức POP3 được cải tiến từ giao thức POP. Nhiệm vụ của giao thức POP3 là lấy mail từ mailbox về khi nào người nhận muốn.

Đặc điểm của hệ thống dùng POP là cho phép người sử dụng login vào POP Server và nhận các mail từ mailbox của mình mà không cần phải login vào mạng mặc dù các mailbox thường nằm ở các Mail Server nằm trong mạng (thông thường muốn thâm nhập mạng ta phải có một account trên mạng và phải cung cấp Password khi đăng nhập vào mạng). Người sử dụng có thể truy xuất POP Server từ bất cứ một hệ thống nào trên mạng Internet, từ bất cứ UA nào dùng giao thức POP.

POP3 định nghĩa 3 giai đoạn tạo thành POP Session : Giai đoạn 1 là giai đoạn xác định tính hợp pháp của người nhận mail (Authorization); giai đoạn 2 là giai đoạn giao dịch giữa PC và POP Server (Transaction) và giai đoạn 3 là giai đoạn cập nhật thông tin (Update).

Sau khi thiết lập kết nối với Server, giai đoạn đầu Client sẽ cho Server biết nó là ai. Nếu Client hợp pháp POP Server sẽ mở Mailbox và bắt đầu chuyển sang giai đoạn giao dịch. Giai đoạn giao dịch, chương trình Client sẽ yêu cầu POP3 Server cung cấp các thông tin như danh sách mail..v..v..hay yêu cầu gởi về cho nó một bức mail xác định nào đó. Giai đoạn cuối cùng sẽ cập nhật và đóng kết hiện hành.

Các lệnh thông dụng của giao thức POP3 :

Lệnh	ý nghĩa
User	Cho biết tên của user cho POP Server
Pass	Yêu cầu một Password cho người sử dụng trên Server
Quit	Đóng kết nối TCT đã được thiết lập trước đó
Stat	POP Server trả về số lượng Mail có trong mailbox của người sử dụng cùng kích thước chúng
List	Trả về các ID và size của các Message
Retr	Nhận một Message từ Mailbox (yêu cầu tham số là ID của mail cần nhận)
Dele	Đánh dấu một Message để xóa (yêu cầu tham số là ID của mail cần xóa)
Noop	POP Server trả về +OK nhưng không làm gì cả
Last	Yêu cầu POP Server trả về số Message đã truy nhập
Top	Liệt kê Header của Mail
Rset	Hủy đánh dấu trên Message bị đánh dấu để xóa

POP3 chỉ định nghĩa 2 loại trả lời cho mỗi câu lệnh là : +OK để chỉ thao tác hoàn thành tốt và - ERR để báo có lỗi. Ví dụ cách dùng một số lệnh của POP3 như sau (các hàng sau dấu chấm phẩy để chú thích lệnh).

Giai đoạn 1 : Nhận dạng user

```
CLIENT : USER user01     ; cho biết tên user là user01
SERVER : +OK             ; báo thành công
CLIENT : PASS abc        ; cho biết password là abc
SERVER : +OK user01's    ; maildrop has 2 messages ( 520 octets)
```

Giai đoạn 2 : Trao đổi

```
CLIENT : STAT            ; số mail có trong mailbox
SERVER : +OK 2 520       ; Có 2 mail với tổng kích thước là 520
CLIENT : LIST            ; Liệt kê các ID và kích thước các mail
SERVER : +OK 2 message ( 520 octets )
SERVER : 1 110           ; mail thứ 1 kích thước 110
SERVER : 2 410           ; mail thứ 2 kích thước 410
CLIENT : LIST 1  ; Cho thông tin về mail có ID là 1
SERVER : +OK 1 110
CLIENT : LIST 4
SERVER : -ERR            no such message, only 2 message in maildrop
. . .v. . .v . . .
```

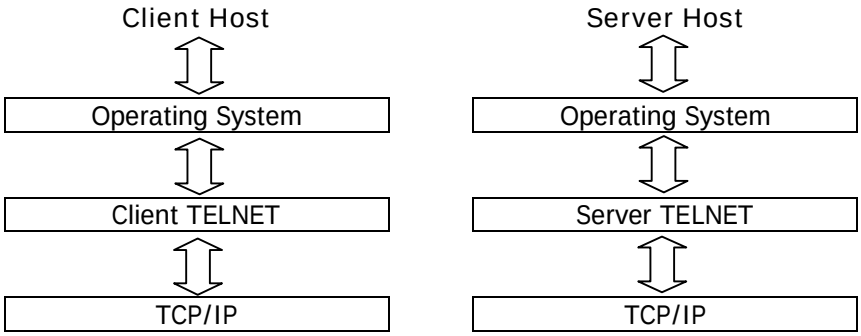
Giai đoạn 3 : Kết thúc

```
CLIENT : QUIT ; đóng kết nối TCP hiện hành
SERVER : +OK dhbk POP3 server signing off
```

Chú ý rằng các message bị đánh dấu để xóa bằng lệnh DELE thực sự chưa bị xóa ngay để nếu sau đó ta có thể dùng lệnh phục hồi không xóa bằng lệnh RSET,

chúng chỉ thực sự bị xóa bỏ khỏi maildrop khi bước vào giai đoạn Update (khi gởi lệnh QUIT).

10.7 Dịch vụ truy cập từ xa - TELNET



Hình 10-11. Phương thức truy nhập từ xa Telnet.

Chương trình Telnet (TELecommunication NETwork) cho phép truy cập từ xa hoặc có các thiết bị ảo thông qua mạng (điều này có nghĩa là bình thường thì bạn không thể có được thiết bị này nhưng nay nhờ có dịch vụ Telnet, bạn có thể truy cập và dùng được các thiết bị đầu cuối do đó gọi là các thiết bị đầu cuối ảo). Nói cách khác, một user A có thể truy cập vào một máy B ở bất cứ nơi nào trong mạng và làm việc với máy đó giống như đang ngồi trước máy đó. Dịch vụ Telnet được cung cấp qua cổng số 23 của TCP/IP. Khái niệm Telnet để chỉ cả *dịch vụ và giao thức* cung cấp các dịch vụ truy cập từ xa này.

Giao thức Telnet dùng một khái niệm Network ***Virtual Terminal*** (NVT), để định nghĩa kết nối Telnet cho cả hai phía. Mỗi đầu của kết nối (mỗi NVT) có một bàn phím và một máy in logic. Máy in logic có thể hiển thị các kí tự và bàn phím logic có thể tạo các kí tự. Máy in logic thường là một màn hình của thiết bị đầu cuối, trong khi đó bàn phím logic thường là bàn phím của người dùng

Khi một kết nối Telnet được thiết lập, Telnetd (hay bất kỳ một chương trình nào khác mà làm việc như là Telnet server) bắt đầu quá trình chạy một số các ứng dụng. Mỗi phím được ấn sẽ phải qua Telnet, Telnetd, và các ứng dụng được dùng trong quá trình thực hiện một phiên làm việc của kết nối Telnet.

Người sử dụng đưa vào lệnh và số liệu, chương trình Telnet ở máy khách (client Telnet) sẽ chuyển lệnh và số liệu đến chương trình Telnet trên máy chủ (server telnet) tương ứng. Server telnet xử lý và gửi kết quả trở lại cho Client Telnet.

10.7.1.1 Các lệnh của Telnet

Hai hệ thống Telnet Client/Server liên lạc với nhau bằng những lệnh gồm những *kí tự đơn* hay một *chuỗi kí tự*, nó được mã hoá trong dạng chuẩn NVT (Network Virtual Terminal - Mạng đầu cuối ảo).

Khi một kết nối Telnet được thiết lập, một số dịch vụ có thể sẵn sàng để lựa chọn. Giá trị của chúng có thể thay đổi trong một phiên làm việc Telnet (*Telnet Session*) nếu cả hai phía của kết nối đồng ý sự thay đổi đó. (Có thể xảy ra trường hợp một đầu của kết nối Telnet không thể cho phép hay không cho phép một dịch vụ trong quá trình kết nối Telnet diễn ra do sự cho quyền của nhà quản lý hoặc các thiết lập nguồn (Source settings)). Có bốn giao thức Telnet được dùng để Để nghị (offer), Từ chối (refuse), Yêu cầu (request) và Ngăn chặn (prevent) các dịch vụ, đó là các động từ: WILL, WON'T, DO và DON'T. Các động từ trên được thiết kế đi với nhau theo từng cặp (WILL/WON'T và DO/DON'T).

Lệnh	**Mã thập phân**	**ý nghĩa**
IAC	255	Nhận biết byte tiếp theo là lệnh
NOP	241	Không điều khiển
EC	247	Xóa ký tự (Erase character)
EL	248	Xóa dòng (Erase line)
GA	249	Về đầu (Go ahead)
AYT	246	Are you there
IP	244	Quá trình ngắt (Interrupt process)
AO	245	Xóa bỏ đầu ra (Abort output)
BRK	243	Dừng (break output)
DMARK	242	Phục hồi đầu ra (Resume output)
SB	250	Bắt đầu trao đổi (Start potion request)
SE	240	Kết thúc (End)
WILL	251	Thỏa thuận/Yêu cầu (Agreement/request option)
WONT	252	Từ chối (Refuse option request)
DO	253	Tiếp nhận yêu cầu (Accept request option)
DON'T	254	Từ chối tiếp nhận yêu cầu

- Các hàm chức năng khác :

Tên	**Mã**	**ý nghĩa**
Transmit binary	0	Yêu cầu/T.nhận trao đổi số nhị phân 8 bit
Echo	1	Ký tự phản hồi (Echo character receiving back to sender)
Status	5	Trạng thái (Request/reply status of receiving TELNET)
Timing mark	6	Đánh dấu thời gian.
Terminal type	24	Loại yêu cầu/trả lời của thiết bị đầu cuối.
Line mode	34	Gửi dòng ký tự

Ví dụ các dòng lệnh tiêu biểu như sau :

IAC, SB, WILL, 'O', SE : Yêu cầu bên nhận nhận số nhị phân 8 bit
IAC, SB, DO, 'O', SE : Hệ truy nhập từ xa nhận trả lời tiếp nhận
IAC, SB, DON'T, 'O', SE : Hoặc từ chối
IAC, SB, DO, 'O', SE : Bên nhận yêu cầu
IAC, SB, WILL, 'O', SE : Bên gửi thỏa thuận
IAC, SB, WON'T, 'O', SE : Hoặc từ chối

- Làm việc với Telnet

- Truy nhập vào mạng TCP/IP từ máy trạm
- Gõ lệnh : telnet <ĐịachỉIP hoặc tênmáyServer>
- Thao tác trên màn hình Telnet.

10.7.2 Dịch vụ truyền tập tin FTP

Giao thức truyền tập tin FTP (File Transfer Protocol) cho phép truyền các tập tin giữa hai máy tính, quản lý các thư mục và truy cập vào thư tín điện tử. FTP không được thiết kế để truy cập vào một máy khác và chạy các chương trình ở máy đó. FTP giúp người sử dụng truy cập file và thư mục trên một máy chủ ở xa và thực hiện những thao tác trên thư mục như sau :

- Liệt kê các file trên một thư mục cục bộ hay ở xa.
- Đổi tên và xóa tập tin (nếu có quyền).
- Truyền file đi hay về từ trạm và máy ở xa (download/upload).

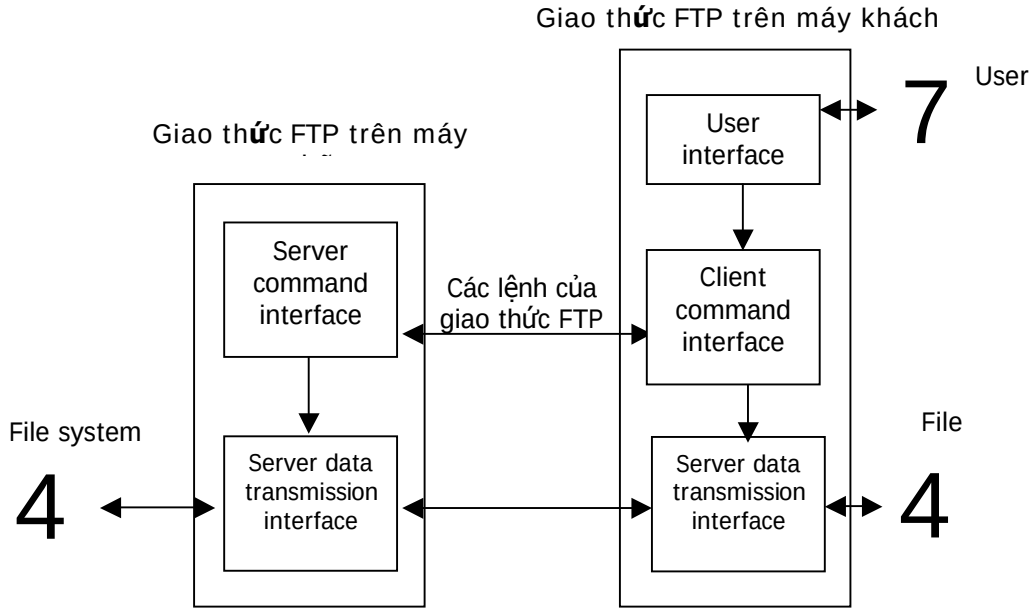
FTP dùng hai kênh TCP, với số hiệu cổng 20 là ***kênh dữ liệu***, và số hiệu cổng 21 là ***kênh lệnh*** (*command channel*). FTP khác các ứng dụng khác của TCP/IP ở là FTP quản lý tất cả việc truyền các tập tin bằng foreground thay vì background. Nói cách khác, FTP không dùng các hàng đợi hay các tiến trình kiểu ống (spooler) do đó bạn có thể quan sát quá trình truyền tập tin trong thời gian thực. Bằng cách dùng TCP, FTP loại trừ được việc quản lý kết nối và độ tin cậy, bởi vì FTP có thể dựa trên TCP để thực hiện các chức năng này một cách chính xác.

Kết nối đầu tiên, kênh lệnh, được khởi tạo thông qua FTP client. Client kết nối với server dựa trên cổng 21 của TCP, cung cấp cho server tên (login) và password và sau đó tiến đến các phiên FTP. Nếu client tạo ra một lệnh yêu cầu một dòng trả lời từ server, kênh lệnh sẽ truyền trả lời này.

Khi client gởi một yêu cầu có nhiều hơn một trả lời để gửi hay nhận dữ liệu, kênh thứ hai được đặt vào hoạt động. Để thiết lập kết nối thứ hai, bạn có 3 tuỳ chọn. Mặc định, server khởi tạo kết nối thứ 2 thông qua cổng 20 của TCP và kết nối đến một socket thứ hai trên client, sử dụng cùng một địa chỉ và cổng như trong kết nối thứ nhất trên client. Tuy nhiên, client có thể chỉ định một địa chỉ khác hay một cổng khác để truyền dữ liệu, trong trường hợp này, server cố gắng kết nối đến client

thông qua việc sử dụng một địa chỉ mới. Tuỳ chọn thứ 3 là client khởi tạo một kết nối truyền dữ liệu là báo cho server chuyển sang chế độ thụ động, server trả lời một địa chỉ và số hiệu cổng để truyền dữ liệu.

Ngay sau khi truyền dữ liệu kết thúc, kết nối để truyền dữ liệu được đóng lại. Kết nối này được mở lại khi client tạo ra một lệnh yêu cầu truyền dữ liÖu.



Hình 10-12. Mô hình giao tiếp FTP.

- FTP hoạt động theo mô hình Client/Server bao gồm thành phần chính :

  + Đơn vị trao đổi dữ liệu (Data Transmission interface)/

  + Đơn vị nhận biết lệnh (Command interface)

10.7.2.1 Chế độ truyền dẫn

Có 3 chế độ được dùng để truyền dữ liệu giữa hai hệ thống. Chế độ đầu tiên là ngầm định nhưng 2 chế độ kia truyền hiệu quả hơn và có thể phục hồi.

- ***Truyền theo dòng:*** đây là chế độ truyền ngầm định, gửi một file dưới dạng một chuỗi các byte; FTP server và client không định dạng file đó. File nguồn không có cách gì để báo hết nội dung truyền, do vậy vấn đề kết thúc file được qui định bằng đóng kết nối dữ liệu.
- ***Truyền theo khối:*** chia file thành các khối, và mỗi khối có thêm các byte điều khiển (header). Trong header có một trường xác định số lượng byte trong khối, trường mô tả mã, nó có thể định đó là khối đặc biệt, kết thúc trong quá trình truyền. Chế độ truyền này cho phép phục hồi khi bị ngắt trong quá trình truyền file thông qua việc báo truyền lại một khối chỉ định trong trường count của header.

- ***Chế độ truyền nén:*** nén file để truyền thông qua việc sử dụng thuật toán mã hoá mã run-length. Thuật toán nhằm làm giảm các byte lặp lại vào trong hai byte kế tiếp. Byte đầu tiên cho biết byte theo sau là nén và số lần nó được lặp lại. Để thể hiện nén, bit đầu tiên của byte điều khiển được thiết lập 1. Nếu bit này là 0, nó cho biết byte theo sau không phải là byte nén. Phần còn lại của byte điều khiển xác định số lượng các byte không nén theo sau. Do vậy, hiệu quả khi nén các kí tự lặp lại đó là không làm mất đi các kí tự không nén.

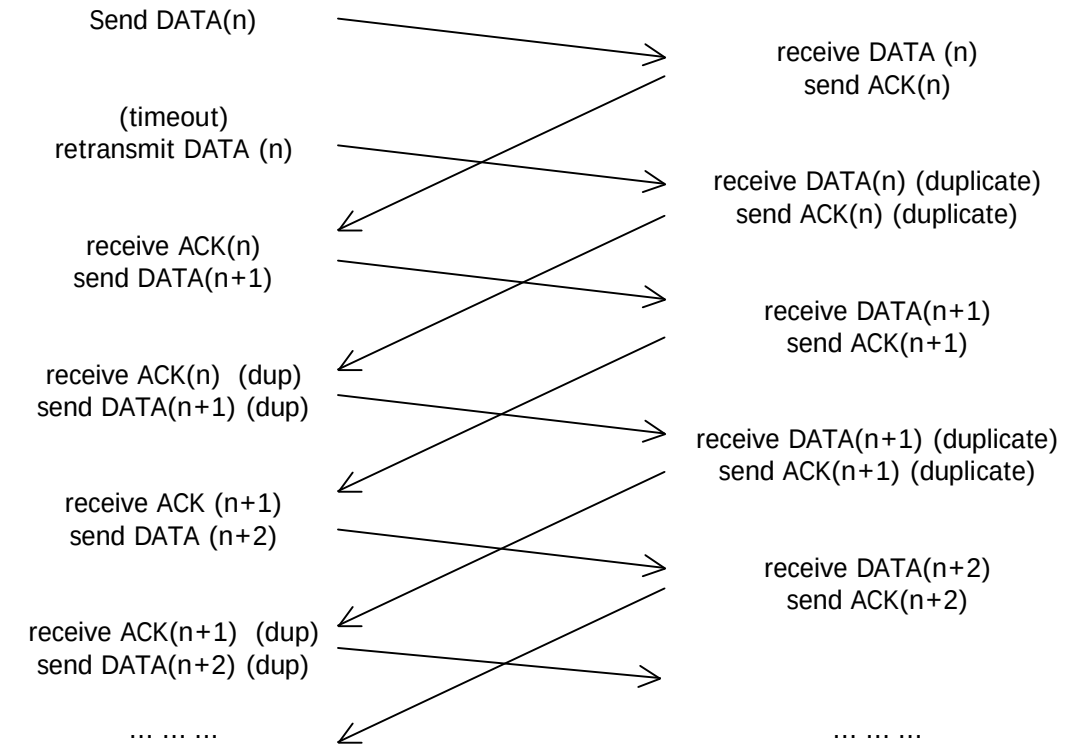
Để bắt đầu, client gửi yêu cầu **read** hay **write**. Gói tin trao đổi có độ dài đến 512 bytes. Mỗi block số liệu có đánh số và phải được biên nhận để gửi tiếp hay phát lại. Để tránh phát trùng lặp khi hết thời hạn, phát lại bản tin vừa phát và khi nhận ACK (n) trùng lặp thì không phát gì .

10.7.2.2 Dạng bản tin FTP

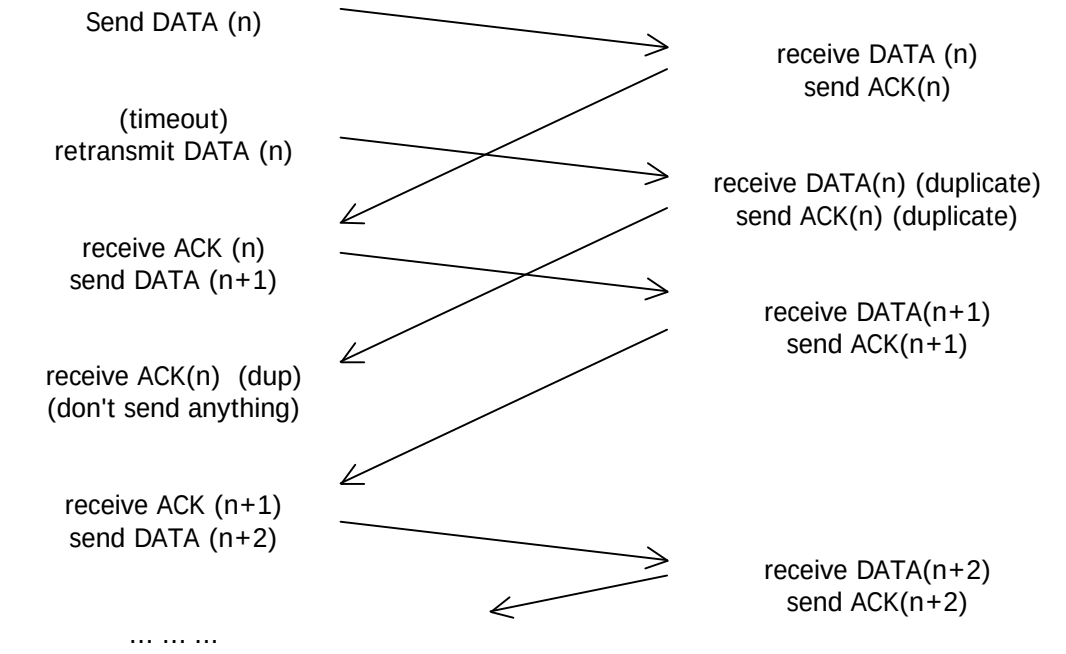
Read request (RRQ)	opcode	String	EOs	String	EOs
	01	File name	0	mode	0
	2 bytes	n bytes	1 byte	n bytes	1 byte
Write request (WRQ)	opcode	String	EOs	String	EOs
	02	File name	0	mode	0
	2 bytes	n bytes	1 byte	n bytes	1 byte
DATA	opcode	Block#	Data		
	03				
	2 bytes	2 bytes	n bytes, 0≤n≤512		
Acknowled-gement (ACK)	opcode	Block#			
	04				
	2 bytes	2 bytes			
Read request (RRQ)	opcode		String	EOs	
	05	Errorcode	Err String	0	(EOs : End of String)
	2 bytes	2 bytes	n bytes	1 byte	

Hình 10-13. Khuôn dạng bản tin FTP.

Ví dụ: Quá trình phát lặp :



+ Tr¸nh ph¸t lÆp



Hình 10-14. Quá trình phát lặp bản tin FTP.

10.7.2.3 Quá trình làm việc FTP

1. Truy nhập vào mạng TCP/IP từ máy trạm.
2. Gõ lệnh : ftp <Địa_chỉ_máy_Server>.
3. Làm việc với FTP.

Khi một kết nối FTP được thiết lập, thực hiện các bước như sau:

- Duyệt tên và mật khẩu (ID) của người dùng.
- Xác định thư mục bắt đầu làm việc.
- Định nghĩa chế độ truyền tập tin.
- Cho phép các lệnh của người dùng.
- Huỷ kết nối.

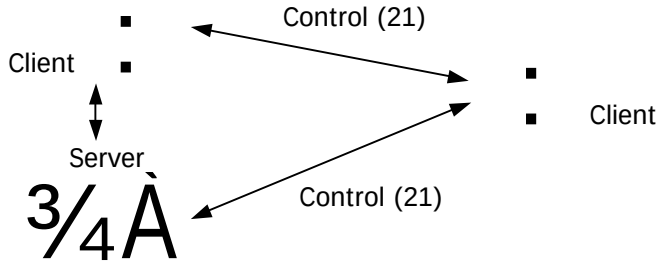
Lệnh FTP	**Mô tả**
ascii	Chuyển sang chế độ truyền ascii
bell	âm thanh của chương trình sau khi truyền mỗi tập tin
binary	Chuyển sang chế độ truyền nhị phân
cd *directory*	Chuyển đổi thư mục hiện hành trên server
cdup	Lùi thư mục hiện hành về một cấp trước đó
close	Huỷ kết nối
delete *filename*	Xoá một tập tin trên server
dir *directory*	Hiển thị thư mục *directory* của server
get *filename*	Truyền tập tin trên server về máy cục bộ
hash	Hiển thị/làm mất dấu # cho mỗi khối các ký tự đã truyền được
help	Hiển thị các trợ giúp
lcd *directory*	Chuyển đổi thư mục hiện hành trên máy cục bộ
ls *directory*	Xem danh sách các tập tin trong thư mục directory trên Server
mdelete *files*	Xóa nhiều tập tin trên máy Server
mdir *directories*	Liệt kê các tập tin trong nhiều thư mục trên máy Server
mget *files*	Lấy một số file trên Server về thư mục hiện hành của máy cục bộ
mkdir *directory*	Tạo thư mục directory trên máy Server
mput *files*	Gửi một số tập tin từ máy cục bộ lên máy Server
open *host*	Kết nối với Server host từ xa
put *filename*	Truyền tập tin từ máy cục bộ lên máy Server
pwd	Hiển thị thư mục hiện thời của server
status	Hiển thị trạng thái của ftp
rename *file1 file2*	Đổi tên *file1* trên máy Server thành *file2*
quote	Cung cấp một lệnh FTP một cách trực tiếp
quit	Chấm dứt kết nối và thoát khỏi ftp
?	Hiển thị danh sách lệnh

Đề truyền một tập tin từ *thư mục hiện hành* trên máy Client đến máy Server bạn dùng lệnh *put*, ngược lại, muốn tải tập tin từ máy Server về máy Client, bạn dùng lệnh *get*. Cú pháp như sau :

```
ftp>put   local_file   remote_file
ftp>get   remote_file   local_file
```

Khi truy cập vào hệ thống, nếu chưa có account, người sử dụng có thể sử dụng một login name đặc biệt là *anonymous* để truy cập vào hệ thống. Acount này không có mật khẩu.

FTP cho phép truyền các tập tin thông qua máy thứ 3, máy này nằm giữa client và server. Thủ tục này được gọi là truyền tay ba điều này cần thiết để có được có được sự cho phép chính xác để truy cập vào máy ở xa. Hình sau mô tả sơ đồ của thủ tục này :



Hình 10-15. Truyền các tập tin thông qua máy thứ 3.

10.7.2.4 Khuôn dạng dữ liệu

Khi truyền dữ liệu giữa hai hệ thống, có thể sử dụng 4 kiểu dữ liệu để truyền. Trong số các kiểu dữ liệu này thì có 2 kiểu dữ liệu hay được sử dụng nhất hiện nay, hai kiểu khác vẫn được hỗ trợ nhưng ít được sử dụng. Các hệ thống ở cả hai đầu trong quá trình đàm thoại FTP phải hỗ trợ tất cả các kiểu dữ liệu sau đây:

- Kiểu ASCII, đây là kiểu mặc định được dùng trong các phiên FTP. Nó được dùng để truyền các file text. Nếu bạn cố truyền các file nhị phân mà bạn không thay đổi mode thì bạn cũng nhận được kết quả ở dạng text, do vậy nội dung của file đã bị thay đổi.
- EBCDIC được sử dụng để truyền các file giữa giữa các host, sử dụng EBCDIC như một tập các kí tự bên trong của nó. Về mặt kỹ thuật thì kiểu dữ liệu ASCII và EBCDIC là giống nhau, chỉ khác một điều là tập các kí tự mà nó sử dụng
- Kiểu nhị phân là kiểu được sử dụng để truyền các file nhị phân như các file ảnh và các file chương trình (các file ZIP và các file DOC). Việc truyền các file này dưới dạng một chuỗi các byte, kiểu dữ liệu này không quan tâm đến môi trường của máy đích và cấu trúc từ. Tất cả các cài đặt FTP nên hỗ trợ kiểu truyền dữ liệu này cũng như kiểu ASCII.
- Kiểu dữ liệu cục bộ. Kiểu dữ liệu này dựa trên byte, xác định cho các host cục bộ. Khuôn dạng phải khả dụng với các hệ thống khác để cấu trúc lại dữ liệu dựa vào dựa trên khuôn dạng ban đầu.

Kiểu dữ liệu ASCII và EBCDIC có thể có tham số tuỳ chọn thứ hai để xác định các ràng buộc dữ liệu. Khi được sử dụng, tham số này là một tuỳ chọn được thêm vào để xác định kiểu dữ liệu. Các ràng buộc định dạng phụ thuộc vào việc sử dụng của file được truyền. Liệu một file có thể được in, xem, hay được xử lí như một đầu vào. Việc định dạng một file có thể khác nhau ở mỗi đích. Các khuôn dạng dữ liệu sau được ít sử dụng hơn kiển dữ liệu ngầm định:

- Khuôn dạng không in: là kiểu dữ liệu ngầm định ASCII và EBCDIC. Khuôn dạng file này không có thông tin định dạng. Chú ý rằng, định dạng sử các dạng chuẩn cho kí tự cách và phân lề.
- Định dạng Telnet được sử dụng cho các file một thiết bị đầu cuối dùng để hiển thị. Định dạng này gồm các kí tự điều khiển, kí tự xuống dòng, tab.
- Kiểm soát di chuyển bao gồm các kí tự điều khiển định dạng in. Theo khuôn dạng này, kí tự đầu tiên của mỗi dòng không được in ra. Thay vào đó, kí tự này xác định sự di chuyển theo trục đứng so với mép giấy trước khi một bản ghi hay một dòng nào đó được in ra.

10.7.2.5 Các cấu trúc dữ liệu

Giao thức FTP cho phép truyền các file có cấu trúc với 3 cấu trúc file khác nhau. Các cấu trúc tập tin này chủ yếu dùng để truyền các tập tin giữa các hệ thống có cấu trúc lưu trữ khác nhau. Có các dạng như sau :

- Cấu trúc theo kiểu file, xem file một chuỗi các byte dữ liệu nối tiếp nhau mà không được cấu trúc bên trong.
- Cấu trúc bản ghi được sử dụng để truyền các file là một chuỗi các bản ghi. Cấu trúc này được sử dụng cho các Host IBM nhưng hiện nay ít sử dụng.
- Cấu trúc trang được sử dụng cho các file được chia thành các đối tượng với kích thước khác nhau, có thể có các thông tin khác được thêm vào trong đó. Cấu trúc trang có một cấu trúc header để định nghĩa kích thước của trang, theo sau là nội dung của trang. Header của mỗi trang còn chứa số hiệu trang logic của các trang dữ liệu nhưng số hiệu trang đó không cần thiết khi truyền.

10.7.3 UserNEWS

Biểu tượng	Ý nghĩa	Biểu tượng	Ý nghĩa
:-)	Tôi hạnh phúc	=):=)	ABC Lincol
:-(	Tôi buồn/ tức giận	=):=)	Bác Sorn
:-|	Tôi thờ ơ	*<:-)	ông già Noel
;-)	Tôi nháy mắt	<:-(	Người tối dạ
;-(0)	Tôi kêu la	(-:	Người Uực
:-(*)	Người nôn (mửa)	:-)x	Man with bowtic
:+)	Cằm chẻ	# -)	Tóc mướt
:-))	Cằm chẻ	8 -)	Mang kính
:-{)	Ria	C:-)	Mão lớn

Khi mà có nhiều người thuê bao USENET, nhu cầu về những newsgroup mới, chuyên biệt hơn luôn được đòi hỏi. Kết quả là một thủ tục để tạo ra newsgroup mới, chuyên biệt hơn luôn được đòi hỏi. Kết quả là một thủ tục để tạo ra những

Newsgroup mới được. Trên Newsgroup, người ta có thể thảo luận, bầu cử, trao đổi với nhau.

10.7.4 WORLD-WIDE-WEB

World Wide Web (WWW) là một hệ thống quản lý thông tin phi cấu trúc. Bao gồm các Server cung cấp thông tin theo định dạng siêu văn bản (Hypertext) và các client (Browser, trình duyệt) nhận thông tin từ người sử dụng và đồng thời hiển thị thông tin mà các Server cung cấp theo định dạng được chỉ định bởi người sử dụng.

Thông tin trên WWW được biểu diễn trong các trang Web. Mỗi trang Web có thể là một chỉ mục hoặc một tài liệu chứa văn bản, hình ảnh, âm thanh, các liên kết... Người sử dụng có thể truy cập thông tin cần thiết trên WWW thông qua các đối tượng đã được đánh dấu trong tài liệu.

Các lệnh được dùng với WWW đã được định nghĩa trong giao thức HTTP (HyperText Transfer Protocol). Đây là giao thức chuẩn để liên lạc giữa Client và Server. Yêu cầu được gửi tới Server thông qua Client. Server xử lý các yêu cầu và gửi kết quả về cho Client yêu cầu. Kết quả sẽ được trình bày dưới dạng thích hợp cho người sử dụng.

- Phía máy chủ

Mỗi web Site có một máy chủ đảm nhận việc “lắng nghe” TCP tại cổng 80 cho những kết nối đến từ các máy khách (thường là các trình duyệt). Sau khi một kết nối được thiết lập, máy khách gửi yêu cầu và máy chủ trả lời đáp lại, kết nối chấm dứt. Giao thức HTTP định nghĩa cho các yêu cầu và trả lời hợp lệ.

Ví dụ người dùng kích lên một mẫu văn bản hoặc có thể là biểu tượng trỏ đến trang có tên (tức là URL hay địa chỉ tới máy trạm Internet). Một URL có 3 phần sau: tên của giao thức (http), tên của máy nơi có chứa trang web, và tên của tập tin chứa trang đó (hypetext/WWW/TheProject.html). Từ khi người dùng nhấp chuột cho đến khi trang web được hiện ra trên màn hình đã xảy ra các sự kiện sau :

1. Trình duyệt kiểm tra URL (xem xét đối tượng được chọn là gì).
2. Trình duyệt hỏi DNS về địa chỉ IP của URL.
3. DNS trả lời là 18.23.0.23
4. Trình duyệt tạo một kết nối TCP đến cổng 80 trên địa chỉ 18.23.0.23
5. Trình duyệt gửi lệnh GET /hypetext/WWW/TheProject.html.
6. Máy chủ gửi đến tập tin TheProject.html
7. Giải phóng kết nối TCP.

8. Trình duyệt hiển thị tất cả các văn bản trong tập tin TheProject.html.
9. Trình duyệt tiếp tục lấy về và hiển thị tất cả các hình ảnh có trong TheProject.html.

10.7.4.1 Ngôn ngữ HTML

HTML (HyperText Markup Laguage) là một ngôn ngữ HTML là một ngôn ngữ có cấu trúc, nó bao gồm các thẻ (TAGS) và các thực thể (ENTITY), dùng để cung cấp các chỉ thị định dạng để phục vụ cho việc trình bày văn bản trên Web.

Một tập tin HTML là một tập tin văn bản trong đó một số xâu ký tự được coi là các thẻ đánh dấu các vùng tài liệu và ấn định các ý nghĩa đặc biệt cho chúng. Các thẻ là các xâu ký tự được bắt đầu là dấu nhỏ hơn (<) và kết thúc bằng dấu lớn hơn (>). Các thẻ có thể được phân làm nhiều loại tùy theo nội dung, chức năng, kiểu tác động của chúng như: Thẻ mô tả định dạng, thẻ mô tả cấu trúc, thẻ rỗng, thẻ chứa...

Cấu trúc tổng quát của một tài liệu HTML như sau :

<HTML> *Thông báo cho trình duyệt đây là một văn bản tài liệu HTML*
<HEAD> *Thông báo bắt đầu phần đầu của tài liệu*
<TITLE> *Tiêu đề của tài liệu* </TITLE>
Phần đầu của tài liệu đặt tại đây
</HEAD> *Kết thúc phần đầu*
<BODY> *Thông báo bắt đầu phần thân tài liệu*
.....
Nội dung tài liệu HTML được đặt tại đây
</BODY> *Kết thúc phần thân tài liệu*
</HTML> *Kết thúc tài liệu HTML*

Phần đầu đề của tài liệu HTML thường chứa tiêu đề của tài liệu, tên tác giả, lời chú thích, tóm tắt... Đây là phần giúp ích cho việc tìm kiếm thông tin trên WEB hoặc cho các dịch vụ tìm kiếm có thể đánh chỉ mục, tiến hành tìm kiếm một cách dễ dàng. Một số các thẻ phục vụ trong phần đầu như: Title, Meta, Isindex...

Phần thân là phần chính của tài liệu HTML, nằm giữa cặp thẻ <BODY> và </BODY>, nó định nghĩa, hiển thị toàn bộ nội dung bên trong của tài liệu. Trong phần thân ta có thể sử dụng các thẻ để định dạng văn bản, chèn các hình ảnh, bảng biểu, liên kết...

Người sử dụng có thể tạo một tài liệu HTML bằng cách sử dụng các trình soạn thảo Web chuyên dụng như Microsoft Front Page 2000, hoặc Microsoft Word, Notepad ...

Một số thẻ HTML quan trọng :

1. *Thẻ <!-- (chú thích) -->:*

Dùng để thêm những dòng chú thích trong file HTML, người ta dùng thẻ nầy. Nội dung văn bản nằm giữa <!-- và --> sẽ được chương trình Browse bỏ qua. Cho phép có khoảnh trắng giữa -- và >, nhưng không được có khoảng trắng giữa <! và -- .

Thí dụ:

<HEAD> <TITLE>The HTML Reference</TITLE>

<!-- Created by Nguyen Tan Khoi, April 1996 --> </HEAD>

2. *Thẻ <A>*

Dùng để tạo các siêu liên kết (HyperLink). WWW cho phép kết nối và giao tiếp giữa các tài nguyên một cách dễ dàng nhờ định nghĩa các loại liên kết sau:

1. Liên kết giữa các thành phần khác nhau trong một tài liệu HTML.
2. Liên kết giữa các tài liệu HTML khác nhau.
3. Liên kết với các dạng tài liệu Multimedia.
4. Truy cập tới các dịch vụ thông tin khác trên mạng Intranet/Internet

Các thuộc tính của thẻ <A> như sau:

a. Liên kết đến điểm neo trong trang HTML

- NAME: Thuộc tính NAME xác định một vị trí để những thành phần khác trong tài liệu hoặc trong tài liệu khác có thể tham trỏ đến (gọi là điểm neo trong tài liệu HTML). Thí dụ :

  <A NAME="coffee"> Coffee</A>

  Các tài liệu khác có thể liên kết với tài liệu này ngay tại vị trí xác định.

b. Liên kết đến một trang HTML

<A HREF = "URL_HTML[#Name_Anchor]"> *Nội dung thông báo* </A>

Trong đó URL_HTML là địa chỉ để tham chiếu tới tài liệu HTML.

Nếu chỉ ra Name_Anchor thì có nghĩa ta định nghĩa một điểm neo dùng để chuyển đến một vị trí được quy định sẵn trong tài liệu HTML này. Thí dụ:

The <A HREF="document.html#glossary"> GLOSSARY </A>

Trong thí dụ trên, nếu kích vào "GLOSSARY" sẽ được chuyển đến tài liệu document.html, ngay tại vị trí điểm neo có tên glossary trong tài liệu này.

c. *Liên kết với các kiểu dữ liệu khác nhau*

Để liên kết giữa tài liệu hiện thời với các kiểu dữ liệu khác nhau như: hình ảnh, âm thanh, video...

<A HREF ="URL_DATA"> ...</A>

Trong đó URL_DATA là địa chỉ tới kiểu dữ liệu cần liên kết. Ví dụ:

<A HREF ="car.jpg">

<IMG SRC ="carla.gif" WIDTH=87 HEIGHT=60> </A>

d. *Liên kết với các dịch vụ thông tin khác trên mạng*

<A HREF = "URL_Service" > ... </A>

Trong đó URL_Service là một địa chỉ đến các dịch vụ trên internet.

<A HREF="http://..."> Liên kết với 1 Web Site.

<A HREF="ftp://..."> Với 1 Ftp Site.

<A HREF="gopher://..."> Với 1 Gopher server.

<A HREF="news:..."> Liên kết với 1 nhóm Tin.

<A HREF="mailto:..."> liên kết tới 1 địa chỉ gởi Mail. Liên kết nầy sẽ kích hoạt chương trình Mail và tự động điền địa chỉ vào mục To dùm bạn. Bạn có thể khai báo luôn cả chủ đề thư (?subject).

Thí dụ: <A HREF="mailto:cmlehunt@swan.ac.uk?

subject=The HTMLib is fantastic">link text</A>

- TARGET: Chương trình Browser có thể nạp đối tượng liên kết vào 1 cửa sổ chỉ định bằng thẻ nầy. Nếu cửa sổ nầy cha có, trình Browse sẽ mở 1 cửa sổ mới. Chủ yếu thẻ nầy dùng cho frames.

  Dạng chung:

  <A HREF="url.html" TARGET="window_name">Link text</A>

  Trong đó window_name là tên đặt cho Frame.

  Khi kích chuột vào dòng "Link text", trang "url.html" sẽ được nạp vào frame có tên chỉ định.

Ngoài ra ta còn có thể chèn thêm các Script sau vào thẻ <A> dựa vào các phương thức như sau :

Phương thức	Giải thích
OnMouseOver	Khi bạn di chuyển Mouse đến liên kết, sẽ có 1 dòng văn bản mô tả xuất hiện trong thanh trạng thái của trình Browse. Thí dụ: <A HREF="index.html" OnMouseOver="self.status=('Back to the main page')">Link text</A> Dòng chữ "Back to the main page" sẽ hiện trong thanh trạng thái khi dời Mouse đến chữ "Link text".
OnMouseOut	Tương tự như trên nhưng dòng chữ nầy lại xuất hiện khi kéo Mouse ra khỏi liên kết. Thí dụ: <A HREF="index.html" OnMouseOut="alert('Oh please go to this document')">Link text</A>
OnClick	Khi bấm Mouse lên liên kết, sẽ xuất hiện hộp thoại yêu cầu xác nhận. Thí dụ: <A HREF="http://www.netscape.com/" OnClick="confirm('Are you want to go to the Netscape site?')">Link text</A>

3. *Thẻ <INPUT>*

Dùng để tạo một field để nhận tác động của người sử dụng.

<INPUT TYPE = "*Kiểu*" NAMQ = "*TênĐT*"
SIZQ = "*KíchThước*"
VALUQ = "*Giatri*" MAXLQNGTH = "n" . . . >

Các thuộc tính:

Thuộc tính	Giải thích
ALIGN	So hàng cho field.
CHECKED	Kiểm tra người dùng đã đánh dấu cho checkbox hay radio button chưa.
MAXLENGTH	Chỉ định độ dài ký tự có thể nhập vào text field, độ dài nầy có thể lớn hơn kích thước Text field. Mặc định là không giới hạn.

NAME	Tên của Field.
SIZE	Khai báo kích thước hay số lượng ký tự cho field.

- TYPE: Chỉ định kiểu của Field:

Giá trị	Giải thích
BUTTON	Chèn một nút bấm vào tài liệu. Giá trị VALUE dùng chỉ định Text sẽ hiện trong nút nầy. Thí dụ: <input type="button" value="hello" name="btnhello">
HIDDEN	Với thuộc tính nầy, field sẽ không hiển thị ra nhưng nội dung của field vẫn có giá trị. Dùng trao đổi thông tin ngầm giữa Client/Server.
PASSWORD	Giống như Text, nhưng ký tự nhập vào sẽ không hiển thị ra.
CHECKBOX	Chèn 1 chechbox vào tài liệu. Thí dụ : So thich <input type="checkbox" name="C1" value="ĂN">The thao <input type="checkbox" name="C2" value="ĂN">Xem phim
RADIO	Chèn 1 field có dạng nút Radio. Ví dụ : Gioi tinh <input type="radio" checked value="V1" name="R1">Nam <input type="radio" name="R2" value="V2">Nu
RESET	Chèn 1 nút bấm dùng phục hồi lại tình trạng cũ cho các field. Đặt tên của nút nầy qua thuộc tính Values.
SUBMIT	Một dạng nút bấm giống RỌSỌT. Có tác dụng giống nh xác nhận đồng ý. Thí dụ: <input type="submit" value="Submit" name="B1"> <input type="reset" value="Reset" name="B2"> Chèn 1 nút có tên "SUBMIT" và sẽ hiển thị thông báo "Xin chao các bạn" khi người sử dụng Mouse vào nút nầy : <INPUT TYPE="SUBMIT" OnClick="Xin chao các bạn'">
TEXT	Nhập 1 dòng text vào fields. Dùng thuộc tính SIZỌ và MAXLENGTH để quy định kích thước. Trong trường hợp cần nhập

	nhiều dòng, phải dùng thẻ <TEXTAREA>.
VALUE	Chỉ định Text sẽ hiển thị trên các nút bấm.
IMAGE	Chèn field chứa hình ảnh để người dùng bấm Mouse khi chọn. <INPUT TYPE="IMAGE" SRC="../ iexplore.gif" ALIGN="middle">

4. *Thẻ TEXTAREA*

Cho phép nhập nhiều dòng văn bản vào một hộp Text.

Thí dụ:

```
<TEXTAREA
NAME="descr"
COLS="30"          ROWS="3"
OnBlur="count_char(document.egForm.descr.value)">Enter a short description here
 </TEXTAREA>
```

Ví dụ : <textarea name="Ghichu" rows="2" cols="20"></textarea>

5. *Thẻ FORM*

Forms là một thiết lập nhỏ trong HTML, nó cho phép người sử dụng đưa vào các thông tin. Giao diện Forms tạo nên sự thuận lợi trong việc tương tác giữa người sử dụng và các dịch vụ. Trên Form ta có thể tạo các thành phần như các nút lệnh, các trường văn bản (Text) hay các danh sách lựa chọn ... Khi forms được hoàn thành bởi người sử dụng, Client sẽ gửi thông tin đến Server, Server sẽ thực thi các chương trình kết hợp với form và các tham số là các thông tin nhận từ Form.

Thông thường các Form sử dụng cho hai mục đích chính:

- Dùng để thu thập thông tin từ người sử dụng.
- Là trung gian để tương tác qua lại giữa người sử dụng và hệ thống.

Cú pháp : <Form ACTION = “Action” METHOD="PhuongThuc">

Action: là một URL hoặc một Script mà khi nút ***Submit*** được nhấn nó sẽ thực thi.

Method=GET/POST : Xác định kiểu yêu cầu mà trình duyệt gởi đến cho Server.

- METHOD = GET: trình duyệt sẽ bổ sung dữ liệu đầu vào dưới dạng một biến môi trường là CGI_QueryString.
- METHOD=POST: Forrn dữ liệu đầu vào sẽ đợi từ các thiết bị nhập của Server cùng với một số dữ liệu được lưu trữ trong biến môi trường CGI_ContentLength.

EncType: cung cấp kiểu Mime của tập được dùng như đầu vào trong các biểu mẫu.

Ví dụ : <Form ACTION = METHOD="GET">

6. *Thẻ TABLE*

- Dùng để tạo ra một bảng. Bảng được tạo thành từ các hàng, trên mỗi hàng có các ô (cell).

```
<TABLE>
        <TABLE   BORDER = “n” . . . >
        <TR>
                <TD>  . . . </TD> <TD>  . . . </TD> <TD>  . . . </TD>
        </TR>
        . . . .
        <TR>
                <TD>  . . . </TD> <TD>  . . . </TD> <TD>  . . . </TD>
        </TR>
</TABLE>
```

7. *Thẻ SELECT*

- Hiển thị hộp ComboBox cho phép chọn lựa một trong nhiều giá trị :

```
<SELECT  NAME =”TenĐT”>
        <OPTION SELECTED VALUE =”Gia trị 1”> Nội dung 1
        <OPTION SELECT VALUE =”Gia trị 2”> Nội dung 2
                . . .
</SELECT>
```

Ví dụ :

```
<p>Que quan <select size="1" name="cboQuequan">
        <option selected>Da Nang</option>
        <option>Hue</option>
        <option>Ha Noi</option>
        </select></p>
```

8. *Thẻ <APPLET>*

Dùng để chèn Applet Java vào trang Web. Có dạng tổng quát sau:

```
<APPLET
    [CODEBASE = URL] [CODE = appletFile]
    [NAME = appletInstanceName]:
    [ARCHIVE = compressed file] [ALT = alternateText]
    [WIDTH = pixels]  [HEIGHT = pixels]  [ALIGN = alignment]
    [VSPACE = pixels] [HSPACE = pixels]
    [ARCHIVE = URL to archive]
 </APPLET>
```

Trong đó :

Tham số	Giải thích
CODEBASE=URL	Chỉ định địa chỉ tuyệt đối của Applet.
CODE=appletFile	Chỉ định địa chỉ tương đối của Applet.
ALT=alternateText	Chỉ định dòng text sẽ hiển thị trong trường hợp trình Browse không hiểu Applet.
NAME = appletInstanceName	Đặt tên cho Applet để phục vụ cho việc tìm kiếm.
WIDTH=pixels HEIGHT=pixels	Chỉ định kích thước cho Applet.
ALIGN=alignment	Dùng canh lề, có các giá trị sau: LEFT, RIGHT, TOP, TEXTTOP, MIDDLE, ABSMIDDLE, BASELINE, BOTTOM, ABSBOTTOM.
VSPACE=pixels HSPACE=pixels	Chỉ định khoảng trống bao chung quanh Applet.
ARCHIVE=compressed file	Khai báo các file nén cần thiết của Applet để trình Browse tải về máy cá nhân, phục vụ cho việc đọc lại sau nầy.

Ví dụ:

```
<APPLET CODEBASE=http://200.201.202.180/applets/ NervousText
CODE="NervousText.class"
WIDTH=400      HEIGHT=75
ALIGN=CENTER>
<PARAM NAME="text" VALUE="This is the Applet Viewer.">
</APPLET>
```

Chỉ thị cho trình Browse nạp Applet ở địa chỉ htttp://java.sun.com/JDK-prebeta1/applets/NervousText/NervousText.class". Chỉ định kích thước là 400x75 pixels và canh giữa dòng. Nếu trình Browse hiểu Applet, dòng "This is the Applet Viewer." sẽ hiển thị và Applet tạo hiệu ứng cho dòng chữ nầy. Nếu trình Browse không hiểu Applet, nó sẽ bỏ qua nội dung của <APPLET> cũng như <PARAM> và chỉ hiển thị nội dung của <BLOCKQUOTE>

9. ***Thẻ <IMG>***

Dùng để chèn 1 file hình vào tài liệu HTML

Các thuộc tính :

- ALIGN="left/right/top/texttop/middle/absmiddle/baseline/bottom/absbotto": So hàng hình ảnh với Text.
- ALT="Alternative Text": Cho hiển thị 1 dòng text thay thế cho file hình trong trường hợp trình Browse đang ở trong chế độ không hiển thị hình ảnh. Dòng Text nầy cũng hiển thị theo dạng ToolTip khi dời chuột đến hình.

Ví dụ: <IMG SRC="triangle.gif" ALT="Warning:"> Read these instructions.

- SRC="URL of image": Chỉ định đia chỉ file hình chèn vào trang Web.

Ví dụ : <IMG SRC="warning.gif">Be sure to read these instructions.

- WIDTH=value/ HEIGHT=value: Chỉ định khoảng cách dành sẳn cho hình trong khi trình Browse nạp toàn bộ hình.
- BORDER=value: Chỉ định cho hiển thị đường viền bao quanh hình ảnh. Ta có thể chọn "0" để hiển thị đường viền màu xanh khi có liên kết.
- VSPACE=value HSPACE=value: Quy định khoảng trống giữa hình và Text. VSPACE cho trên và dới hình, HSPACE cho trái và phải hình. Value tính theo pixel.
- LOWSRC: Thuộc tính nầy cho phép hiển thị 2 hình lần lượt trong cùng 1 vị trí. Thường dùng để nạp một hình nhỏ trong khi chờ đợi nạp hình chính có dung luợng file lớn hơn:

Ví dụ: <IMG SRC="hiquality.gif" LOWSRC="lowquality.gif">

Đầu tiên trình Browse sẽ hiển thị file hình "lowquality.gif". Sau khi nạp hoàn tất cả trang, trình duyệt sẽ nạp file hình chính thức vào thay thế.

10.7.4.2 Chỉ định tài nguyên trong URL

Để chỉ định vị trí của tài nguyên HTTP dùng URL (Uniform Resouce Locators) đó là tên quy ước để nhận diện một cách duy nhất vị trí của một thư mục

hoặc một tập tin trên Intranet/Internet. Trong URL cũng chỉ định giao thức kết nối như HTTP, GOPHER... cần thiết cho việc tìm kiếm và lấy tài nguyên. Nếu ta biết URL của một tài nguyên ta có thể truy xuất nó một cách trực tiếp hoặc thông qua các siêu liên kết trong các tài liệu.

URL sử dụng một dòng đơn các ký tự ASCII. Sơ đồ này bao gồm các giao thức trên Intranet/Internet như FTP, Gopher, http... URL là một trong những công cụ cơ sở của WWW và được dùng trong các tài liệu HTML để tham chiếu đến các tài nguyên trên mạng.

Một URL gồm các thông tin sau :

a. Tên các giao thức khi truy cập Server (như HTTP, Gopher, Wais...).

b. Tên miền của Server thực thi, theo bất cứ thông tin về user và password của site trên Intranet/Internet.

c. Số cổng mà server sử dụng. Nếu điều này không được chỉ rõ trình duyệt sẽ dùng số cổng mặc định trong giao thức (cổng 80).

d. Định vị của tài nguyên trong kiến trúc phân cấp của Server.

10.7.4.3 Giao thức HTTP

Giao thức HTTP (Hyper Text Transfer Protocol - Giao thức truyền siêu văn bản) sử dụng cho các dịch vụ truyền thông đa phương tiện WWW, dựa trên mô hình Client/Server. Dịch vụ WWW cho phép NSD kết hợp văn bản, âm thanh, hình ảnh, hoạt hình tạo nên nguồn thông tin tư liệu. Đặc biệt ở đây là thông tin tư liệu trong WWW có dạng HyperText - là dạng tư liệu chuẩn trong WWW. Giao thức cho phép lấy và đọc nhanh các tư liệu đó. HTTP là giao thức truyền thông nhưng có thêm ưu điểm là thông tin tư liệu cần truy cập lại có chứa các liên kết tới cá tư liệu khác nằm khắp nơi trên mạng Internet.

Phần mềm cho WWW Server là một chương trình điều khiển sự thu nhập các tư liệu WWW trên một máy chủ. Để truy cập WWW, cần thiết phải chạy hệ thống ứng dụng WWW là một trình duyệt (browser) trên máy của WWW Client.

HTTP là một giao thức Internet Client/Server, được thiết kế để truyền các dạng dữ liệu siêu văn bản. HTTP là một giao thức không trạng thái, nghĩa là khi Server đáp ứng dữ liệu được yêu cầu bởi Client xong thì server huỷ bỏ kết nối đó không tốn bộ nhớ cho sự kiện. Không trạng thái là yếu tố làm cho tốc độ truyền dẫn giữa HTTP Server và HTTP Client rất nhanh.

Các giao tiếp HTTP truyền dữ liệu dưới dạng các ký tự 8 bit hay một octet. Điều này đảm bảo truyền dẫn an toàn mọi dạng dữ liệu bao gồm hình ảnh, âm thanh, các tài liệu HTML hay các chương trình khả thi.

1. *Các giai đoạn kết nối của HTTP*

Một HTTP Server kết nối thông qua 4 giai đoạn:

- ***Mở kết nối:*** Client tiếp xúc với Server tại địa chỉ internet và số cổng chỉ định trong URL (cổng mặc định là 80)
- ***Tạo yêu cầu :*** Client gửi một thông điệp tới Server yêu cầu dịch vụ. Yêu cầu bao gồm các tiêu đề HTTP, nó định nghĩa phương thức được yêu cầu cho tác vụ và cung cấp thông tin về khả năng của Client (được theo sau dữ liệu gửi tới Server). Các phương thức HTTP điển hình là GET để nhận các đối tượng từ Server hoặc POST để chuyển dữ liệu cho đối tượng (ví dụ như các chương trình GateWay) trên Server.
- ***Gửi đáp ứng :*** Server trả lời cho Client bao gồm các tiêu đề để trả lời trạng mô tả trạng thái của tác vụ (ví dụ thành công, không thành công...) theo sau dữ liệu thật sự.
- ***Đóng kết nối:*** Kết nối được đóng, Server không giữ lại dấu vết của tác vụ đã hoàn thành. Thủ tục này có nghĩa là mỗi kết nối chỉ xử lý một tác vụ và do đó chỉ có thể tải xuống Client chỉ một tệp dữ liệu. Tính chất không trạng thái của tác vụ cũng có nghĩa là mỗi kết nối không hề biết về các kết nối trước đó.

2. *Các phương thức của giao thức HTTP*

Phương thức	Giải thích
GET	Lấy dữ liệu hiển thị trong URL. Dữ liệu cũng có thể gửi trong URL thông qua một chuỗi truy vấn. Đây cũng là nơi dữ liệu gửi từ ISINDEX hoặc Form với thuộc tính METHOD="GET"
HEAD	Lấy thông tin của HTTP, Header chỉ định trong URL.
POST	Gửi dữ liệu đến cho URL nếu URL là tồn tại. Phương thức này được dùng bởi những thành phần của Form trong HTML với giá trị thuộc tính METHOD="POST".
PUT	Là nơi mà dữ liệu gửi bởi Client biểu thị trong URL, nó sẽ thay thế nội dung của URL đã có.
DELETE	Xóa tài nguyên cục bộ tại nơi được chỉ định bởi URL.
LINK	Liên kết một đối tượng đã tồn tại với một đối tượng khác.
UNLINK	Huỷ bỏ một liên kết đã được tạo bởi phương thức LINK.

BÀI TẬP

1. Những nguyên tắc cơ bản giám sát và quản trị hệ thống mạng máy tính
2. Khảo sát cấu trúc và hoạt động dịch vụ DNS
3. Khảo sát cấu trúc và hoạt động của giao thức SNMP
4. Khảo sát cấu trúc và hoạt động của giao thức HTTP
5. Tìm hiểu giao thức DHCP.

- -

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Nguyễn Thúc Hải, ***Mạng máy tính và các hệ thống mở*** , NXB Giáo dục, 1997

[2] Lê Văn Sơn, ***Giáo trình mạng máy tính***, Trường ĐH Bách Khoa Đà Nẵng, 1998

[3] Nguyễn Hồng Sơn, ***Giáo trình hệ thống mạng máy tính CCNA,*** Nhà XB Lao động, 2002

Tiếng Anh

[4] Douglas E.Comer, ***Computer Networks and Internets***, Prentice Hall, 1997

[5] Ed Taylor, ***TCP/IP complete***, McGraw-Hill, 1998

[6] Microsoft Press, ***Networking Essentials***

[7] Stallings W., ***Data and Computer Communications***, Macmillan Publishing, 1995

[8] Tanenbaum Andrew S., ***Computer*** **Networks,** Prentice Hall, 1997

[9] Pujolle, ***Les rĐseaux***, EYROLLES, 2003

@2004, Nguyễn Tấn Khôi

Khoa Công Nghệ Thông Tin - Trường Đại học Bách Khoa Đà Nẵng

----------o & o----------

BÀI GIẢNG
MÔN: **MẠNG MÁY TÍNH**

Biên soạn: ThS. Trần Bá Nhiệm

GIỚI THIỆU MÔN HỌC

- Mục đích của môn học
  - Kiến thức cơ bản về mạng máy tính
  - Mô hình tham khảo OSI
  - Mô hình TCP/IP
- Thời lượng: 5 buổi học

GIỚI THIỆU MÔN HỌC

- Nội dung môn học
  - Chương 1: Tổng quan về mạng máy tính
  - Chương 2: Cấu trúc của mạng
  - Chương 3: Phương tiện truyền dẫn và thiết bị mạng
  - Chương 4: Data link
  - Chương 5: TCP/IP
  - Chương 6: Khái niệm cơ bản về bảo mật mạng
  - Bài tập

CHƯƠNG 1:
TỔNG QUAN VỀ MẠNG MÁY TÍNH

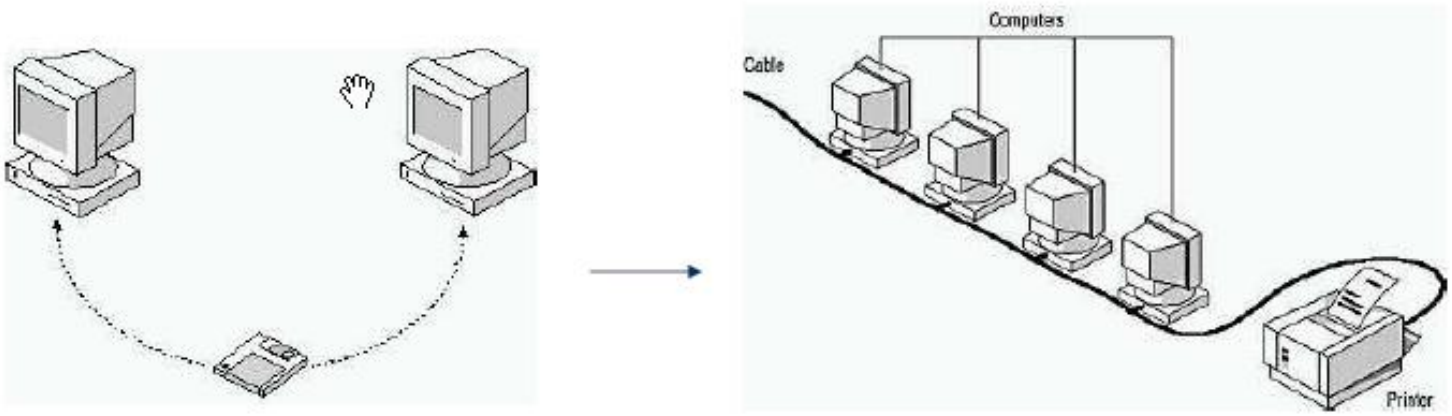
- Khái niệm về mạng máy tính
- Ứng dụng của mạng máy tính
- Phân loại mạng máy tính
- Mô hình OSI

Khái niệm về mạng máy tính

- Một tập hợp của các máy tính độc lập được kết nối bằng một cấu trúc nào đó.
- Hai máy tính được gọi là kết nối nếu chúng có thể trao đổi thông tin.
- Kết nối có thể là dây đồng, cáp quang, sóng ngắn, sóng hồng ngoại, truyền vệ tinh…

Ứng dụng của mạng máy tính

- Chia sẻ thông tin
- Chia sẻ phần cứng và phần mềm
- Quản lý tập trung

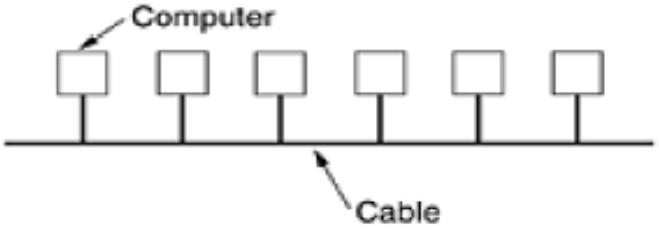


Phân loại mạng máy tính

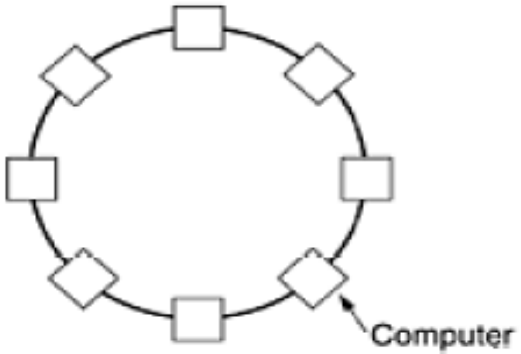
- Cách phân loại mạng máy tính được sử dụng phổ biến nhất là dựa theo khoảng cách địa lý của mạng: Lan, Man, Wan.
- Theo kỹ thuật chuyển mạch mà mạng áp dụng: mạng chuyển mạch kênh, mạng chuyển mạch thông báo, mạng chuyển mạch gói.
- Theo cấu trúc mạng: hình sao, hình tròn, tuyến tính…
- Theo hệ điều hành mà mạng sử dụng: Windows, Unix, Novell…

LANs (Local Area Networks)

- Có giới hạn về địa lý
- Tốc độ truyền dữ liệu cao
- Tỷ lệ lỗi khi truyền thấp
- Do một tổ chức quản lý
- Sử dụng kỹ thuật Ethernet hoặc Token Ring
- Các thiết bị thường dùng trong mạng là Repeater, Brigde, Hub, Switch, Router.

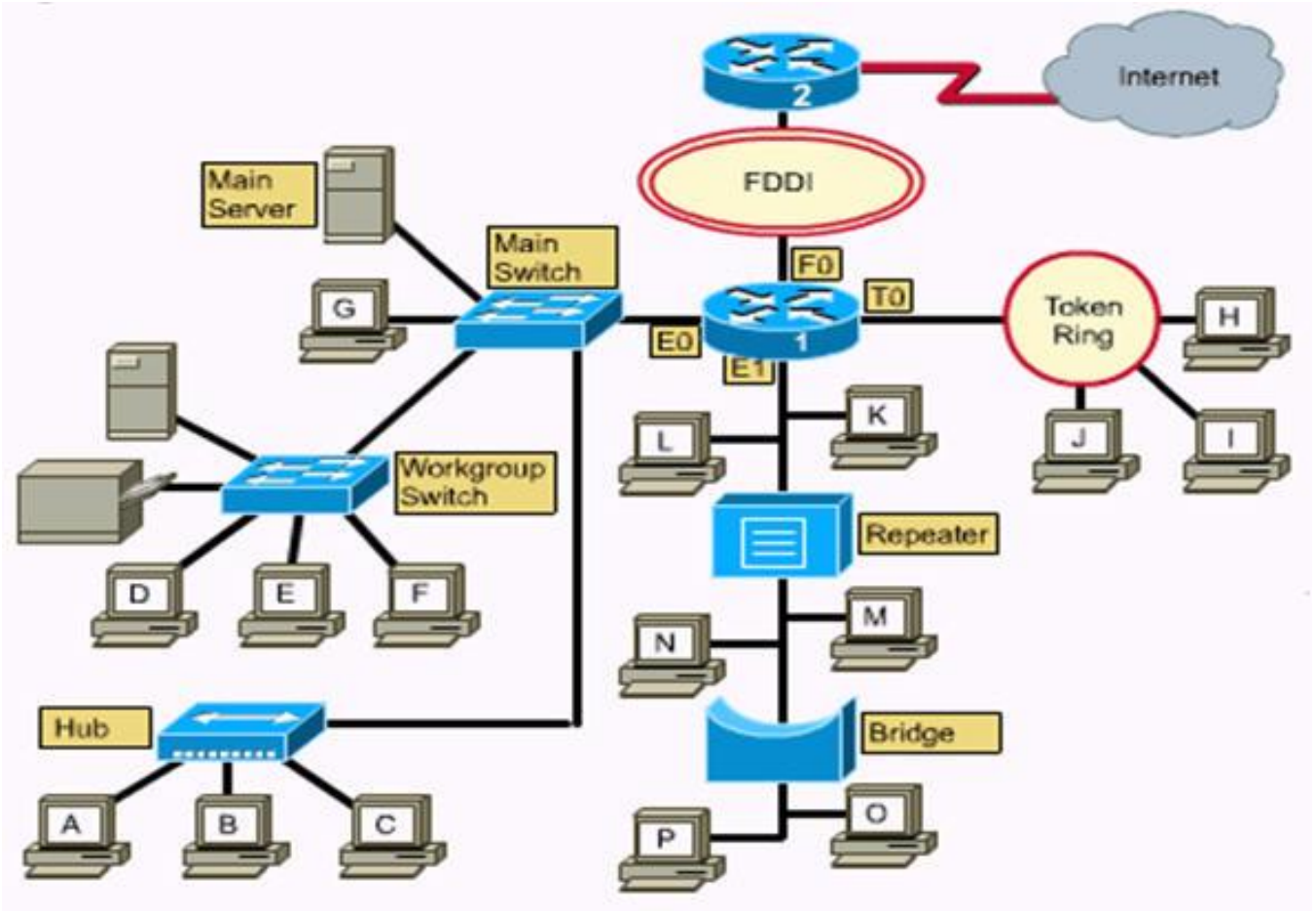


802.3 Ethernet



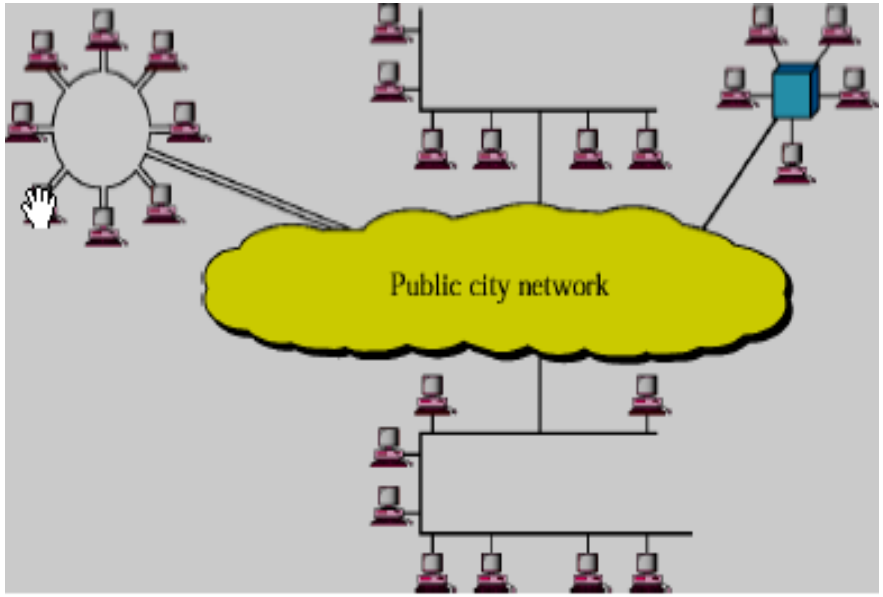
802.5 Token Ring

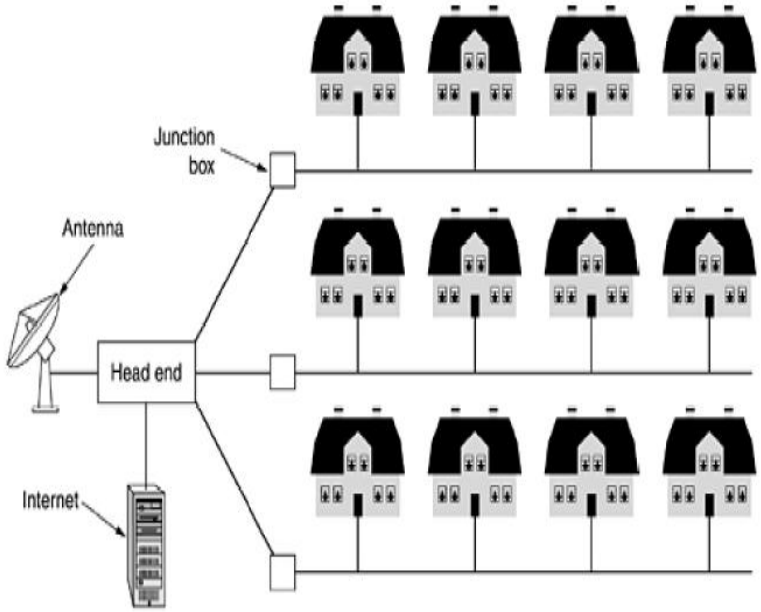
LANs


Internet
2
FDDI
F0
T0
Token Ring
H
J
I
Main Server
Main Switch
G
E0
1
E1
K
L
Workgroup Switch
Repeater
D
E
F
M
N
Hub
Bridge
A
B
C
O
P

MANs (Metropolitan Area Networks)

- Có kích thước vùng địa lý lớn hơn LAN
- Do một tổ chức quản lý
- Thường dùng cáp đồng trục hoặc cáp quang

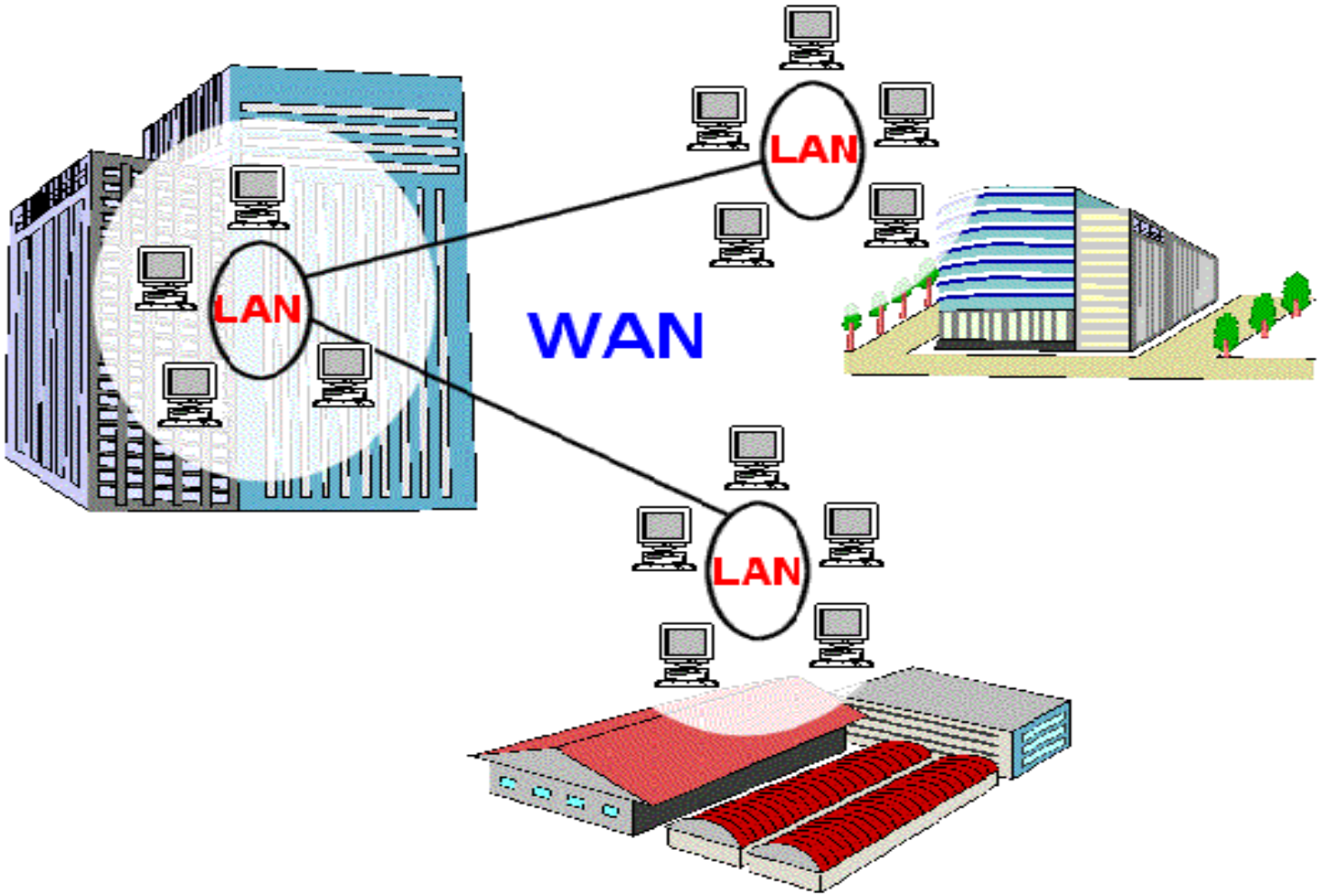




WANs (Wide Area Networks)

- Là sự kết nối nhiều LAN
- Không có giới hạn về địa lý
- Tốc độ truyền dữ liệu thấp
- Do nhiều tổ chức quản lý
- Sử dụng các kỹ thuật Modem, ISDN, DSL, Frame Relay, ATM

WANs (Wide Area Networks)



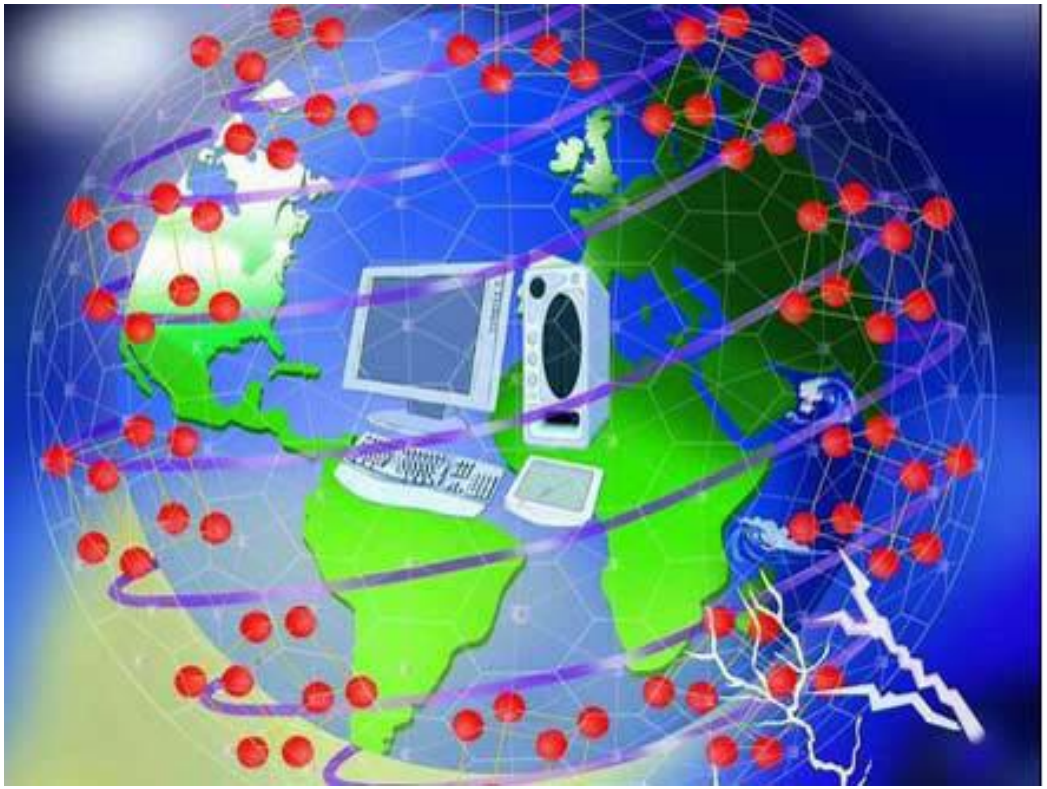
Mạng không dây (Wireless Networking)

- Do tổ chức IEEE xây dựng và được tổ chức Wi-fi Alliance đưa vào sử dụng trên toàn thế giới.
- Có các tiêu chuẩn: chuẩn 802.11a, chuẩn 802.11b, chuẩn 802.11g (sử dụng phổ biến ở thị trường Việt Nam), chuẩn 802.11n (mới có).
- Thiết bị cho mạng không dây gồm 2 loại: card mạng không dây và bộ tiếp sóng/điểm truy cập (Access Point - AP).

Mạng không dây

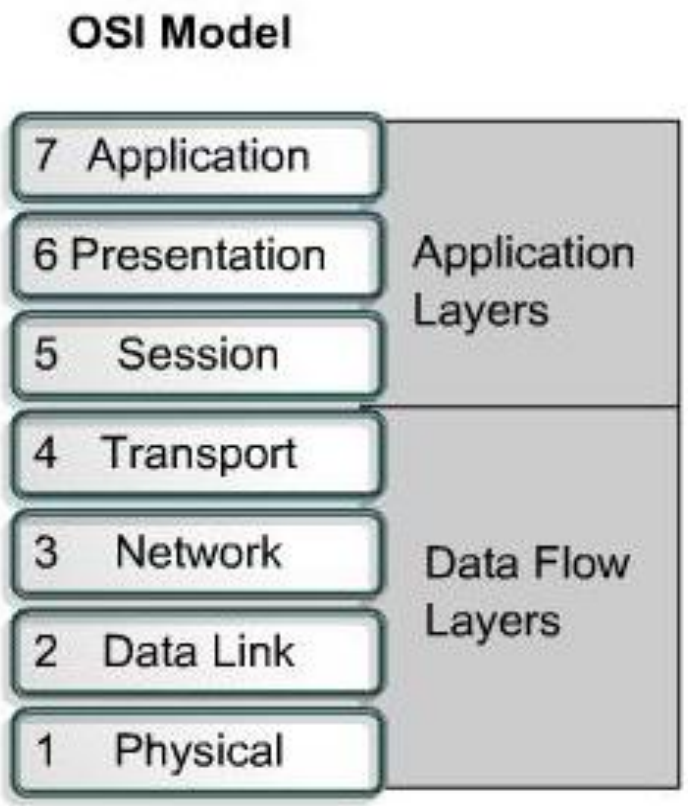
Internet

Một hệ thống mạng của các máy tính được kết nối với nhau qua hệ thống viễn thông trên phạm vi toàn thế giới để trao đổi thông tin.

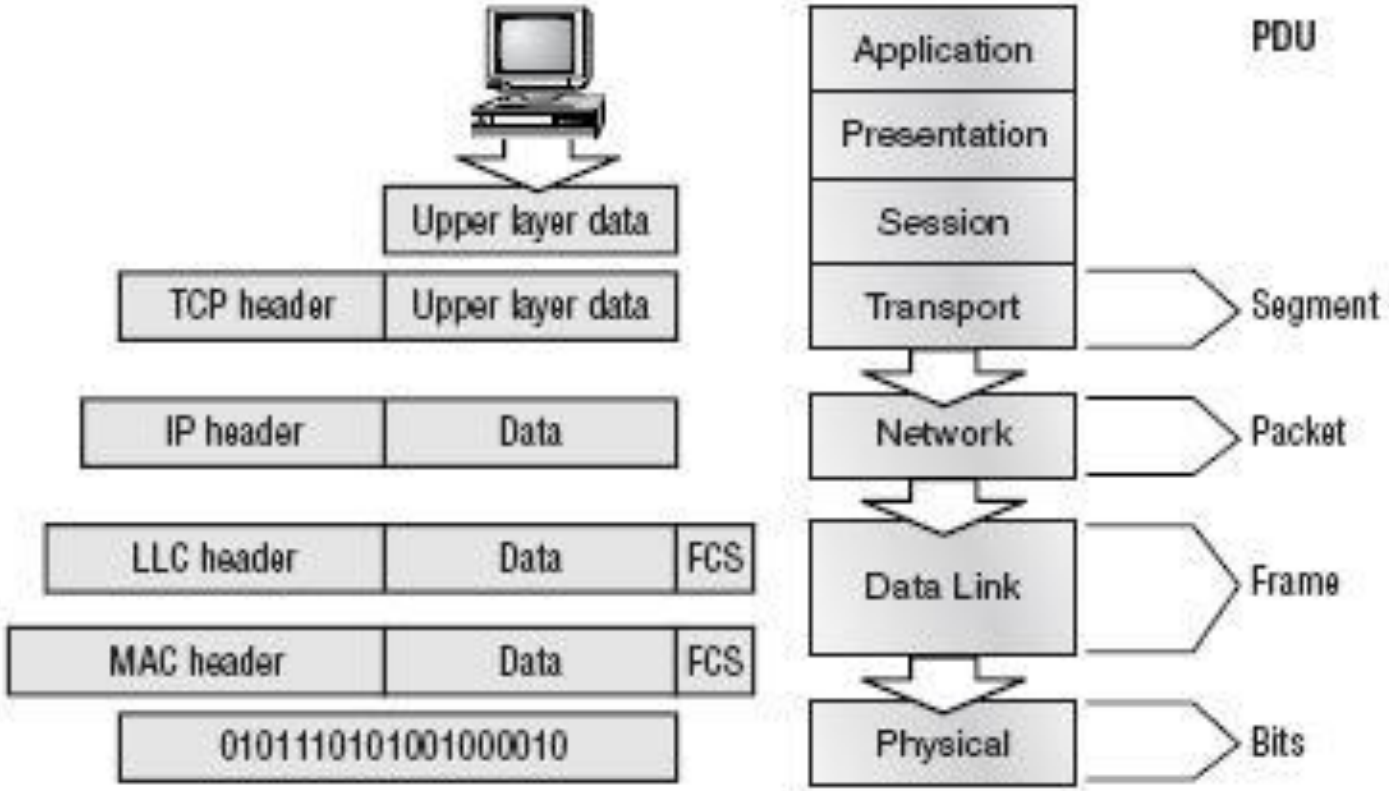


Mô hình OSI
(Open Systems Interconnection)

- Lý do hình thành: Sự gia tăng mạnh mẽ về số lượng và kích thước mạng dẫn đến hiện tượng bất tương thích giữa các mạng.
- Ưu điểm của mô hình OSI:
  - Giảm độ phức tạp
  - Chuẩn hóa các giao tiếp
  - Đảm bảo liên kết hoạt động
  - Đơn giản việc dạy và học

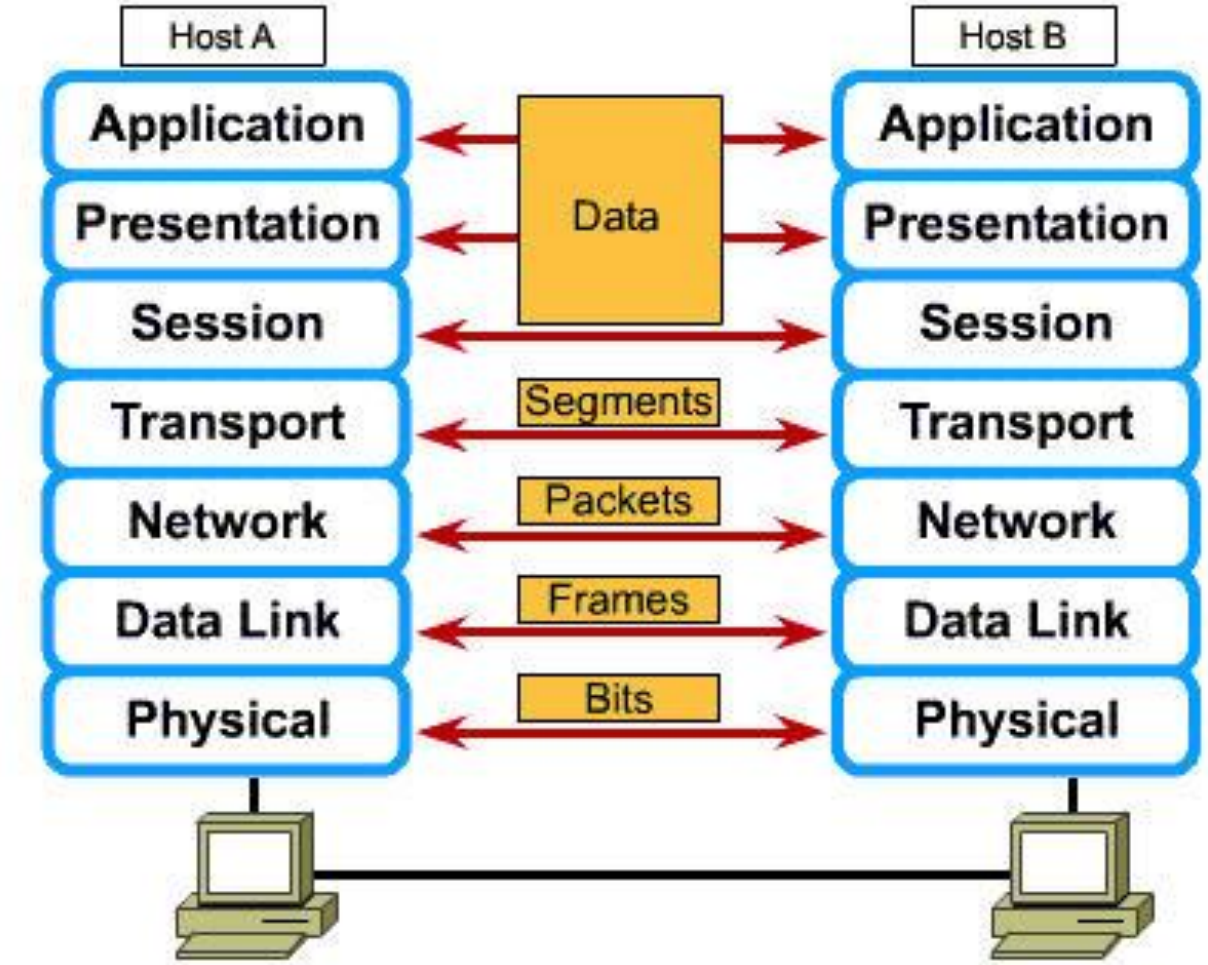


Mô hình OSI

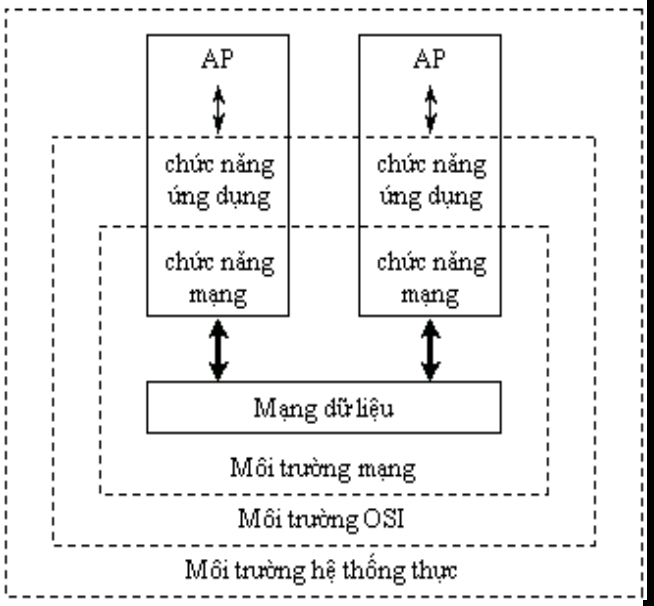


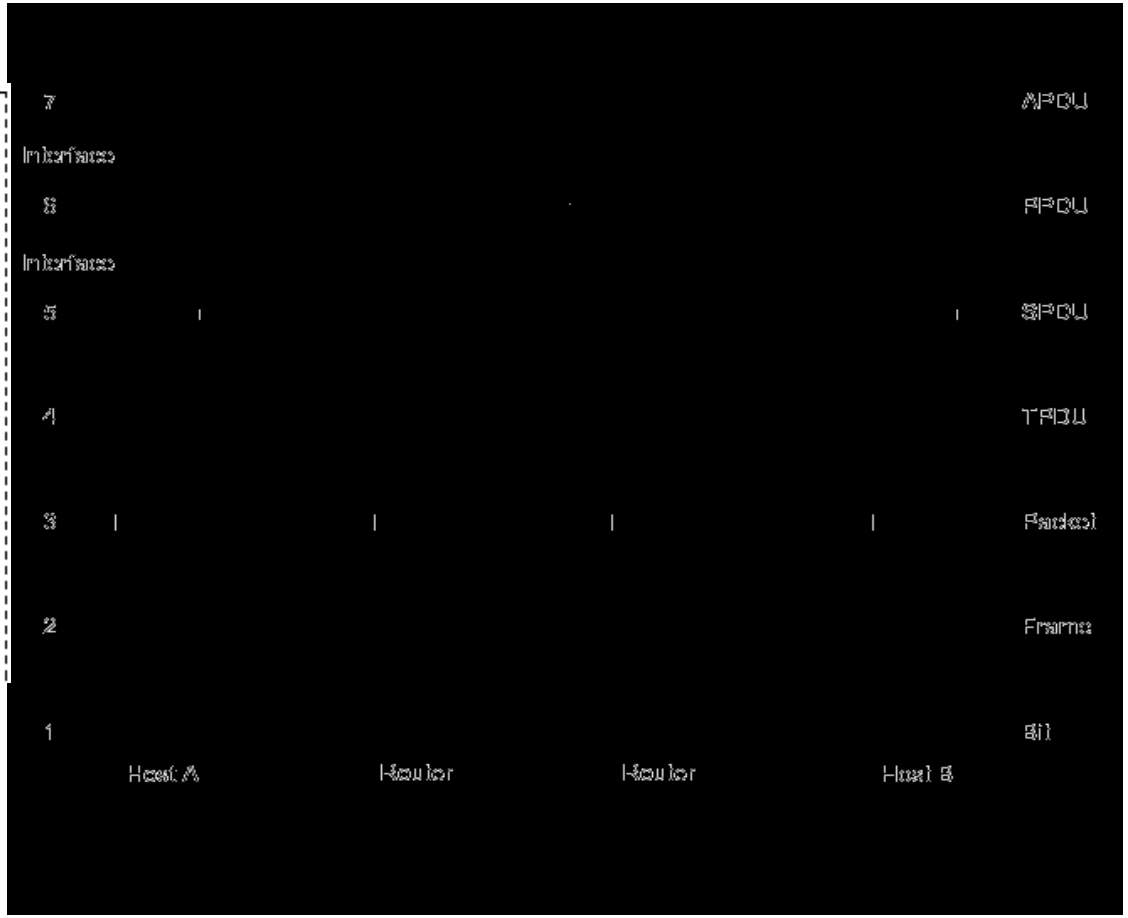
Đóng gói dữ liệu trên mạng

Mô hình OSI


Host A
Host B
Application
Presentation
Session
Transport
Network
Data Link
Physical
Data
Segments
Packets
Frames
Bits
Application
Presentation
Session
Transport
Network
Data Link
Physical

Mô hình OSI


AP
AP
chức năng ứng dụng
chức năng ứng dụng
chức năng mạng
chức năng mạng
Mạng dữ liệu
Môi trường mạng
Môi trường OSI
Môi trường hệ thống thực

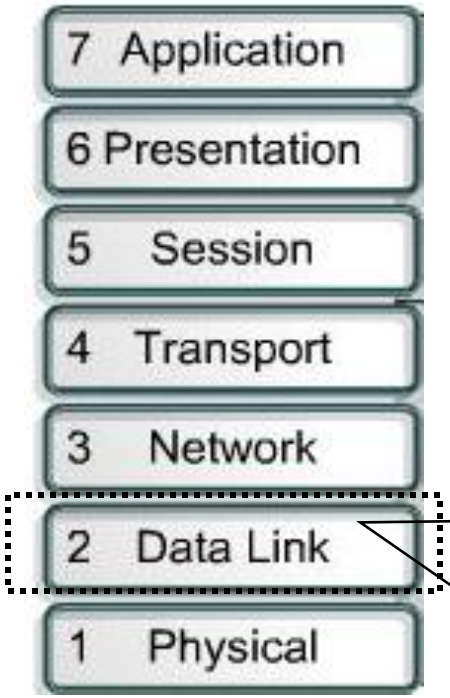

7
APDU
Interface
6
PPDU
Interface
5
SPDU
4
TPDU
3
Packet
2
Frame
1
Bit
Host A
Router
Router
Host B

Mô hình OSI

Truyền dẫn nhị phân

- Dây, đầu nối, điện áp
- Tốc độ truyền dữ liệu
- Phương tiện truyền dẫn
- Chế độ truyền dẫn (simplex, half-duplex, full-duplex)

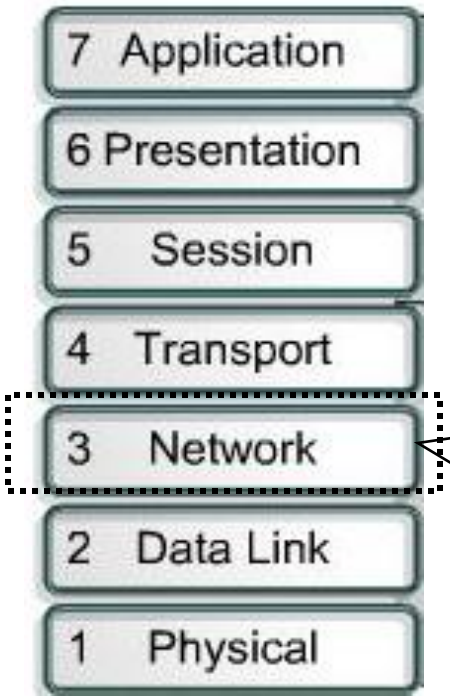
Mô hình OSI



Điều khiển liên kết, truy xuất đường truyền

- Đóng Frame
- Ghi địa chỉ vật lý
- Điều khiển luồng
- Kiểm soát lỗi, thông báo lỗi

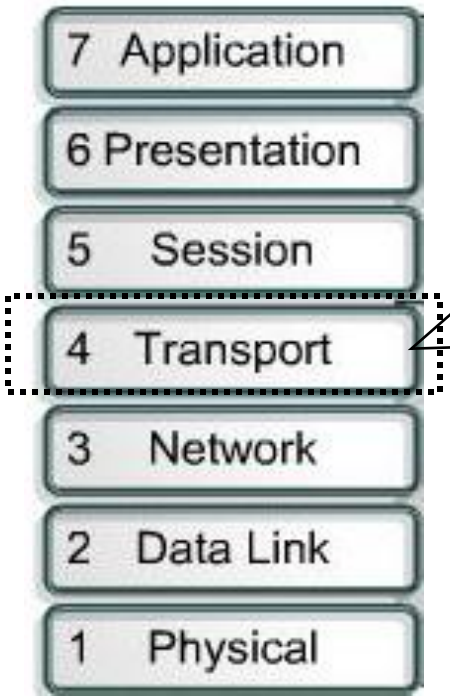
Mô hình OSI



Địa chỉ mạng và xác định đường đi tốt nhất

- Tin cậy
- Địa chỉ luận lý, topo mạng
- Định tuyến (tìm đường đi) cho gói tin

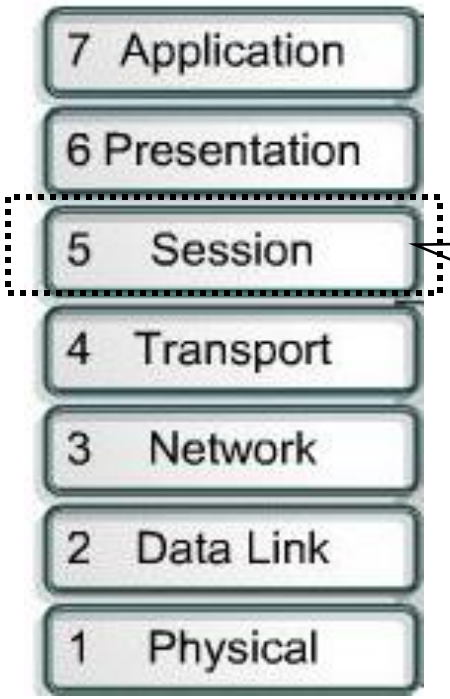
Mô hình OSI



Kết nối end-to-end

- Vận chuyển giữa các host
- Vận chuyển tin cậy
- Thiết lập, duy trì, kết nối các mạch ảo
- Phát hiện lỗi, phục hồi thông tin và điều khiển luồng

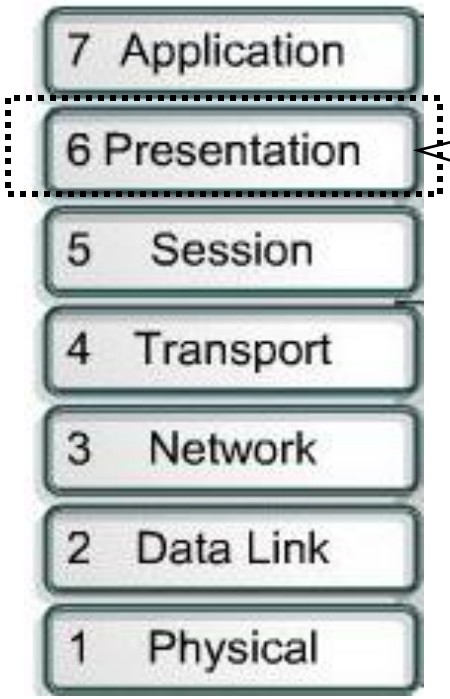
Mô hình OSI



Truyền thông liên host

- Thiết lập, quản lý và kết thúc các phiên giữa các ứng dụng

Mô hình OSI



Trình bày dữ liệu

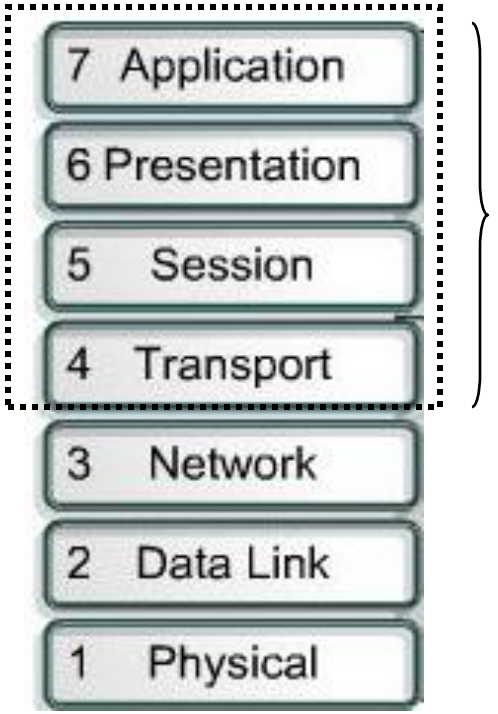
- Định dạng dữ liệu
- Cấu trúc dữ liệu
- Mã hóa
- Nén dữ liệu

Mô hình OSI

Các quá trình mạng của ứng dụng

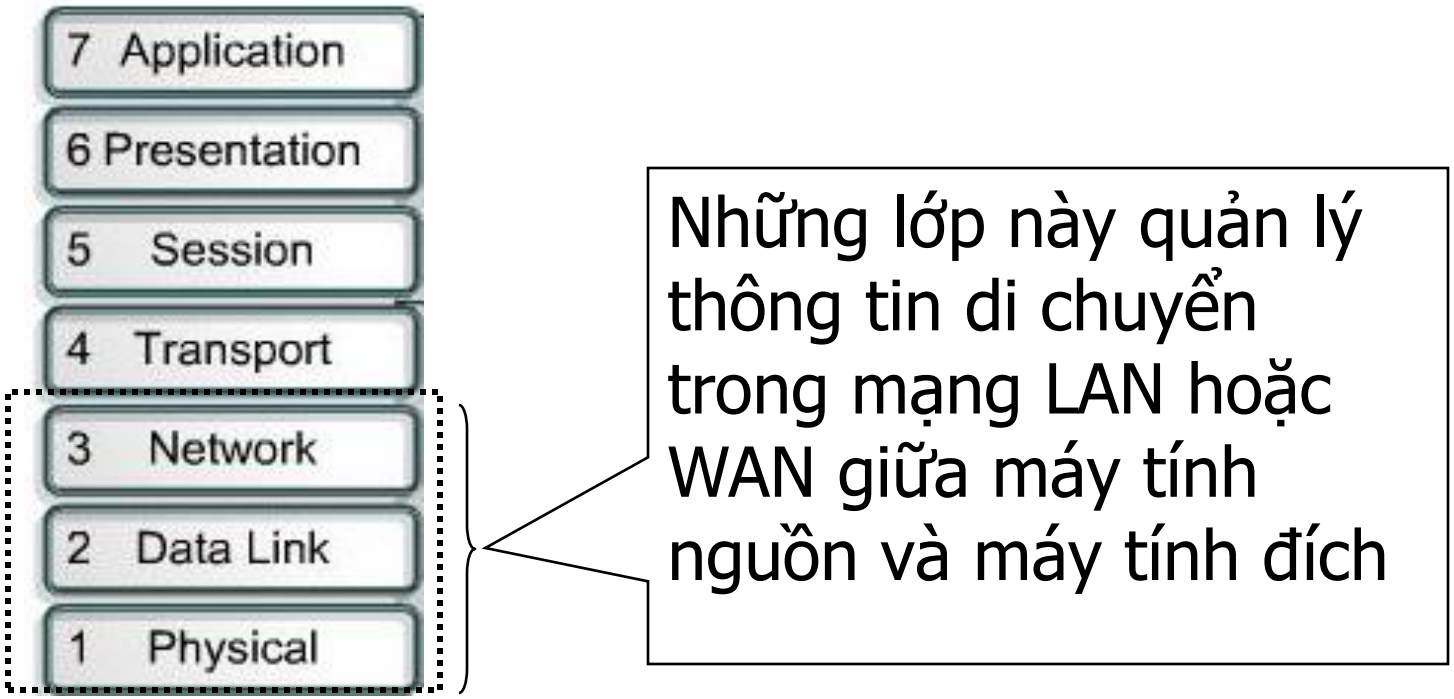
- Xác định giao diện giữa người sử dụng và môi trường OSI
- Cung cấp các dịch vụ mạng cho các ứng dụng như email, truyền file…

Mô hình OSI

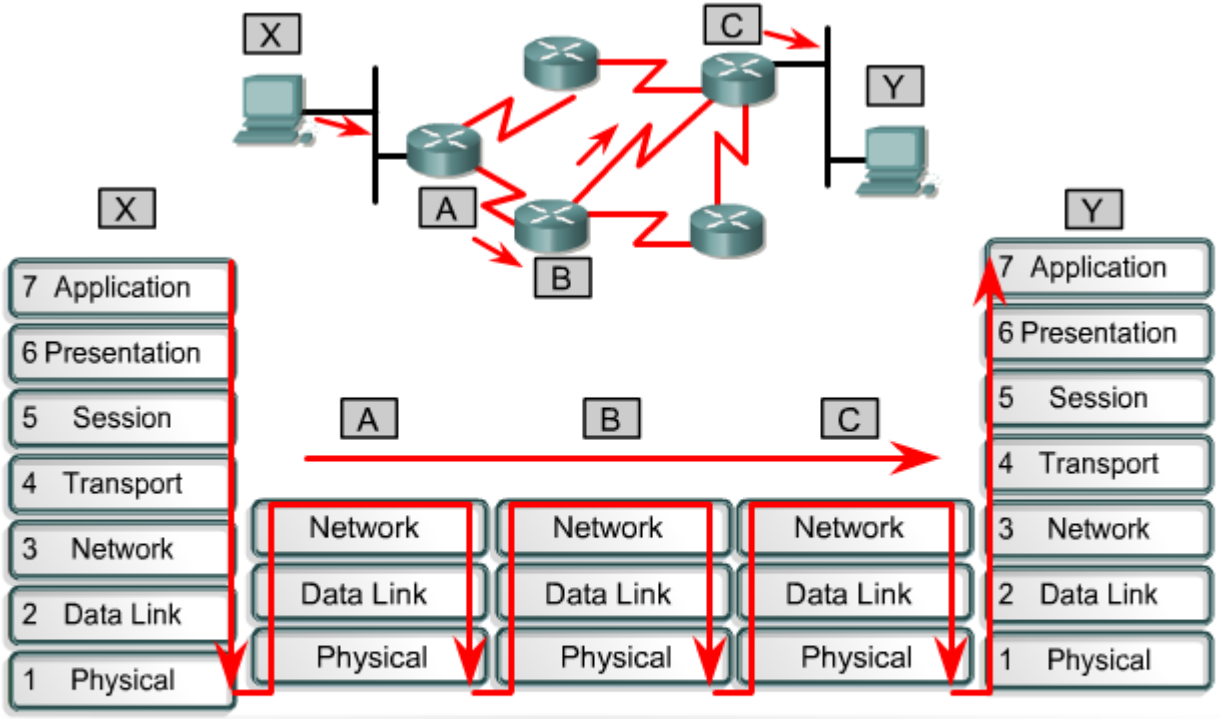


Những lớp này chỉ tồn tại trong máy tính nguồn và máy tính đích

Mô hình OSI


7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data Link
1 Physical
Những lớp này quản lý thông tin di chuyển trong mạng LAN hoặc WAN giữa máy tính nguồn và máy tính đích

Dòng dữ liệu trên mạng



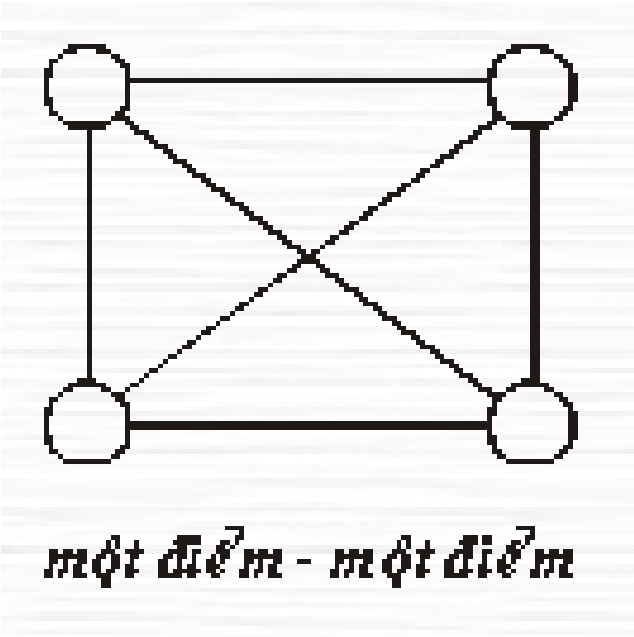
Data flow in a network focuses on layers one, two and three of the OSI model. This is after being transmitted by the sending host and before arriving at the receiving host.

CHƯƠNG 2:
CẤU TRÚC MẠNG (TOPOLOGY)

- Phương thức nối mạng
- Cấu trúc vật lý của mạng
- Giao thức truy cập đường truyền trên mạng LAN

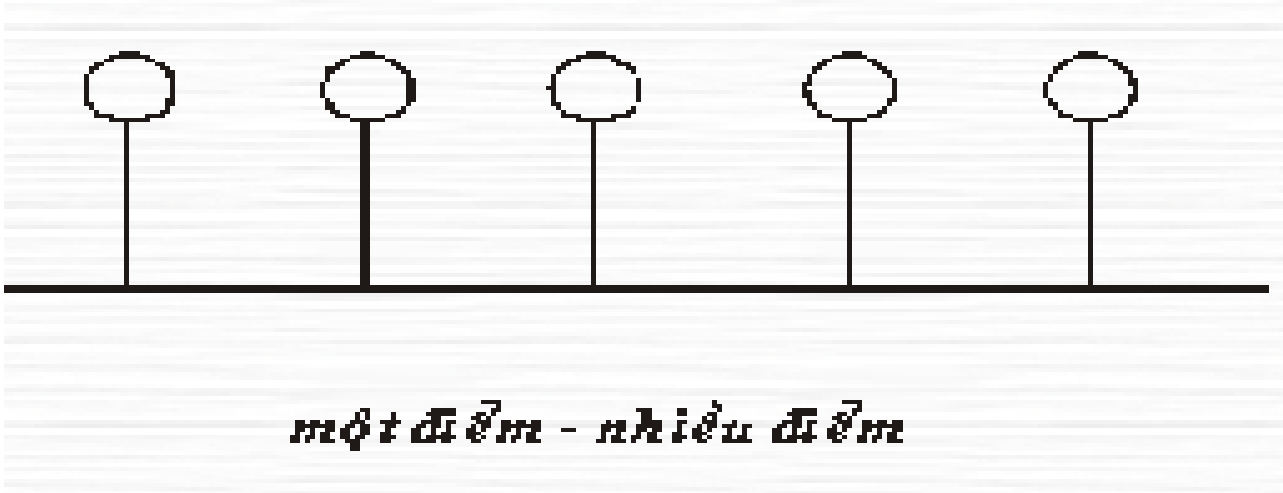
Phương thức nối mạng

- Point-to-point (điểm – điểm): các đường truyền riêng biệt được thiết lập để nối các cặp máy tính lại với nhau.

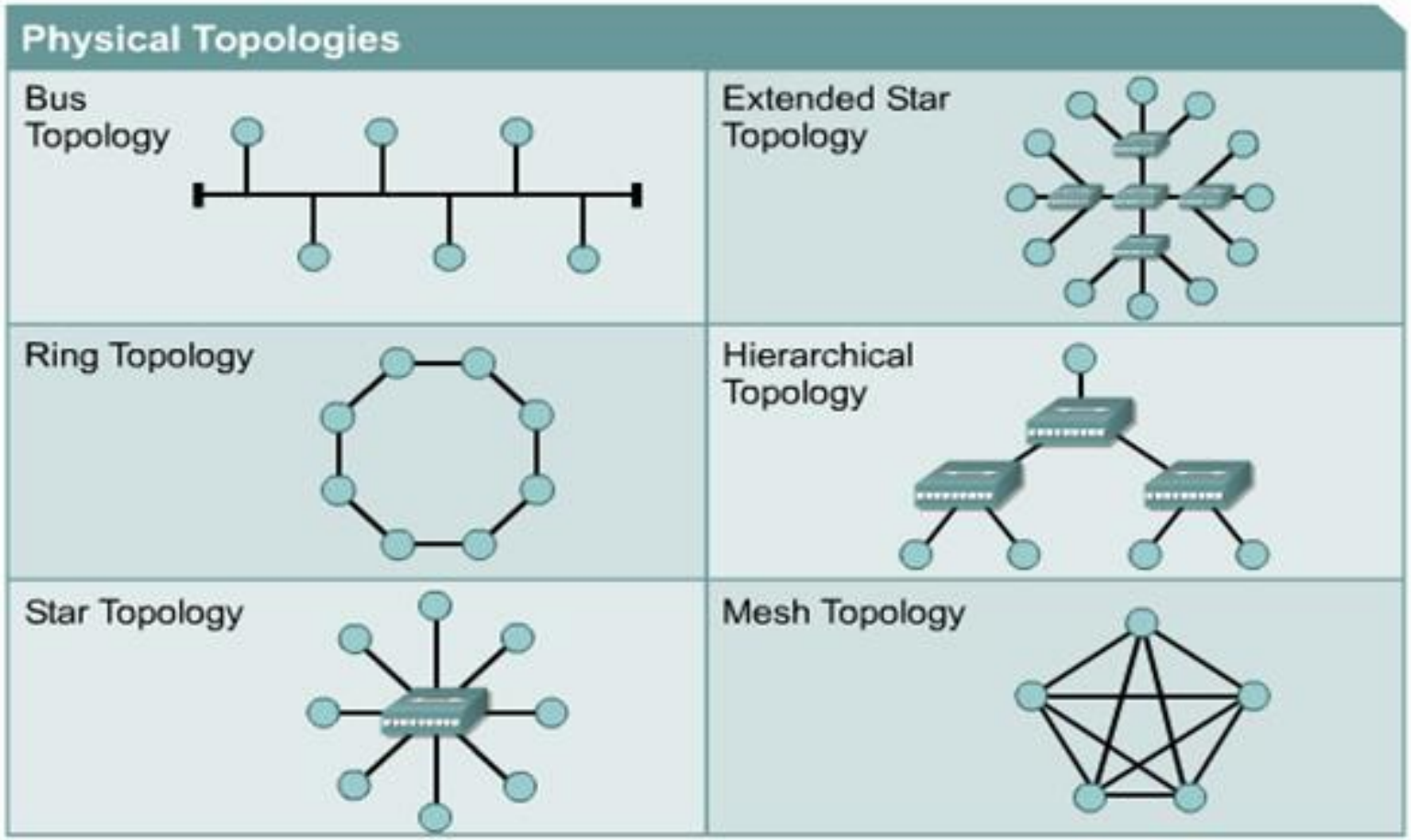


Phương thức nối mạng

- Broadcast (một điểm - nhiều điểm): tất cả các trạm phân chia chung một đường truyền vật lý.



Cấu trúc vật lý của mạng LAN

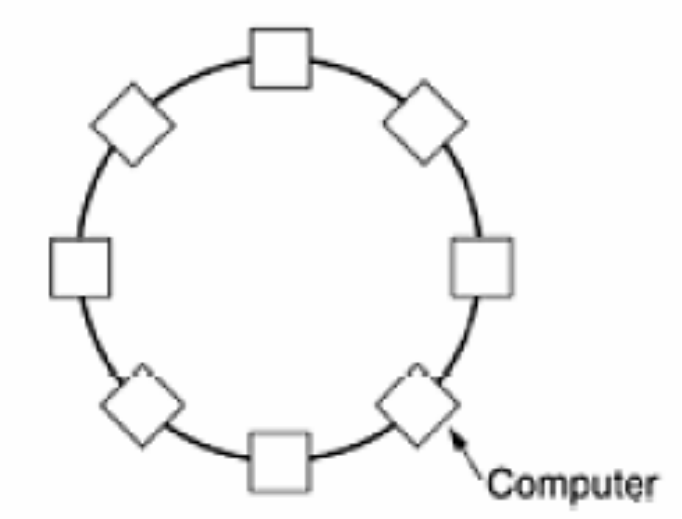


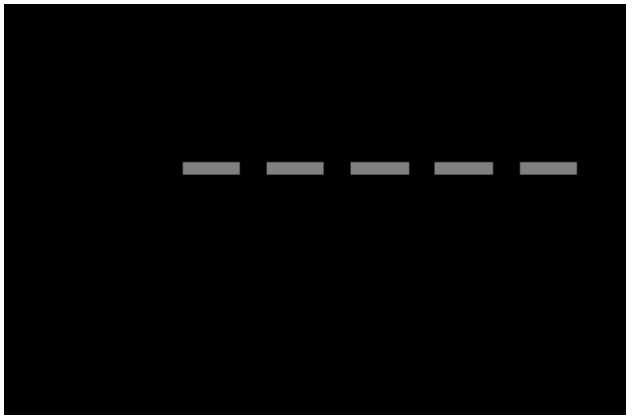
Dạng đường thẳng (Bus Topology)

- Ưu điểm
  - Dễ dàng cài đặt và mở rộng
  - Chi phí thấp
  - Một máy hỏng không ảnh hưởng đến các máy khác.
- Hạn chế
  - Khó quản trị và tìm nguyên nhân lỗi
  - Giới hạn chiều dài cáp và số lượng máy tính
  - Hiệu năng giảm khi có máy tính được thêm vào
  - Một đoạn cáp backbone bị đứt sẽ ảnh hưởng đến toàn mạng

Dạng vòng tròn (Ring Topology)

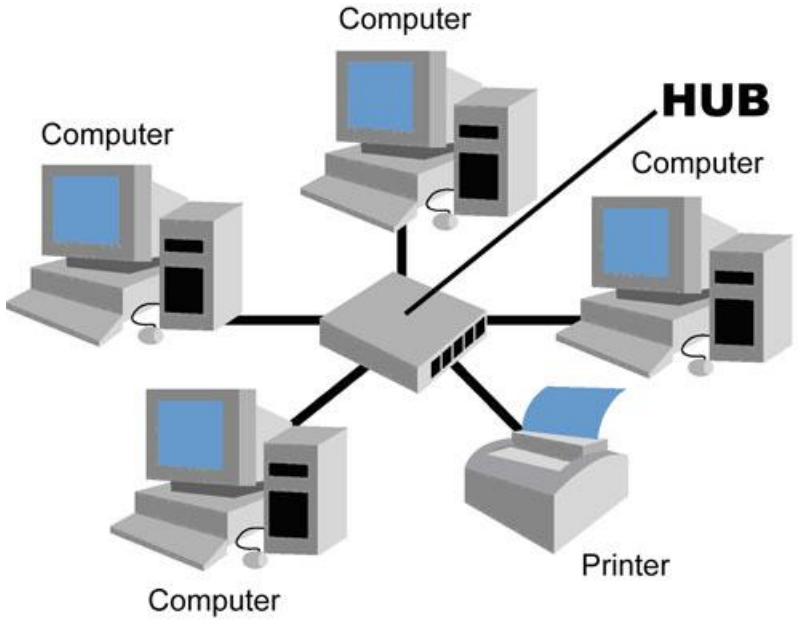
- Ưu điểm
  - Sự phát triển của hệ thống không tác động đáng kể đến hiệu năng
  - Tất cả các máy tính có quyền truy cập như nhau
- Hạn chế
  - Chi phí thực hiện cao
  - Phức tạp
  - Khi một máy có sự cố thì có thể ảnh hưởng đến các máy tính khác





Dạng hình sao (Star Topology)

- Ưu điểm
  - Dễ dàng bổ sung hay loại bỏ bớt máy tính
  - Dễ dàng theo dõi và giải quyết sự cố
  - Có thể phù hợp với nhiều loại cáp khác nhau
- Hạn chế
  - Khi hub không làm việc, toàn mạng cũng sẽ không làm việc
  - Sử dụng nhiều cáp



Giao thức truy cập đường truyền trên mạng LAN

Hai loại giao thức: ngẫu nhiên và có điều khiển

- Ngẫu nhiên
  - Giao thức chuyển mạch
  - Giao thức đường dây đa truy cập với cảm nhận va chạm
- Có điều khiển
  - Giao thức dùng thẻ bài vòng (Token Ring)
  - Giao thức dùng thẻ bài cho dạng đường thẳng (Token Bus)

Giao thức truy cập đường truyền trên mạng LAN

- Giao thức chuyển mạch (yêu cầu và chấp nhận)

  Khi máy tính yêu cầu, nó sẽ được thâm nhập vào đường cáp nếu mạng không bận, ngược lại sẽ bị từ chối.

Giao thức truy cập đường truyền trên mạng LAN

- Giao thức đường dây đa truy cập với cảm nhận va chạm (Carrier Sense Multiple Access/with Collision Detection)

  Gói dữ liệu chỉ được gởi nếu đường truyền rảnh, ngược lại mỗi trạm phải đợi theo một trong 3 phương thức:

  – Chờ đợi một thời gian ngẫu nhiên rồi lại bắt đầu kiểm tra đường truyền
  – Kiểm tra đường truyền liên tục cho đến khi đường truyền rảnh
  – Kiểm tra đường truyền với xác suất p (0<p<1)

Giao thức truy cập đường truyền trên mạng LAN

- Giao thức dùng thẻ bài vòng (Token Ring)
  - Thẻ bài là một đơn vị dữ liệu đặc biệt có một bit biểu diễn trạng thái bận hoặc rảnh.
  - Thẻ bài chạy vòng quanh trong mạng.
  - Trạm nào nhận được thẻ bài rảnh thì có thể truyền dữ liệu.
- Giao thức dùng thẻ bài cho dạng đường thẳng (Token bus)

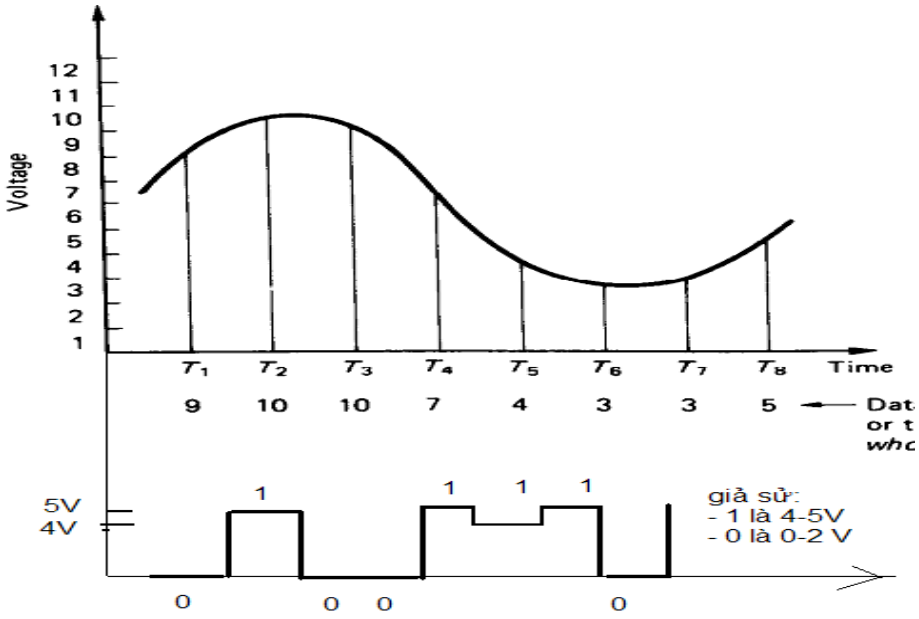
  Tạo ra một vòng logic (vòng ảo) và thực hiện giống Token Ring.

CHƯƠNG 3:
PHƯƠNG TIỆN TRUYỀN DẪN VÀ CÁC THIẾT BỊ LIÊN KẾT MẠNG

- Môi trường truyền dẫn
- Phương tiện truyền dẫn
- Các thiết bị liên kết mạng

Môi trường truyền dẫn

- Là phương tiện vật lý cho phép truyền tải tín hiệu giữa các thiết bị.
- Hai loại phương tiện truyền dẫn chính:
  - Hữu tuyến
  - Vô tuyến
- Hệ thống sử dụng hai loại tín hiệu:
  - Digital
  - Analog



Các đặc tính của phương tiện truyền dẫn

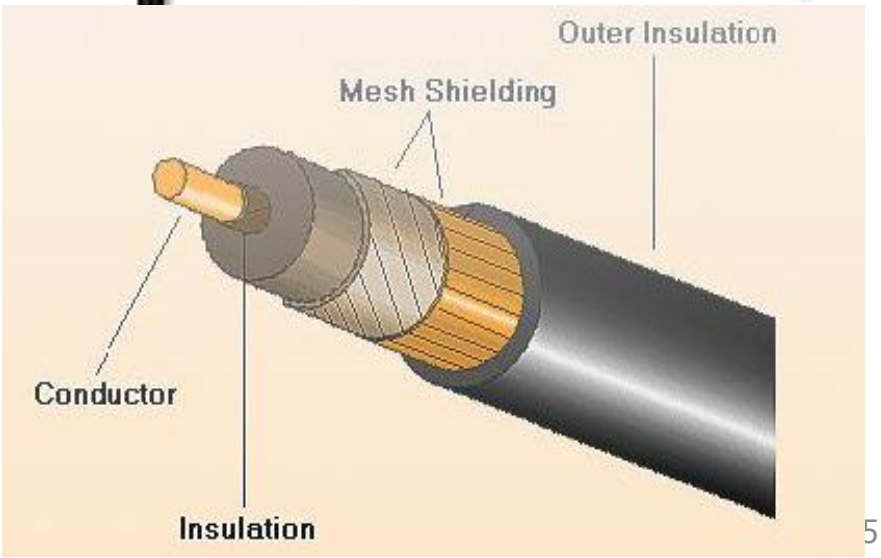
- Chi phí
- Yêu cầu cài đặt
- Băng thông (bandwidth).
- Băng tần (baseband, broadband)
- Độ suy dần (attenuation).
- Nhiễu điện từ (Electronmagnetic Interference - EMI)
- Nhiễu xuyên kênh (crosstalk)

Phương tiện truyền dẫn

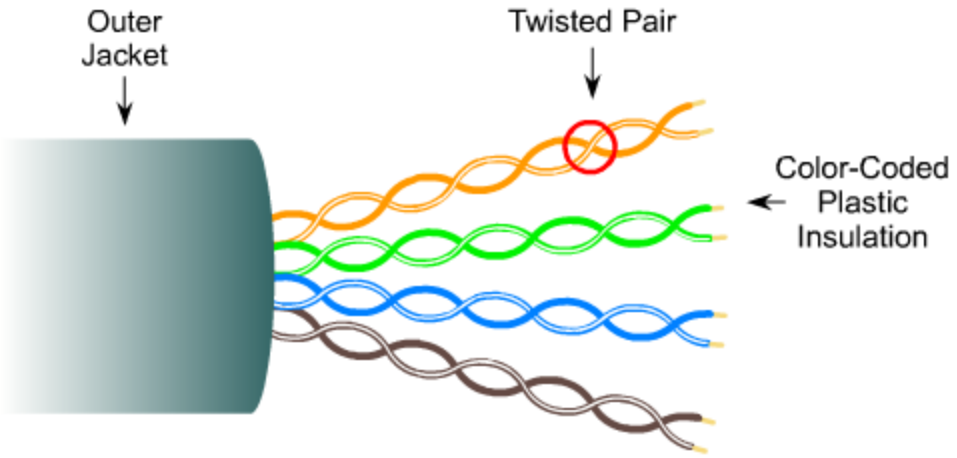
- Cáp đồng trục
- Cáp xoắn đôi
- Cáp quang
- Wireless

Cáp đồng trục (coaxial)

- Cấu tạo
- Phân loại
  - Thinnet/Thicknet
  - Baseband/ Broadband
- Thông số kỹ thuật
  - Chiều dài cáp
  - Tốc độ truyền
  - Nhiễu
  - Lắp đặt/bảo trì
  - Giá thành
  - Kết nối

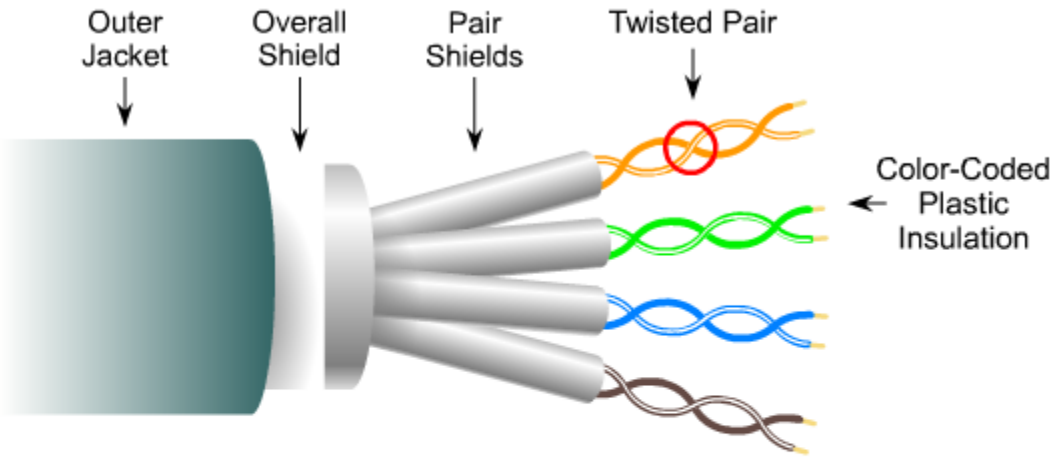


Cáp xoắn đôi
Unshielded Twisted Pair (UTP) Cable



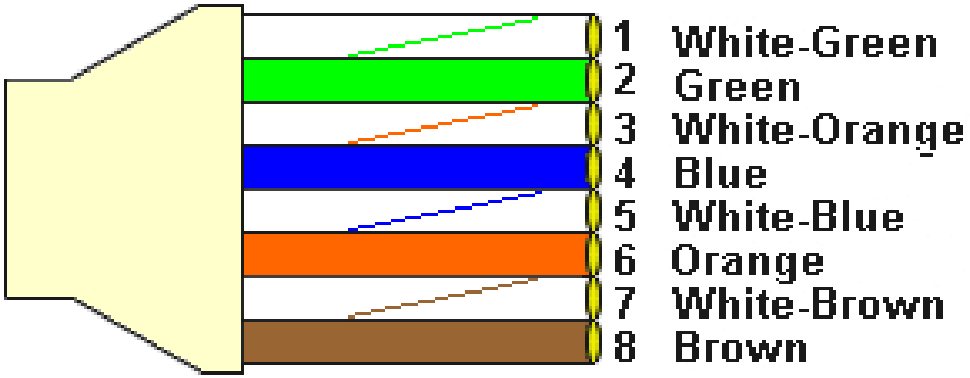
- Speed and throughput: 10 - 100 - 1000 Mbps (depending on the quality/category of cable)
- Average $ per node: Least Expensive
- Media and connector size: Small
- Maximum cable length: 100m

Cáp xoắn đôi
Shielded Twisted Pair (STP) Cable

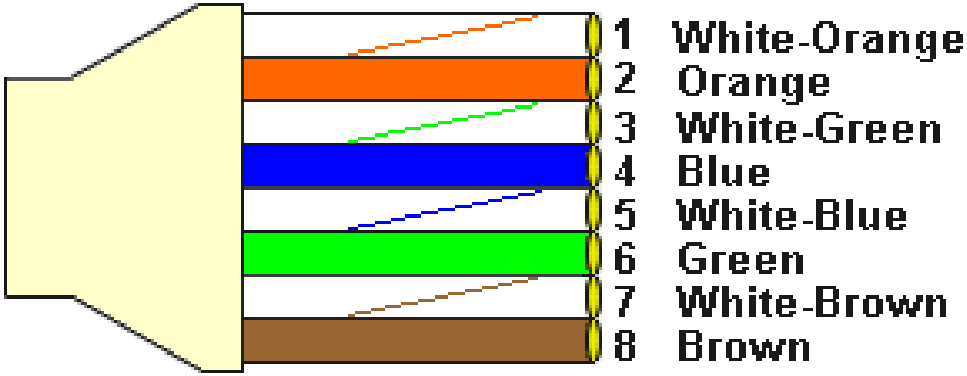


- Speed and throughput: 10 - 100 Mbps
- Average $ per node: Moderately Expensive
- Media and connector size: Medium to Large
- Maximum cable length: 100m

Chuẩn cáp 568A & 568B

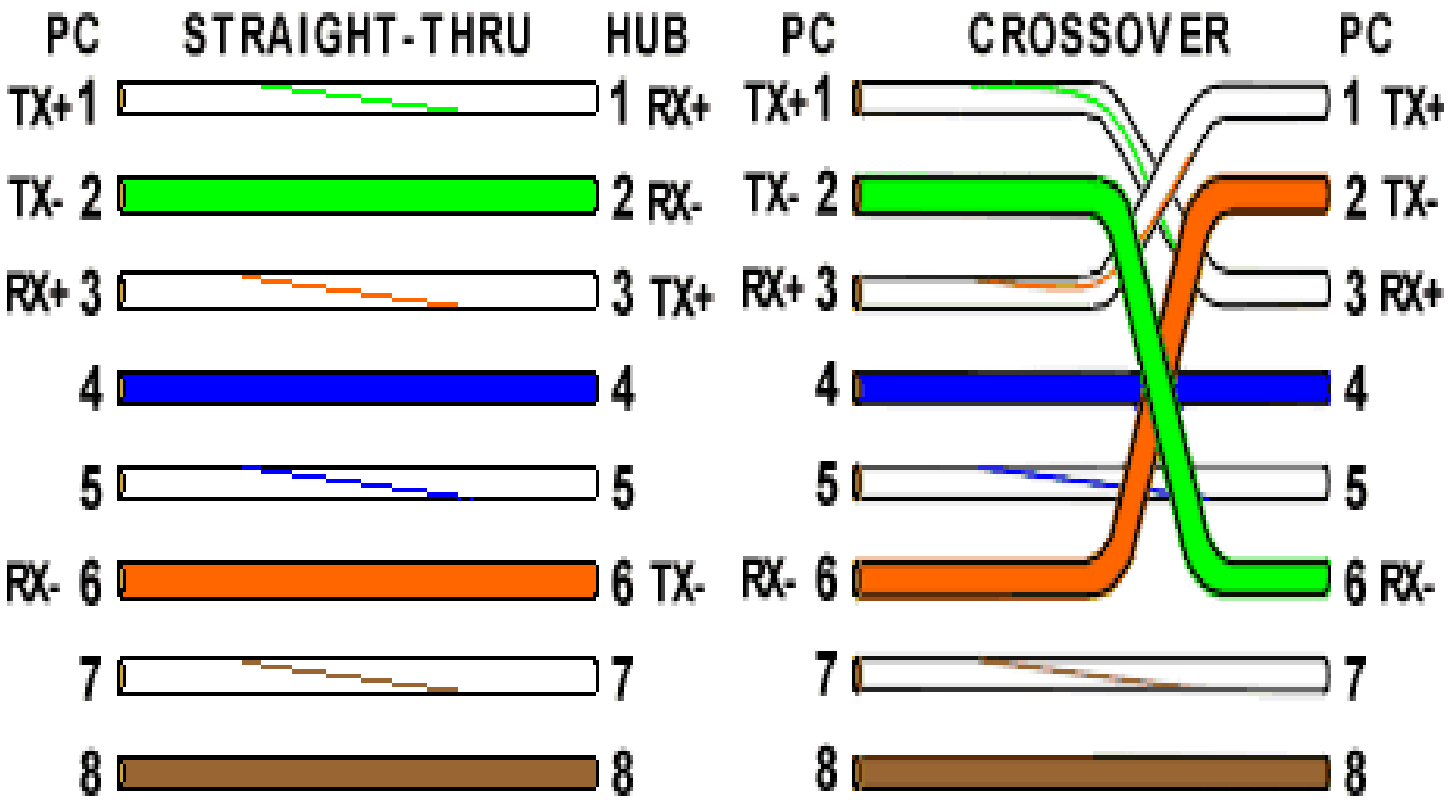


568A CABLE END



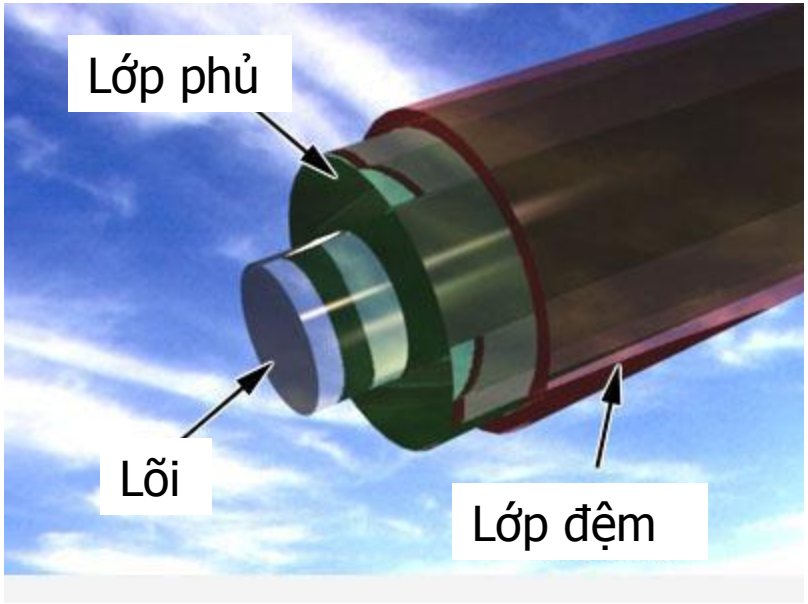
568B CABLE END

Phương thức bấm Cáp

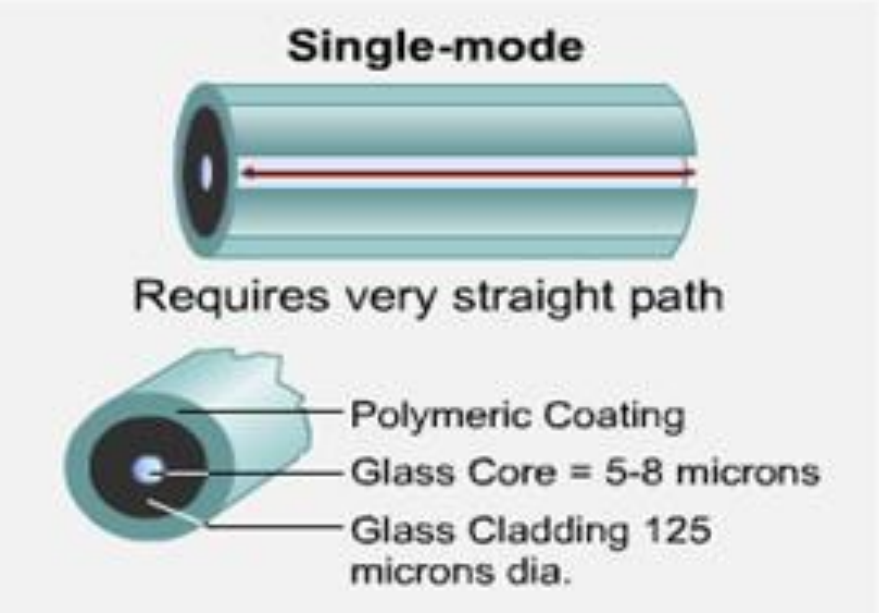


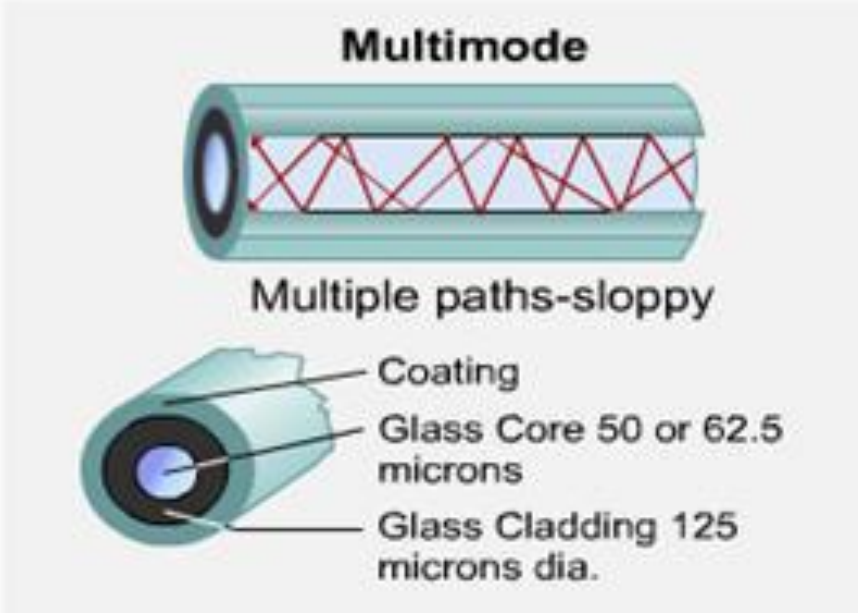
Cáp quang (Fiber optic)

- Thành phần & cấu tạo
  - Dây dẫn
  - Nguồn sáng (LED, Laser)
  - Đầu phát hiện (Photodiode, photo transistor)
- Phân loại
  - Multimode stepped index
  - Multimode graded index
  - Single mode (mono mode)
- Thông số kỹ thuật
  - Chiều dài cáp
  - Tốc độ truyền
  - Nhiễu
  - Lắp đặt/bảo trì
  - Giá thành
  - Kết nối



Cáp quang (Fiber optic)



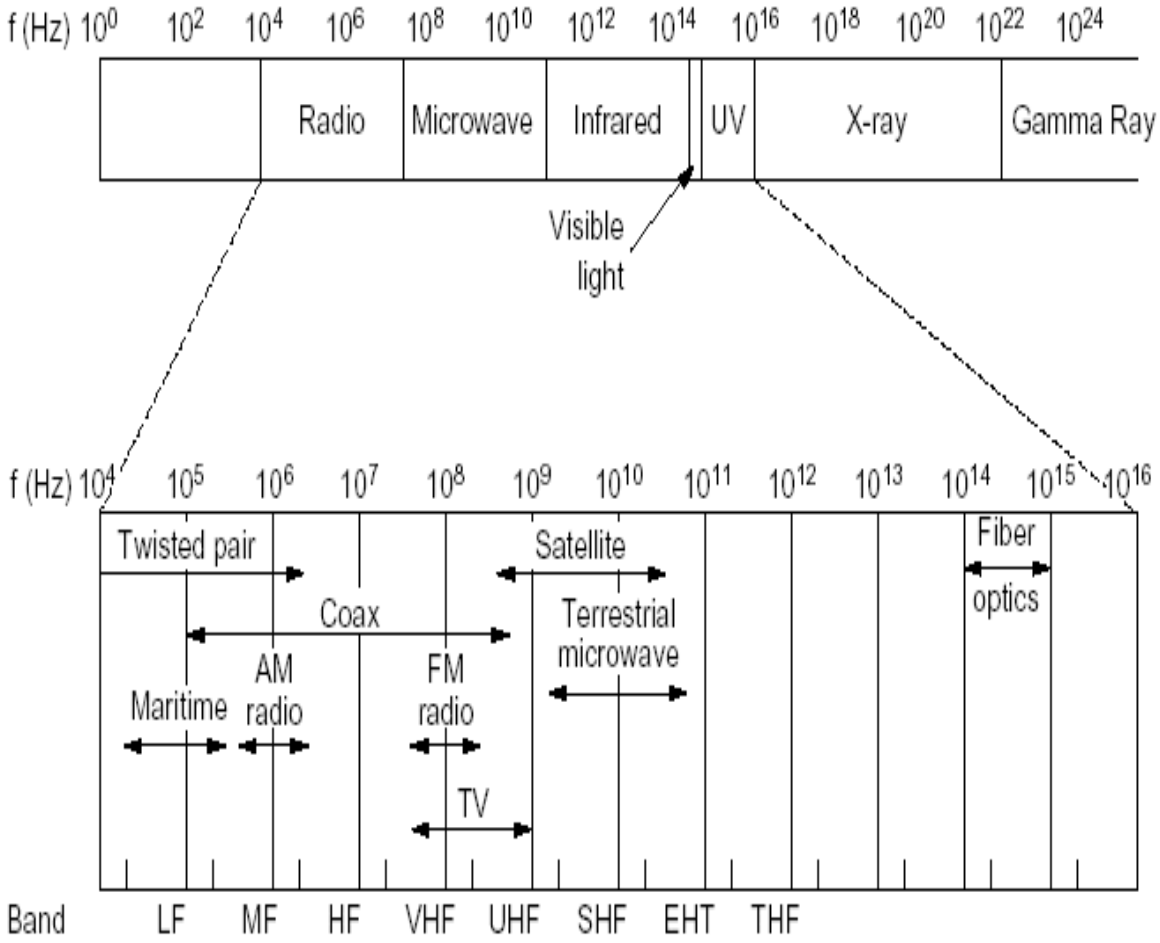


Thông số cơ bản của các loại cáp

Cáp	Chiều dài cáp tối đa	Tốc độ truyền	Lắp đặt	Nhiễu	Giá thành
UTP	100 m	10-100 Mbps	Dễ	Cao	Thấp nhất
STP	100 m	16-500 Mbps	Khá dễ	Thấp	Vừa phải
Thinnet	185 m	10 Mbps	Dễ	Thấp	Thấp
Thicknet	500 m	10 Mbps	Khó	Thấp	Cao
Fiber optics	2000 m	2 Gbps	Khó	Không	Đắt

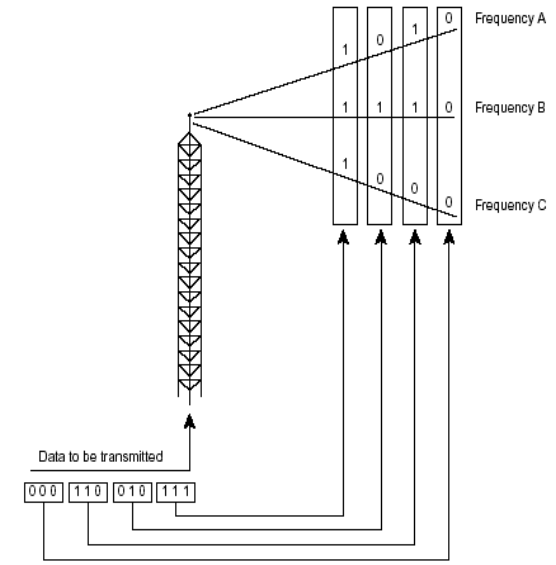
Wireless

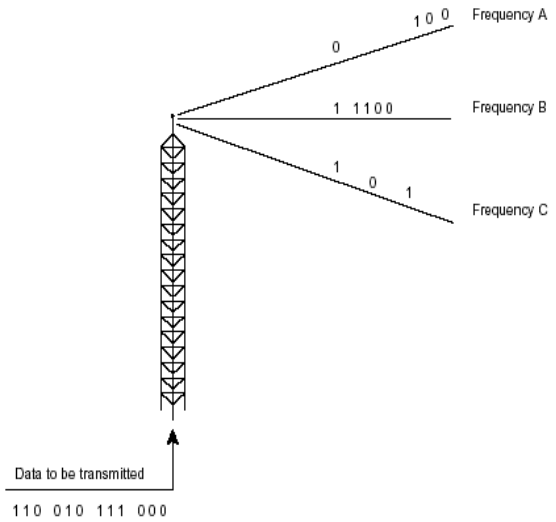
- Wireless?
- Các kỹ thuật
  - Radio
  - Microwave
  - Infrared
  - Lightwave



Radio

- Đặc điểm
  - Tần số
  - Thiết bị: antenna, transceiver
- Phân loại
  - Single-Frequency
    - Low power
    - High power
  - Spread-Spectrum
    - Direct-sequence modulation
    - Frequency-hopping





Microwave (sóng cực ngắn)

- Đặc điểm
- Phân loại
  - Terrestrial Microwave
  - Satellite Microwave
- Thông số

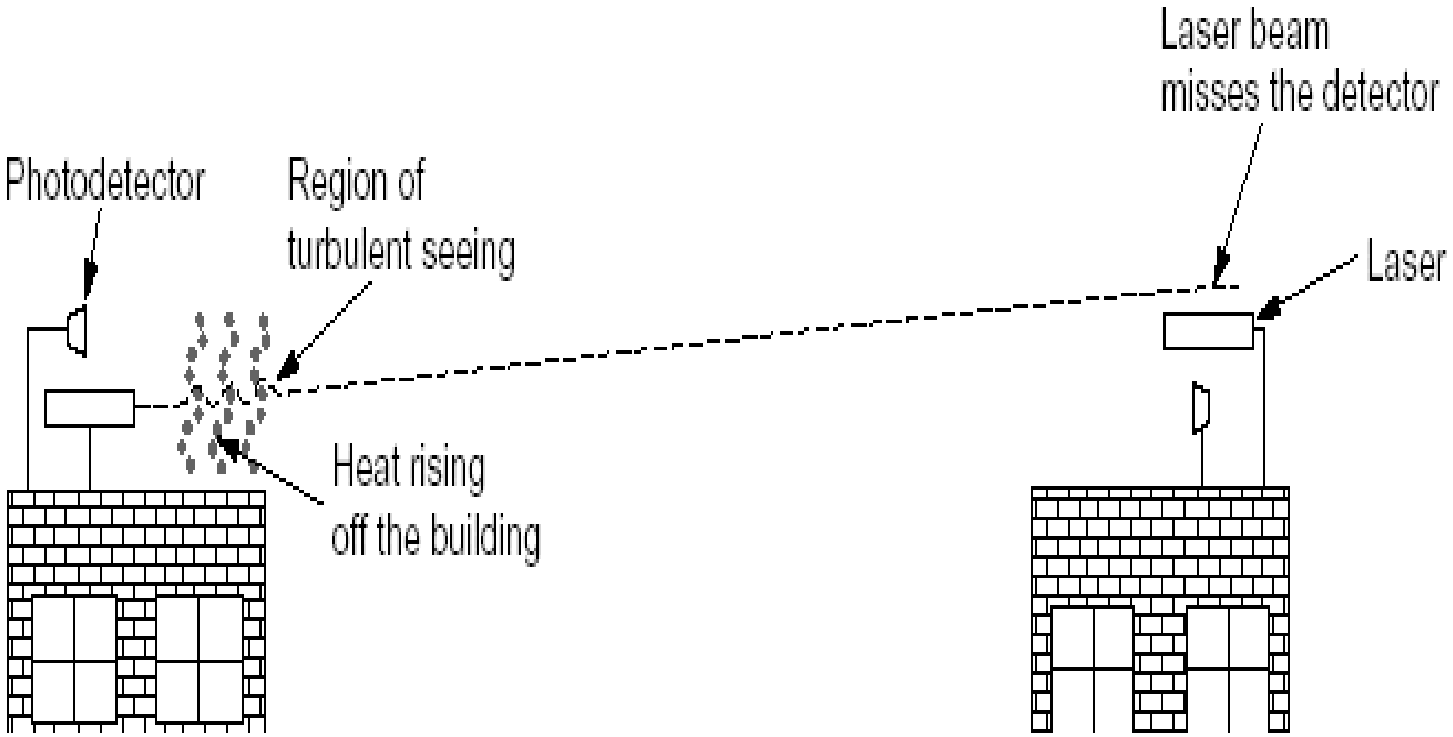
	Terrestrial Microwave	Satellite Microwave
Tần số	4-6 GHz, 21-23 GHz	11-14 GHz
Khoảng cách tối đa	Phụ thuộc công suất và tần số phát (có thể vài chục km)	Toàn cầu
Tốc độ truyền	1 – 10 Mbps	1 – 10 Mbps
Lắp đặt / bảo trì	Khá khó	Khó
Nhiễu	Phụ thuộc thiết bị, thời tiết ...	Phụ thuộc thiết bị, thời tiết ...
Giá	Khá cao	Rất cao
Bảo mật	Thấp (thường được mã hoá)	Thấp (thường được mã hoá)

Infrared (Sóng hồng ngoại)

- Đặc điểm
- Phân loại
  - Point-to-point Infrared
  - Broadcast Infrared
- Thông số

	Point-to-point Infrared	Broadcast Microwave
Tần số	100-1000 GHz	100 GHz - 1000 THz
Khoảng cách tối đa	Có thể vài km	Vài chục mét
Tốc độ truyền	100 Kbps – 16 Mbps	Nhỏ hơn 1 Mbps
Lắp đặt / bảo trì	Vừa phải	Dễ
Nhiễu	Chống nhiễu điện, bị nhiễu ánh sáng	Chống nhiễu điện, bị nhiễu ánh sáng
Giá	Tuỳ thuộc thiết bị	Không cao
Bảo mật	Cao (do line-of-sight và độ dải sáng hẹp)	Thấp

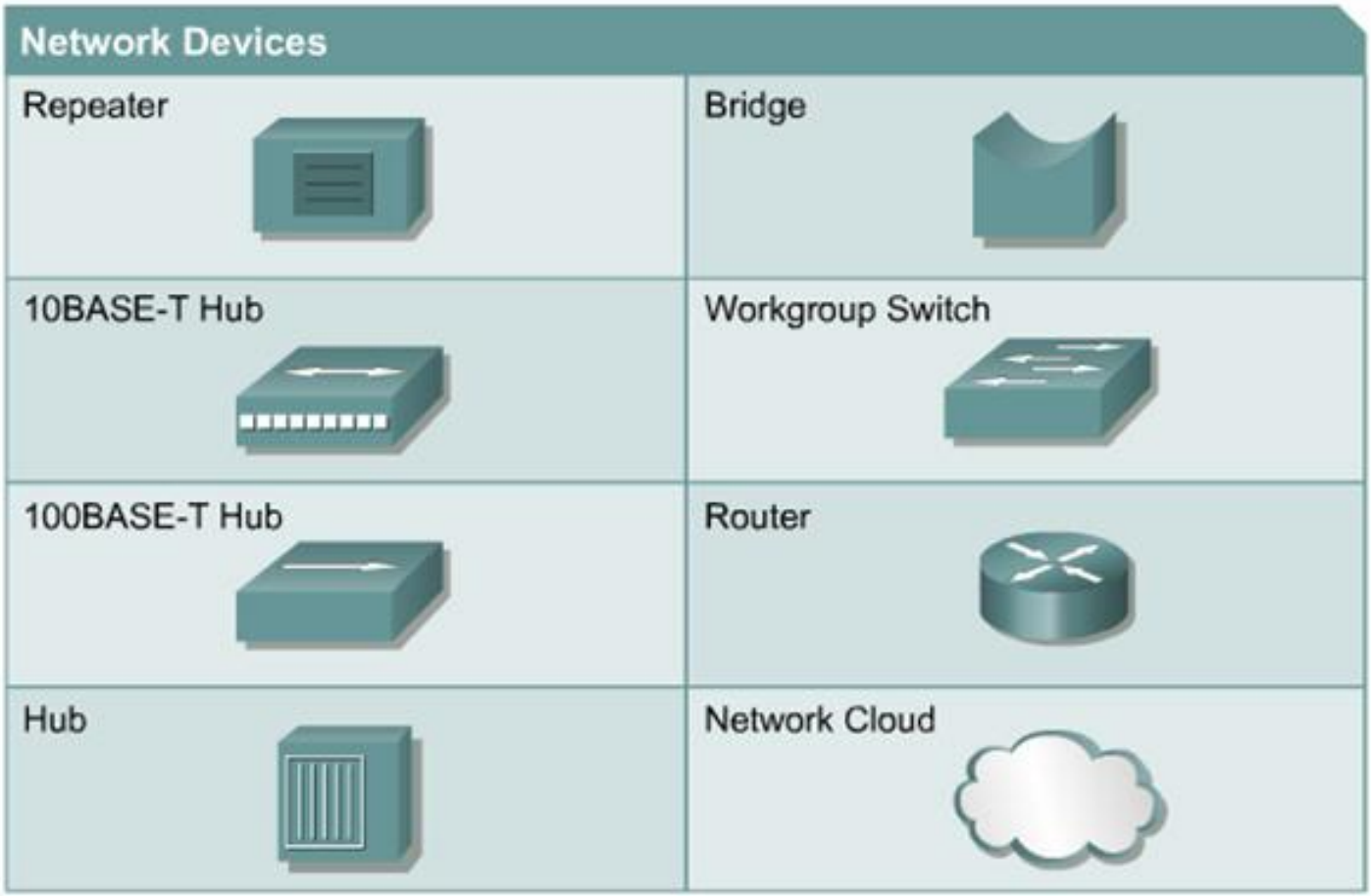
Lightwave


Laser beam
misses the detector
Photodetector
Region of
turbulent seeing
Laser
Heat rising
off the building

Các thiết bị liên kết mạng

- Card mạng (Network Interface Card - NIC)
- Modem
- Repeater (Bộ chuyển tiếp)
- Hub (Bộ tập trung)
- Bridge (Cầu nối)
- Switch (Bộ chuyển mạch)
- Router (Bộ định tuyến)
- Gateway (Cổng nối)

Biểu diễn của các thiết bị mạng trong sơ đồ mạng



Card mạng

- Kết nối giữa máy tính và cáp mạng để phát hoặc nhận dữ liệu với các máy tính khác thông qua mạng.
- Kiểm soát luồng dữ liệu giữa máy tính và hệ thống cáp.
- Mỗi NIC (Network Interface Adapter Card) có một mã duy nhất gọi là địa chỉ MAC (Media Access Control). MAC address có 6 byte, 3 byte đầu là mã số nhà sản xuất, 3 byte sau là số serial của card.

Ca

Organizational Unique Identifier (OUI)	Vendor Assigned (NIC Cards, Interfaces)
24 bits	24 bits
6 hex digits	6 hex digits
00 60 2F	3A 07 BC
Cisco	particular device

Internal network interface card

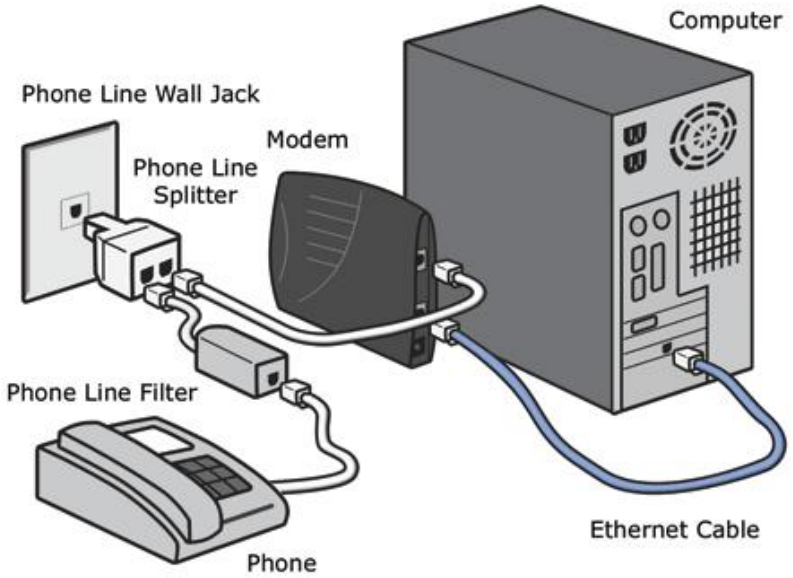
PCMCIA Network interface card

Modem

- Là tên viết tắt của hai từ điều chế (MOdulation) và giải điều chế (DEModulation).
- Điều chế tín hiệu số (Digital) sang tín hiệu tương tự (Analog) để gởi theo đường điện thoại và ngược lại.
- Có 2 loại là Internal và External.

Modem

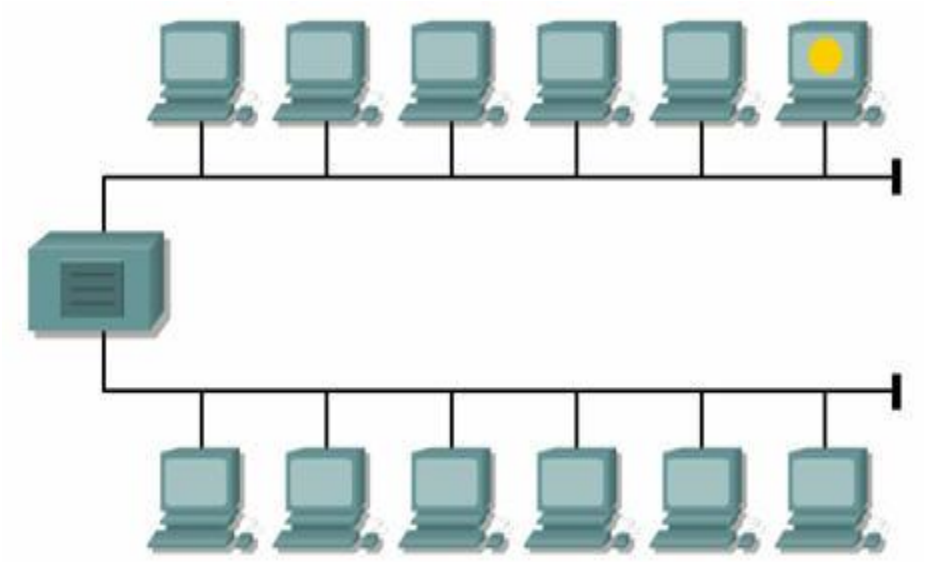

EDIMAX
ADSL ROUTER


Computer
Phone Line Wall Jack
Modem
Phone Line
Splitter
Phone Line Filter
Phone
Ethernet Cable

Repeater (bộ chuyển tiếp)

- Khuếch đại, phục hồi các tín hiệu đã bị suy thoái do tổn thất năng lượng trong khi truyền.
- Cho phép mở rộng mạng vượt xa chiều dài giới hạn của một môi trường truyền.
- Chỉ được dùng nối hai mạng có cùng giao thức truyền thông.
- Hoạt động ở lớp Physical.

Repeater (bộ chuyển tiếp)



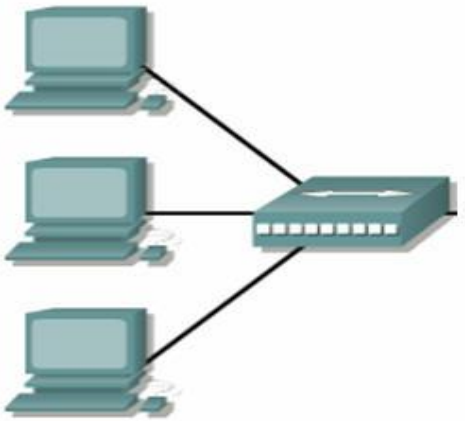
Hub (bộ tập trung)

- Chức năng như Repeater nhưng mở rộng hơn với nhiều đầu cắm các đầu cáp mạng.
- Tạo ra điểm kết nối tập trung để nối mạng theo kiểu hình sao.
- Tín hiệu được phân phối đến tất cả các kết nối.
- Có 3 loại Hub: thụ động, chủ động, thông minh.

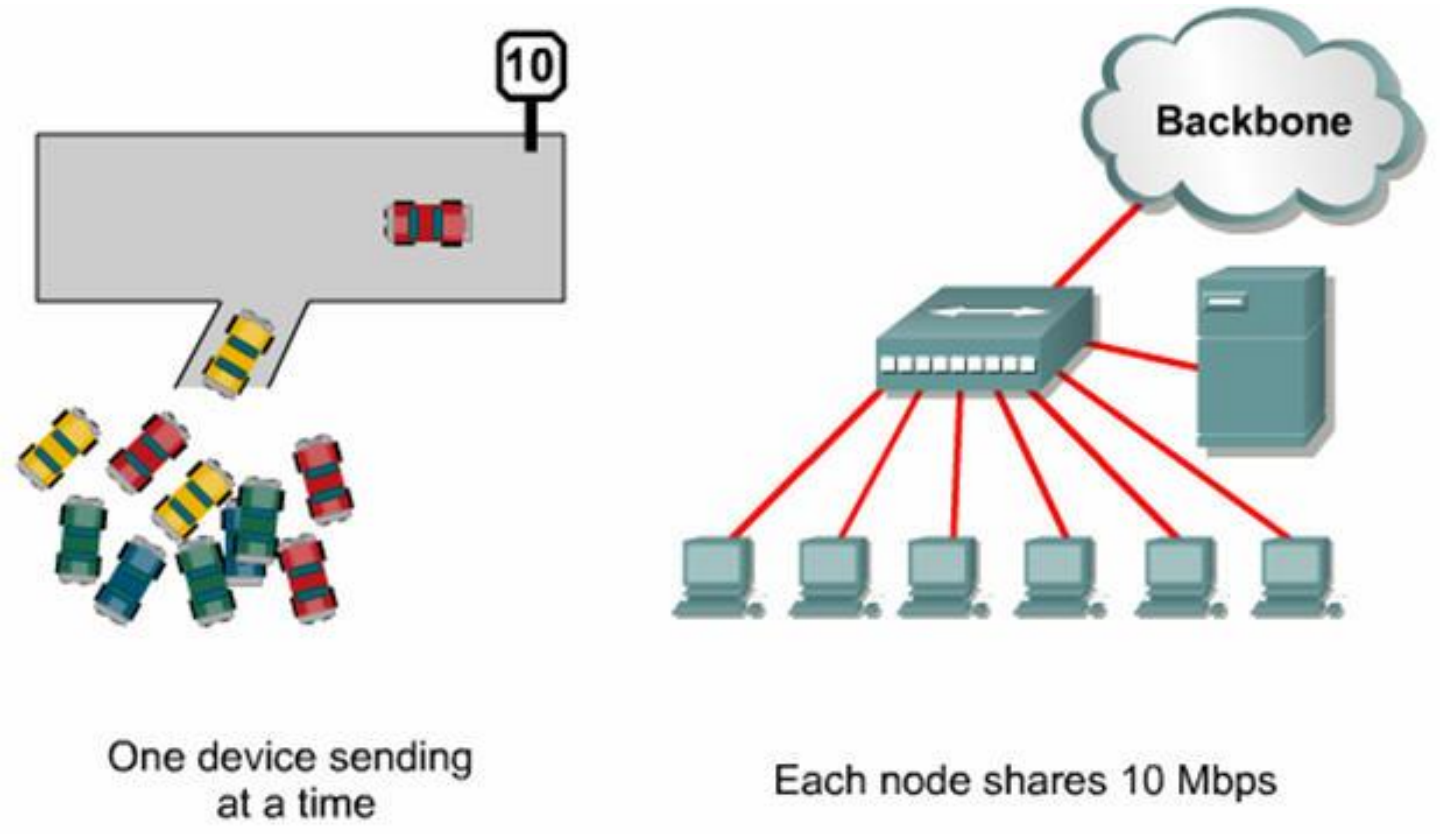
Hub (bộ tập trung)

- Hub thụ động (Passive Hub): chỉ đảm bảo chức năng kết nối, không xử lý lại tín hiệu.
- Hub chủ động (Active Hub): có khả năng khuếch đại tín hiệu để chống suy hao.
- Hub thông minh (Intelligent Hub): là Hub chủ động nhưng có thêm khả năng tạo ra các gói tin thông báo hoạt động của mình giúp cho việc quản trị mạng dễ dàng hơn.

HUB



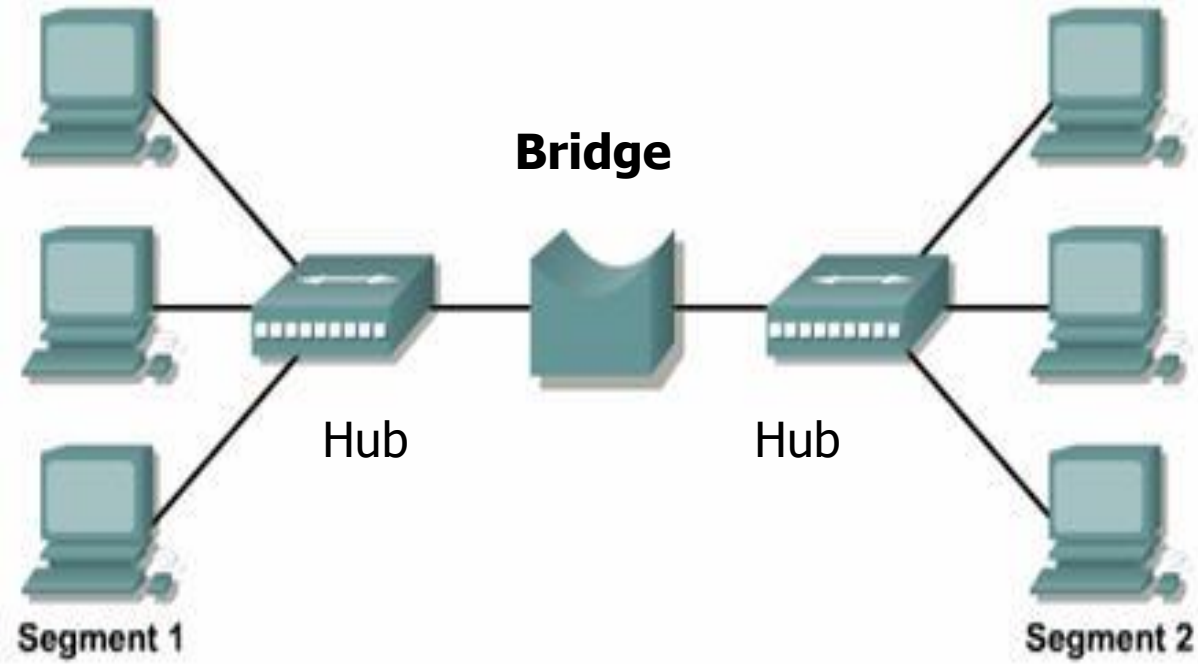
Hub (bộ tập trung)



Bridge (cầu nối)

- Dùng để nối 2 mạng có giao thức giống hoặc khác nhau.
- Chia mạng thành nhiều phân đoạn nhằm giảm lưu lượng trên mạng.
- Hoạt động ở lớp Data Link với 2 chức năng chính là lọc và chuyển vận.
- Dựa trên bảng địa chỉ MAC lưu trữ, Brigde kiểm tra các gói tin và xử lý chúng trước khi có quyết định chuyển đi hay không.

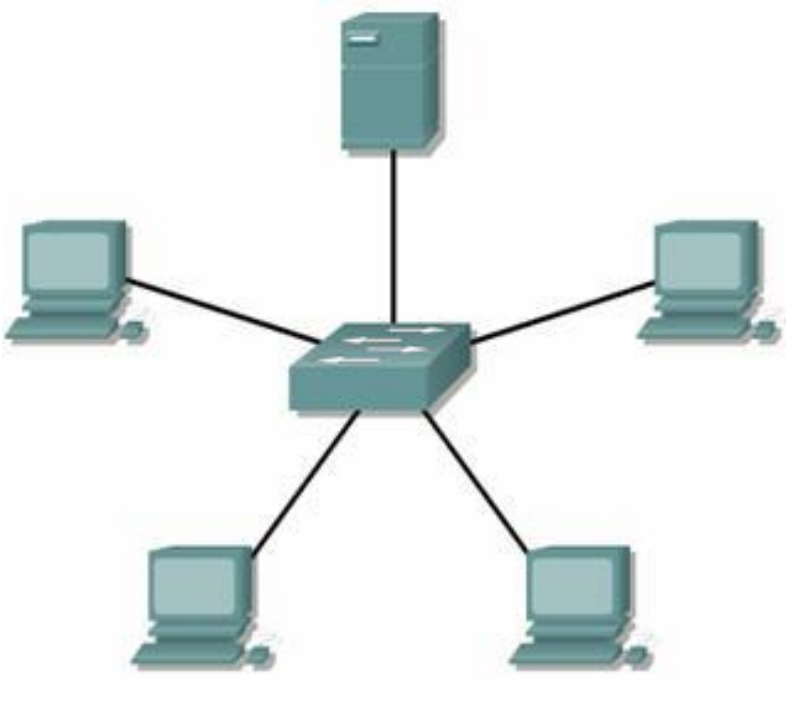
Bridge (cầu nối)



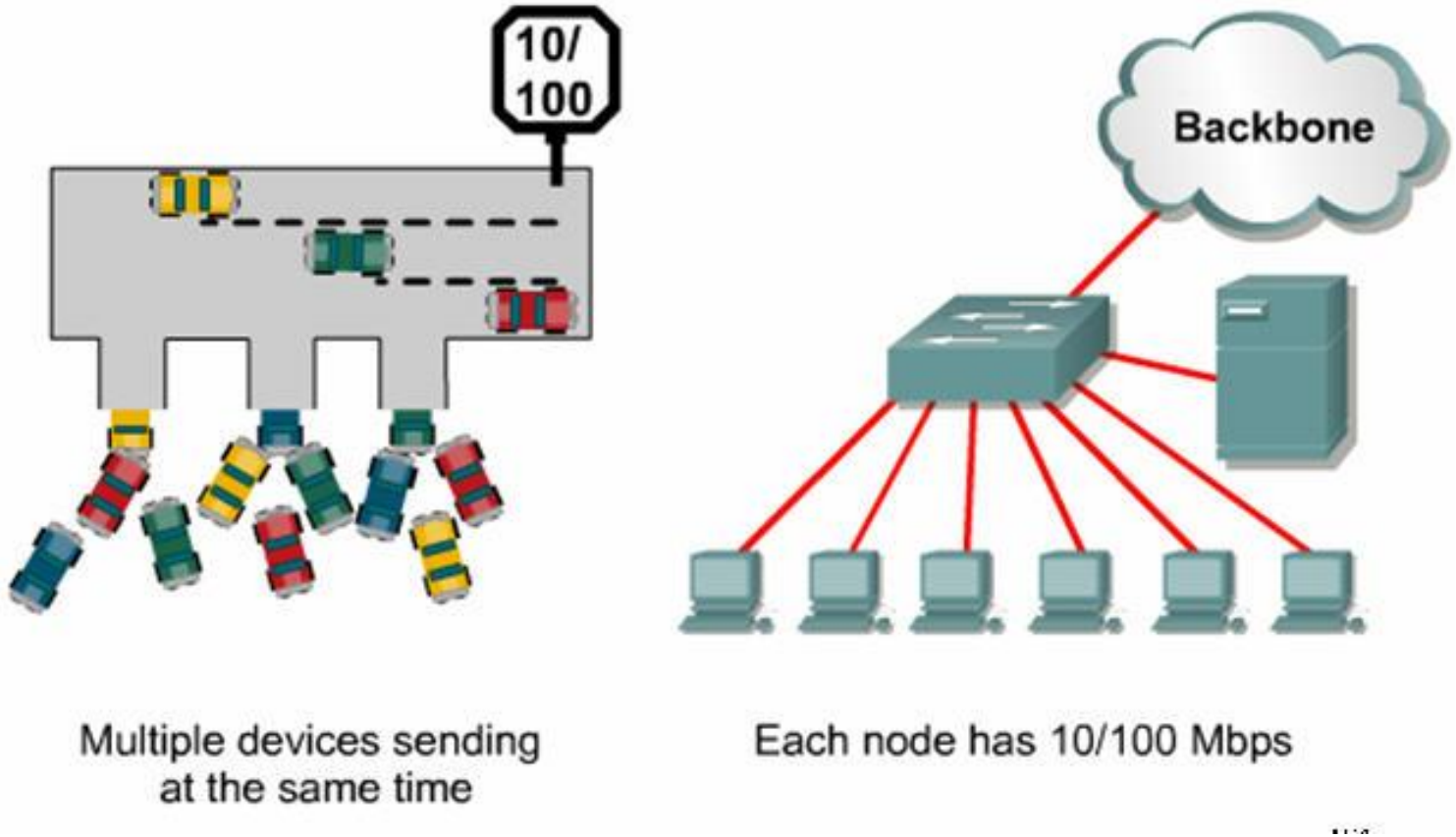
Switch (bộ chuyển mạch)

- Là thiết bị giống Bridge và Hub cộng lại nhưng thông minh hơn.
- Có khả năng chỉ chuyển dữ liệu đến đúng kết nối thực sự cần dữ liệu này làm giảm đụng độ trên mạng.
- Dùng để phân đoạn mạng trong các mạng cục bộ lớn (VLAN).
- Hoạt động ở lớp Data Link.

Switch (bộ chuyển mạch)



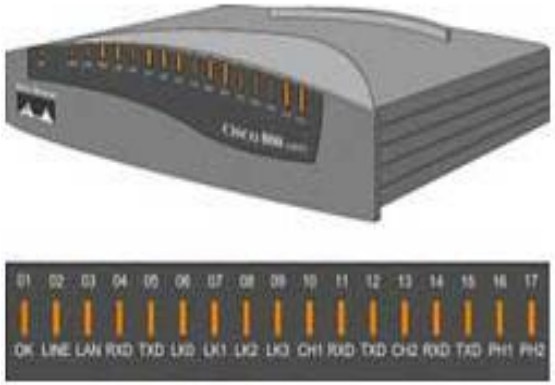
Switch (bộ chuyển mạch)

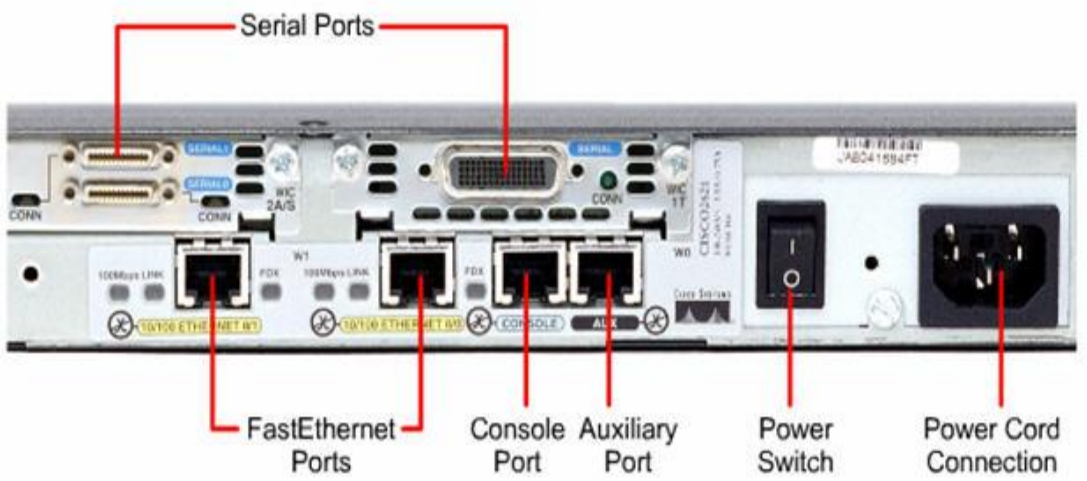


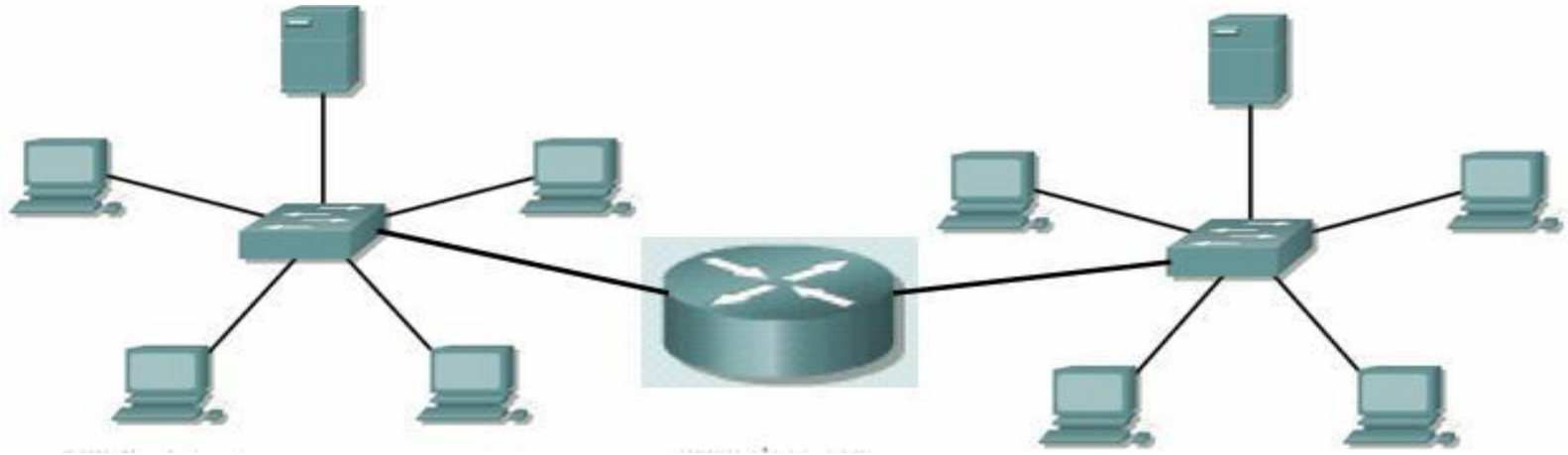
Router (Bộ định tuyến)

- Dùng để ghép nối các mạng cục bộ lại với nhau thành mạng rộng.
- Lựa chọn đường đi tốt nhất cho các gói tin hướng ra mạng bên ngoài.
- Hoạt động chủ yếu ở lớp Network.
- Có 2 phương thức định tuyến chính:
  - Định tuyến tĩnh: cấu hình các đường cố định và cài đặt các đường đi này vào bảng định tuyến.
  - Định tuyến động:
    - Vectơ khoảng cách: RIP, IGRP, EIGRP, BGP
    - Trạng thái đường liên kết: OSPF

Router (Bộ định tuyến)







Gateway (Proxy - cổng nối)

- Thường dùng để kết nối các mạng không thuần nhất, chủ yếu là mạng LAN với mạng lớn bên ngoài chứ không dùng kết nối LAN – LAN.
- Kiểm soát luồng dữ liệu ra vào mạng.
- Hoạt động phức tạp và chậm hơn Router.
- Hoạt động từ tầng thứ 4→7

CHƯƠNG 4: DATA LINK

- Điều khiển luồng (dòng)
- Phát hiện lỗi
- Xử lý lỗi

Điều khiển luồng

- Là kỹ thuật nhằm đảm bảo rằng bên phát không làm tràn dữ liệu bên nhận
- Hai phương pháp được sử dụng:
  - Phương pháp dừng và chờ (Stop and Wait)
    - Đơn giản nhất,
    - Kém hiệu quả, chỉ có một khung tin được truyền tại một thời điểm
  - Phương pháp cửa sổ trượt –(Sliding Window Flow Control)
    - Hiệu quả
    - Cho phép truyền nhiều khung tin cùng một lúc trên kênh truyền

Phương pháp dừng và chờ

- Truyền một gói tin và chờ báo nhận
  - Bên phát truyền một khung tin
  - Sau khi nhận được khung tin, bên nhận gửi lại xác nhận
  - Bên phát phải đợi đến khi nhận được xác nhận thì mới truyền khung tin tiếp theo
- Không hiệu quả
  - Bên nhận có thể dừng quá trình truyền bằng cách không gửi khung tin xác nhận
  - Tại một thời điểm chỉ có một khung tin trên đường truyền → chậm
  - Trường hợp độ rộng của kênh truyền lớn hơn độ rộng của khung tin thì nó tỏ ra cực kỳ kém hiệu quả.

Phương pháp cửa sổ trượt

- Cho phép nhiều khung tin được truyền tại một thời điểm ->Truyền thông hiệu quả hơn.
- A và B được kết nối trực tiếp song công (full-duplex).
- B có bộ đệm cho n khung tin -> B có thể chấp nhận n khung tin, A có thể truyền n khung tin mà không cần đợi xác nhận từ bên B
- Mỗi khung tin được gán nhãn bởi một số thứ tự.
- B xác nhận khung tin đã được nhận bằng cách gửi xác nhận cùng với số thứ tự của khung tin tiếp theo mà nó mong muốn nhận

Phương pháp cửa sổ trượt

- A duy trì danh sách các số thứ tự được phép gửi
- B duy trì danh sách số thứ tự chuẩn bị nhận

  - Gọi là cửa sổ của các khung tin

  - Điều khiển dòng cửa sổ trượt

Phương pháp cửa sổ trượt

- Đối với đường truyền 2 chiều thì mỗi bên phải sử dụng hai cửa sổ:
  - Một cho phát và một cho nhận
  - Mỗi bên đều phải gửi dữ liệu và gửi xác nhận tới bên kia
- Số thứ tự được lưu trữ trong khung tin
  - Bị giới hạn, trường k bit thì số thứ tự được đánh số theo Module của 2^k
  - Kích thước của cửa sổ không nhất thiết phải lấy là maximum (ví dụ trường 3 bit, có thể lấy độ dài cửa sổ là 4)

Phát hiện lỗi

- Lý do một hay nhiều bit thay đổi trong khung tin được truyền:
  - Tín hiệu trên đường truyền bị suy yếu
  - Tốc độ truyền
  - Mất đồng bộ
- Việc phát hiện ra lỗi để khắc phục, yêu cầu phát lại là cần thiết và vô cùng quan trọng trong truyền dữ liệu.

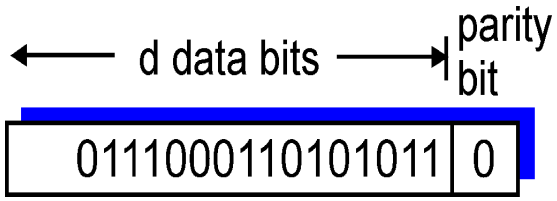
Phát hiện lỗi: Parity Check

- Là kỹ thuật đơn giản nhất.
- Đưa một bit kiểm tra tính chẵn lẻ vào sau khối tin.
- Giá trị của bit này được xác định dựa trên số các số 1 là chẵn (even parity), hoặc số các số 1 là lẻ (odd parity).
- Lỗi sẽ không bị phát hiện nếu trong khung tin có 2 hoặc một số chẵn các bit bị đảo.
- Không hiệu quả khi xung nhiễu đủ mạnh.

Kiểm tra Parity

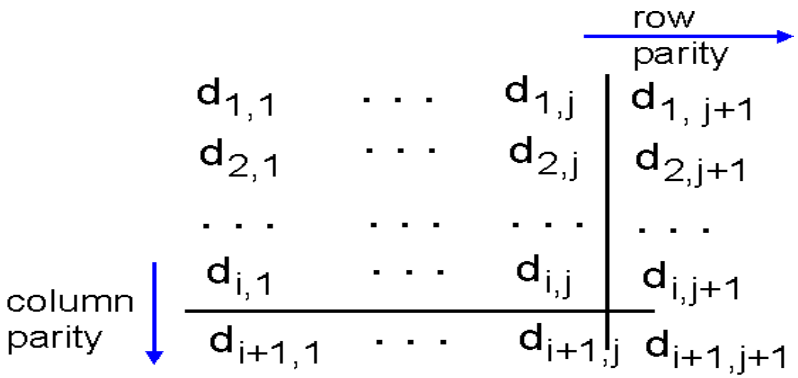
Bit Parity đơn:

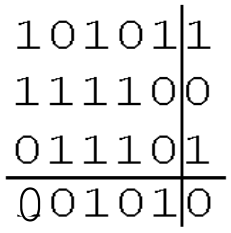
phát hiện các lỗi bit



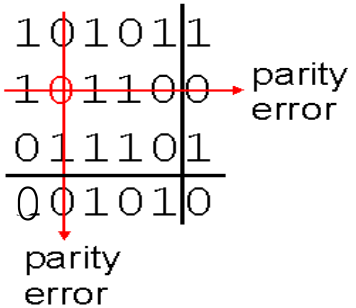
Bit Parity 2 chiều:

phát hiện & sửa các lỗi bit





no errors



correctable single bit error

Phát hiện lỗi: Cyclic redundancy Check (CRC)

Mô tả:

- Khối dữ liệu k bit
- Mẫu n+1 bit (n<k)
- Tạo ra dãy n bit gọi là dãy kiểm tra khung tin-FCS, Frame Check Sequence
- Tao ra một khung tin k+n bit
- Bên nhận khi nhận được khung tin sẽ chia cho mẫu, nếu kết quả là chia hết, việc truyền khung tin này là không có lỗi

Phát hiện lỗi:
CRC dưới dạng module của 2

M: Khối tin k bit

F: FCS n bit, n bit cuối của T

T: khung tin k+n bit

P: Mẫu n+1 bit, đây là một số chia được chọn trước.

Mục tiêu: xác định F để T chia hết cho P

$$T = 2^n M + F$$

Phát hiện lỗi: Các bước tạo và kiểm tra CRC

- Các bước tạo CRC
  - Dịch trái M đi n bit
  - Chia kết quả cho P
  - Số dư tìm được là F
- Các bước kiểm tra CRC
  - Lấy khung nhận được (n+k) bit
  - Chia cho P
  - Kiểm tra số dư, nếu số dư khác 0, khung bị lỗi, ngược lại là không lỗi

Phát hiện lỗi:
CRC- Dạng đa thức nhị phân

Cách thứ 2 để biểu thị CRC là biểu diễn các giá trị như là một đa thức với các hệ số là số nhị phân, đây là các bit của số nhị phân. Gọi T(X), M(X), Q(X), P(X), R(X) là các đa thức tương ứng với các số nhị phân T, M, Q, P, R đã trình bày ở trên, khi đó CRC được biểu thị:

$$\frac{X^n M(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)}$$

$$T(X) = X^n M(X) + R(X)$$

CRC- Dạng đa thức nhị phân

Một số đa thức P(X) tiêu biểu:

CRC-12: $X^{12}+X^{11}+X^3+X^2+X+1$ CRC-16: $X^{16}+X^{15}+X^2+1$

CRC-CCITT: $X^{16}+X^{12}+X^5+1$

CRC32: $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$

Ví dụ:

Tạo CRC:

1. Cho tin M=1010001101 (10 bit)

Mẫu P:110101 (6 bit)

FCS R: được tính theo phương pháp CRC v
sẽ có độ dài là 5 bit

2. Nhân M với 2^5 ta được:

$M2^5$=101000110100000

3. Chia kết quả cho P:

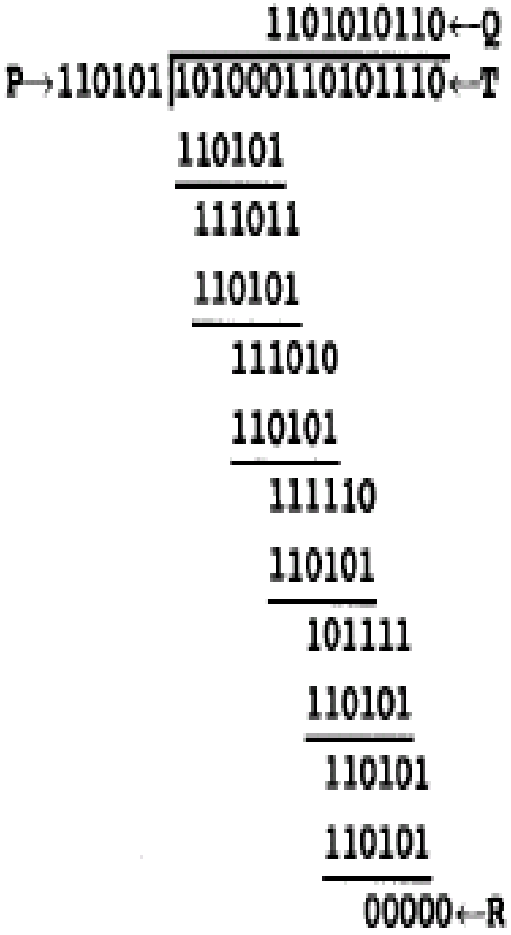
4. Số dư là: 01110, được đưa vào sau tin M

Ta có tin T, được truyền đi là:
101000110101110

```
                1101010110←Q
P→110101 |101000110100000←2^5M
         110101
          111011
          110101
            111010
            110101
              111110
              110101
                101100
                110101
                 110010
                 110101
                   01110←R
```

CRC- Dạng đa thức nhị phân

- Kiểm tra CRC:
- Giả sử bên thu nhận được T, khi đó để kiểm tra là phép truyền có lỗi không ta chia T cho P, số dư là 00000, vậy ta kết luận phép truyền tin M, không có lỗi.



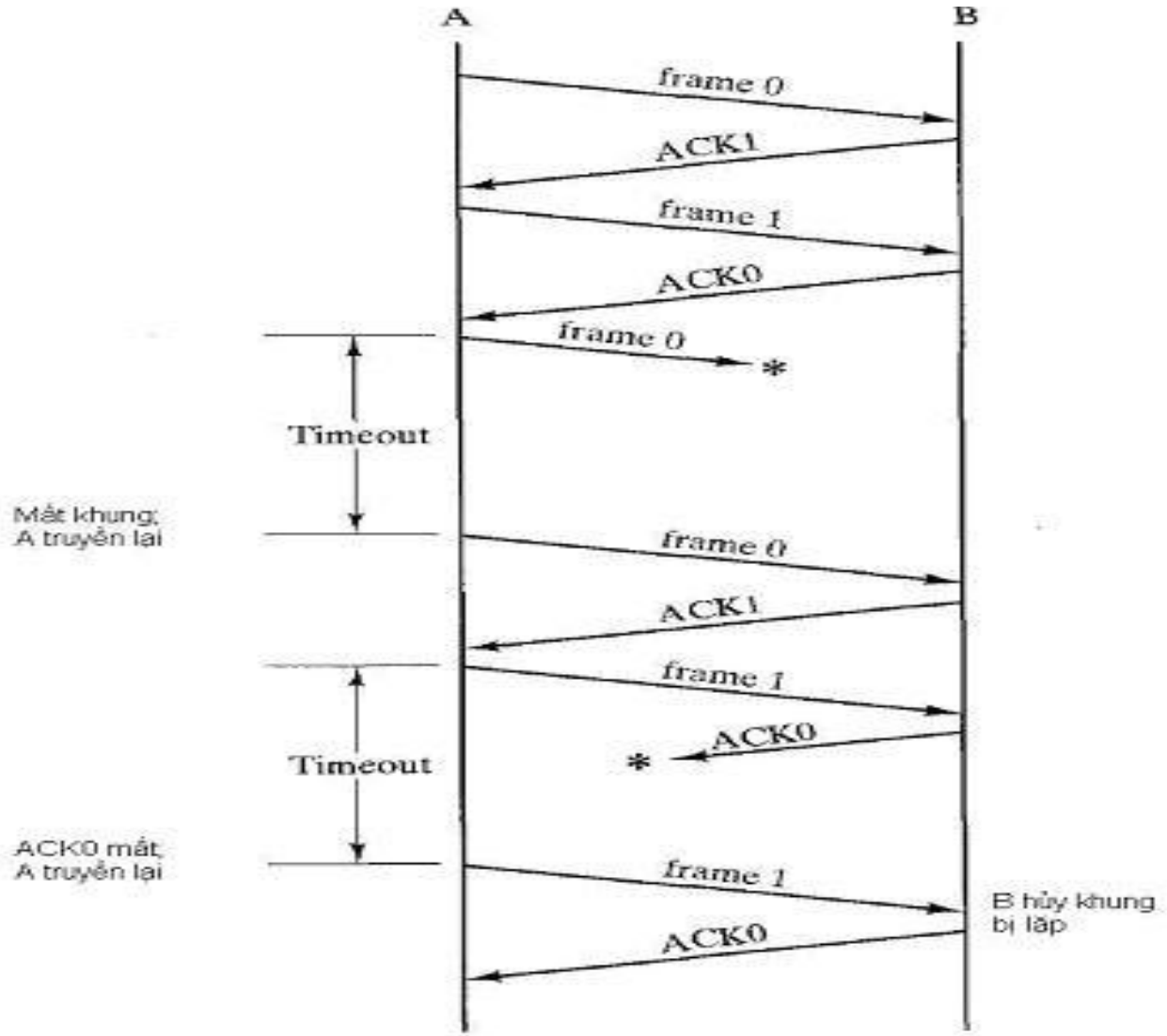
Xử lý lỗi

- Lỗi: Mất khung, hỏng khung
- Kiểm soát lỗi:
  - Phát hiện lỗi
  - Báo nhận: khung tin tốt
  - Truyền lại khi hết thời gian định trước
  - Báo nhận: khung tin lỗi và truyền lại

Xử lý lỗi: ARQ dừng và chờ

- Trên cơ sở kĩ thuật điều khiển luồng dừng-và-chờ
- Kiểm soát lỗi:
  - Khung tin tới bên nhận bị hỏng: Truyền lại, sử dụng đồng hồ đếm giờ time-out
  - Báo nhận bị hỏng: Time-out, bên phát gửi lại, sử dụng label 0/1 và ACK0/ACK1 phát hiện lỗi

Xử lý lỗi: ARQ dừng và chờ



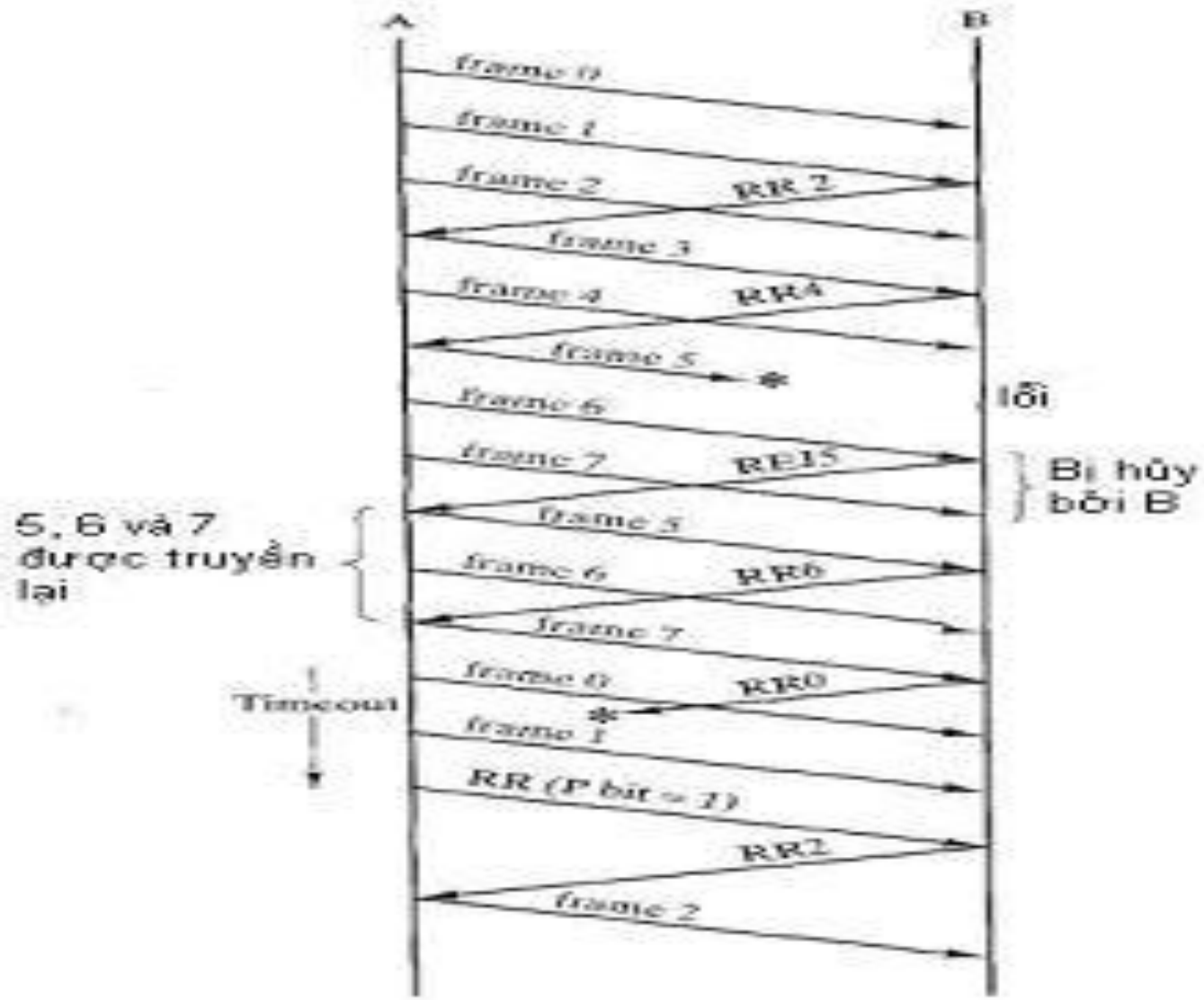
Xử lý lỗi: ARQ Quay-lui-N

- Trên cơ sở kĩ thuật điều khiển luồng bằng Cửa sổ trượt
- Kiểm soát lỗi:
  - Khung hỏng:
    - Khung i-1 thành công, i lỗi, bên nhận gửi SREJ i, bên phát gửi lại
    - Khung i mất, i+1 được nhận không đúng trình tự, REJ i, bên gửi phát lại i và các khung sau đó
    - Chỉ khung i được truyền và bị mất, bên nhận không biết i đã được truyền đi, bên phát gửi time-out và gửi RR với P=1, khi bên phát nhận được RR từ bên nhận nó sẽ phát lại i

Xử lý lỗi: ARQ Quay-lui-N

- RR hỏng:
  - B nhận khung i và gửi RR(i+1), RR(i+1) mất, A có thể nhận RR(>i+1) trước khi RR(i+1) time-out, và có nghĩa là khung i đã thành công.
  - RR(i+1) time-out, A cố gắng gửi RR với P-bit cho đến khi nhận được RR từ B một số lần nhất định, nếu vẫn không nhận được thì Khởi động lại giao thức
- Reject hỏng:
  - A time-out, A gửi RR với P=1 cho đến khi nhận được RRi từ B thì A sẽ gửi lại khung i

Xử lí lỗi: ARQ Quay-lui-N



Xử lý lỗi: ARQ Chọn-Hủy (Selective-Reject)

- Chỉ truyền lại những khung có báo nhận là lỗi (SREJ)
- Phải duy trì đủ bộ đệm độ lớn
- Đảm bảo tính logic phức tạp để gửi và nhận các khung theo đúng trình tự.
- ARQ Chọn-Hủy phải giải quyết được sự chồng chéo giữa cửa sổ gửi và nhận.

Xử lý lỗi: ARQ Chọn-Hủy (Selective-Reject)

- Trạm A gửi các khung từ 0 đến 6 tới trạm B.
- Trạm B nhận tất cả 7 khung và báo nhận tích lũy với RR 7
- Vì lí do nào đó ví dụ như nhiễu làm RR 7 bị mất trên đường truyền.
- Đồng hồ ở A hết hạn và A truyền lại khung 0.
- B đã điều chỉnh trước cửa sổ nhận để có thể nhận các khung 7, 0, 1, 2, 3, 4 và 5. Do đó mà khung 7 được coi là bị mất và khung nhận được này là khung số 0 mới, và được chấp nhận bởi B.

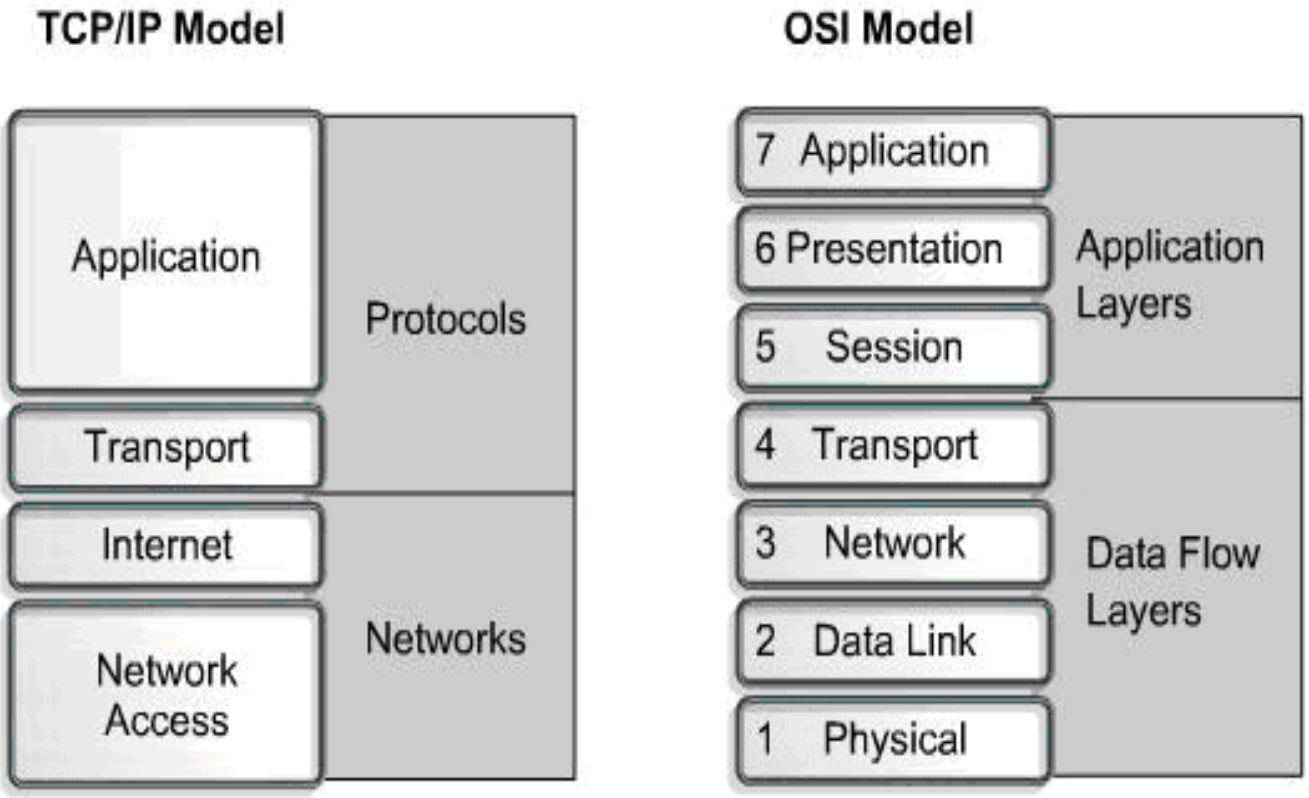
CHƯƠNG 5: TCP/IP

- Khái niệm về TCP và IP
- Mô hình tham chiếu TCP/IP
- So sánh OSI và TCP/IP
- Các giao thức trong mô hình TCP/IP
- Chuyển đổi giữa các hệ thống số
- Địa chỉ IP và các lớp địa chỉ
- NAT
- Mạng con và kỹ thuật chia mạng con
- Bài tập

Khái niệm về TCP và IP

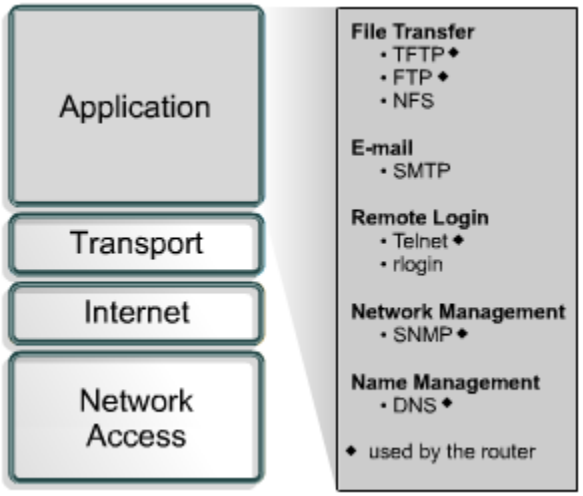
- TCP (Transmission Control Protocol) là giao thức thuộc tầng vận chuyển và là một giao thức có kết nối (connected-oriented).
- IP (Internet Protocol) là giao thức thuộc tầng mạng của mô hình OSI và là một giao thức không kết nối (connectionless).

Mô hình tham chiếu TCP/IP



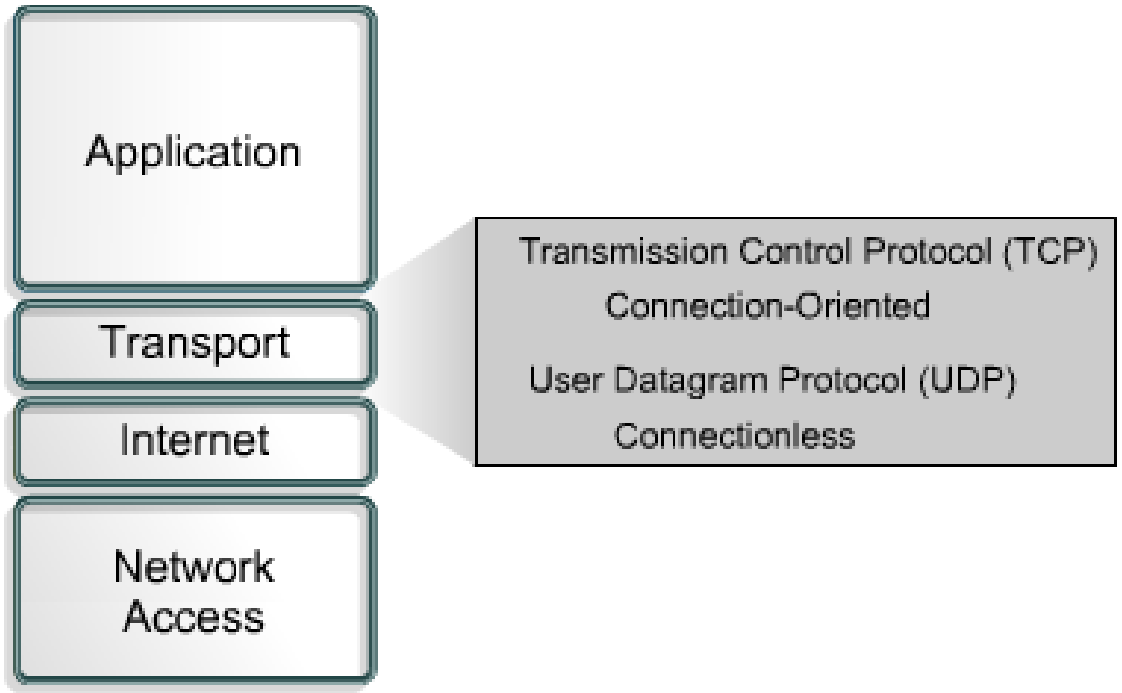
Lớp ứng dụng

Kiểm soát các giao thức lớp cao, các chủ đề về trình bày, biểu diễn thông tin, mã hóa và điều khiển hội thoại. Đặc tả cho các ứng dụng phổ biến.



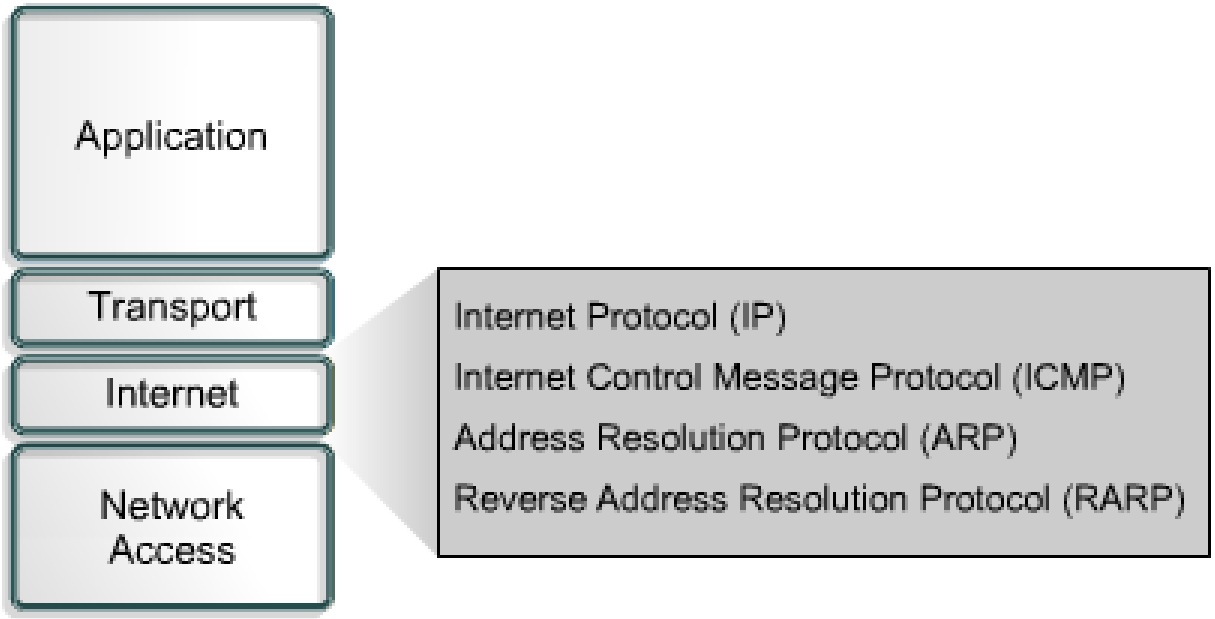
Lớp vận chuyển

Cung ứng dịch vụ vận chuyển từ host nguồn đến host đích. Thiết lập một cầu nối luận lý giữa các đầu cuối của mạng, giữa host truyền và host nhận.



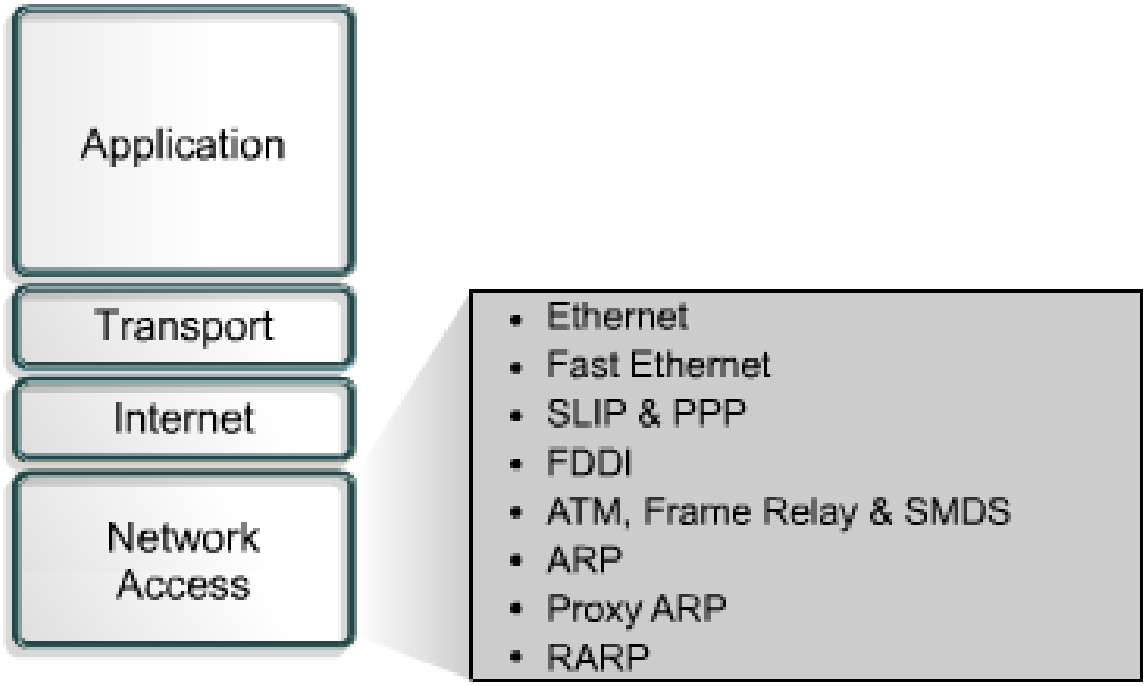
Lớp Internet

Mục đích của lớp Internet là chọn đường đi tốt nhất xuyên qua mạng cho các gói dữ liệu di chuyển tới đích. Giao thức chính của lớp này là Internet Protocol (IP).



Lớp truy nhập mạng

Định ra các thủ tục để giao tiếp với phần cứng mạng và truy nhập môi trường truyền. Có nhiều giao thức hoạt động tại lớp này

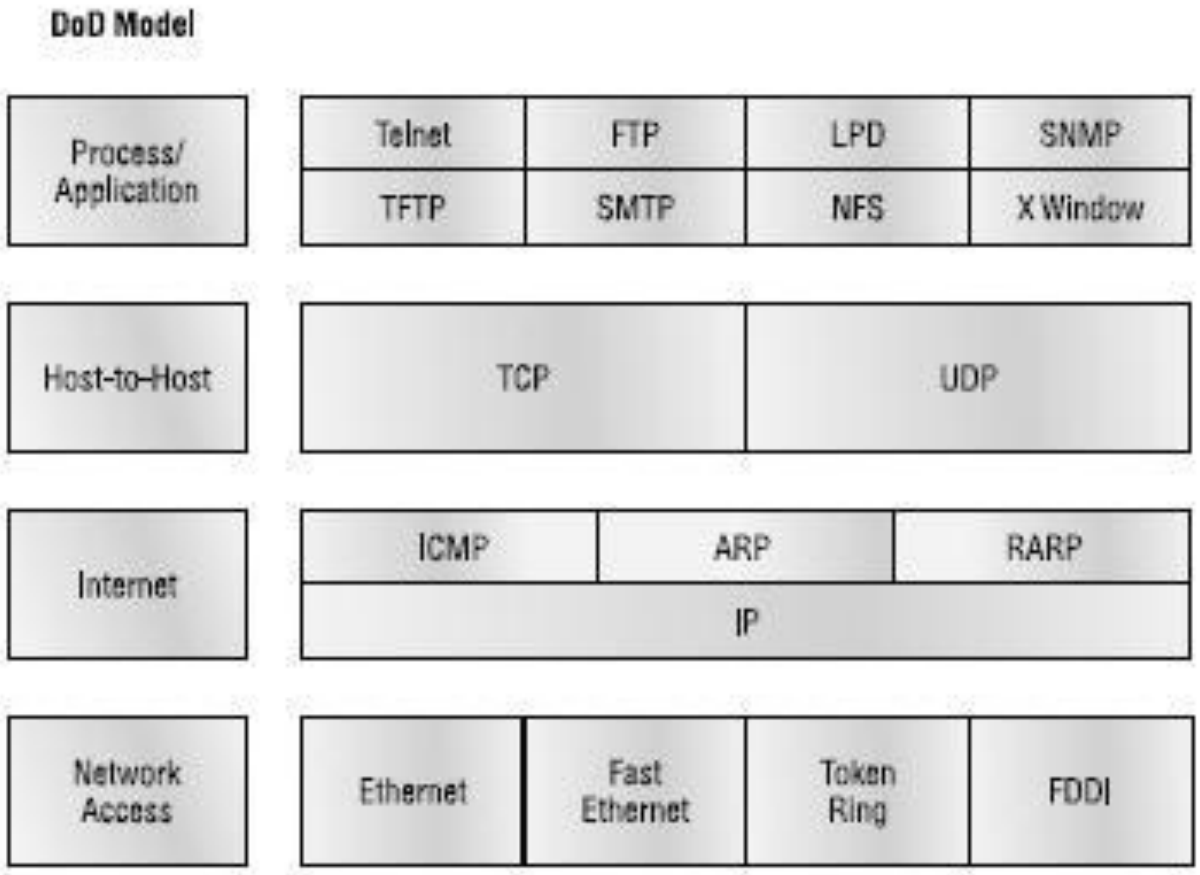


So sánh mô hình OSI và TCP/IP

- Giống nhau
  - Đều phân lớp chức năng
  - Đều có lớp vận chuyển và lớp mạng.
  - Chuyển gói là hiển nhiên.
  - Đều có mối quan hệ trên dưới, ngang hàng.

- Khác nhau
  - TCP/IP gộp lớp trình bày và lớp phiên vào lớp ứng dụng.
  - TCP/IP gộp lớp vật lý và lớp liên kết dữ liệu vào lớp truy nhập mạng.
  - TCP/IP đơn giản vì có ít lớp hơn.
  - OSI không có khái niệm chuyển phát thiếu tin cậy ở lớp 4 như UDP của TCP/IP

Các giao thức trong mô hình TCP/IP



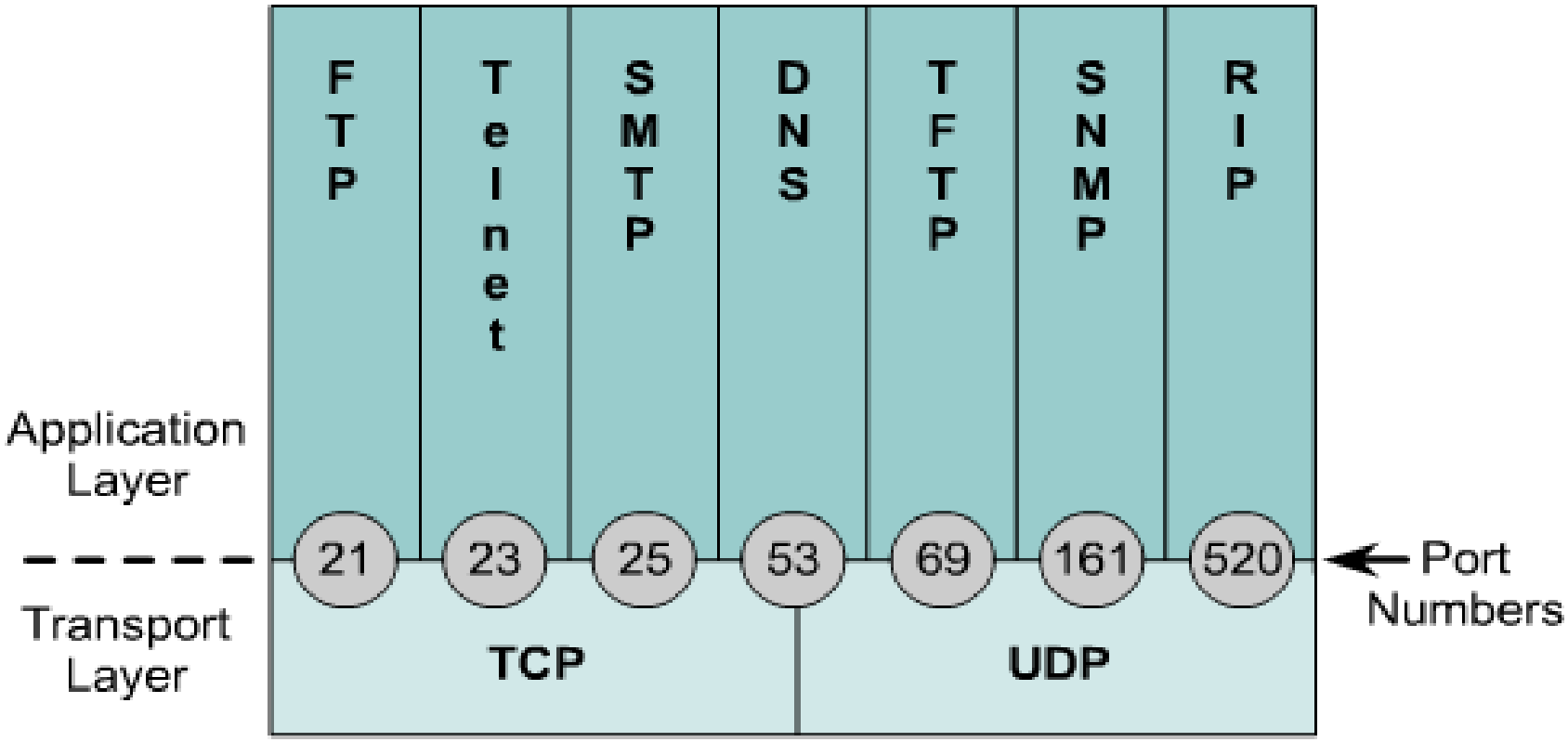
Lớp ứng dụng

- FTP (File Transfer Protocol): là dịch vụ có tạo cầu nối, sử dụng TCP để truyền các tập tin giữa các hệ thống.
- TFTP (Trivial File Transfer Protocol): là dịch vụ không tạo cầu nối, sử dụng UDP. Được dùng trên router để truyền các file cấu hình và hệ điều hành.
- NFS (Network File System): cho phép truy xuất file đến các thiết bị lưu trữ ở xa như một đĩa cứng qua mạng.
- SMTP (Simple Mail Transfer Protocol): quản lý hoạt động truyền e-mail qua mạng máy tính.

Lớp ứng dụng

- Telnet (Terminal emulation): cung cấp khả năng truy nhập từ xa vào máy tính khác. Telnet client là host cục bộ, telnet server là host ở xa.
- SNMP (Simple Network Management): cung cấp một phương pháp để giám sát và điều khiển các thiết bị mạng.
- DNS (Domain Name System): thông dịch tên của các miền (Domain) và các node mạng được công khai sang các địa chỉ IP.

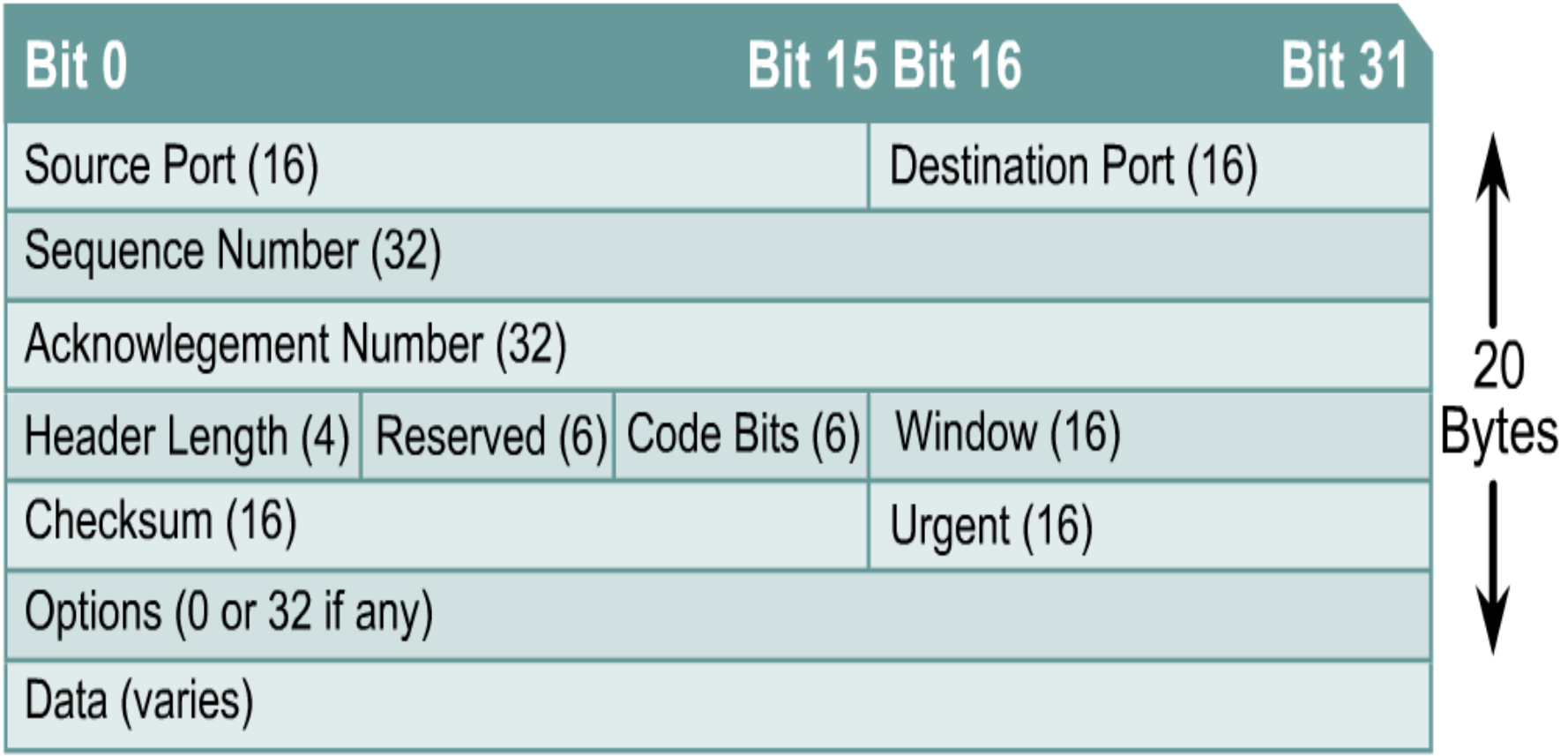
Các cổng phổ biến dùng cho các giao thức lớp ứng dụng



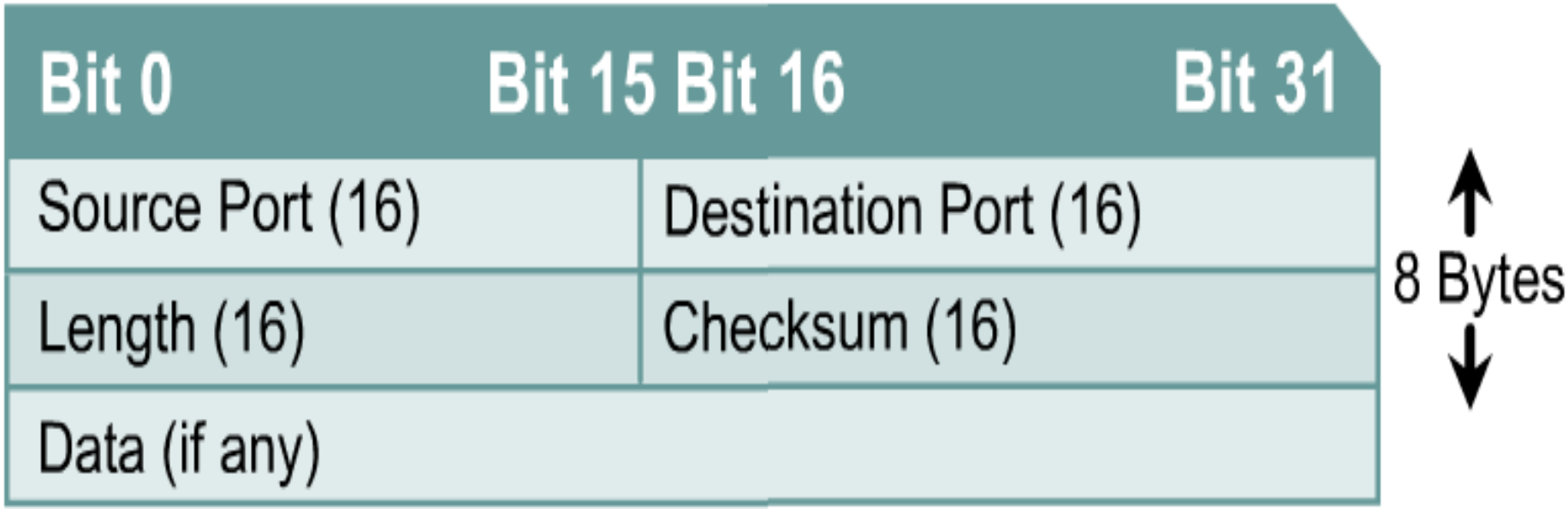
Lớp vận chuyển

- TCP và UDP (User Datagram Protocol):
  - Phân đoạn dữ liệu ứng dụng lớp trên.
  - Truyền các segment từ một thiết bị đầu cuối này đến thiết bị đầu cuối khác
- Riêng TCP còn có thêm các chức năng:
  - Thiết lập các hoạt động end-to-end.
  - Cửa sổ trượt cung cấp điều khiển luồng.
  - Chỉ số tuần tự và báo nhận cung cấp độ tin cậy cho hoạt động.

Khuôn dạng gói tin TCP



Khuôn dạng gói tin UDP



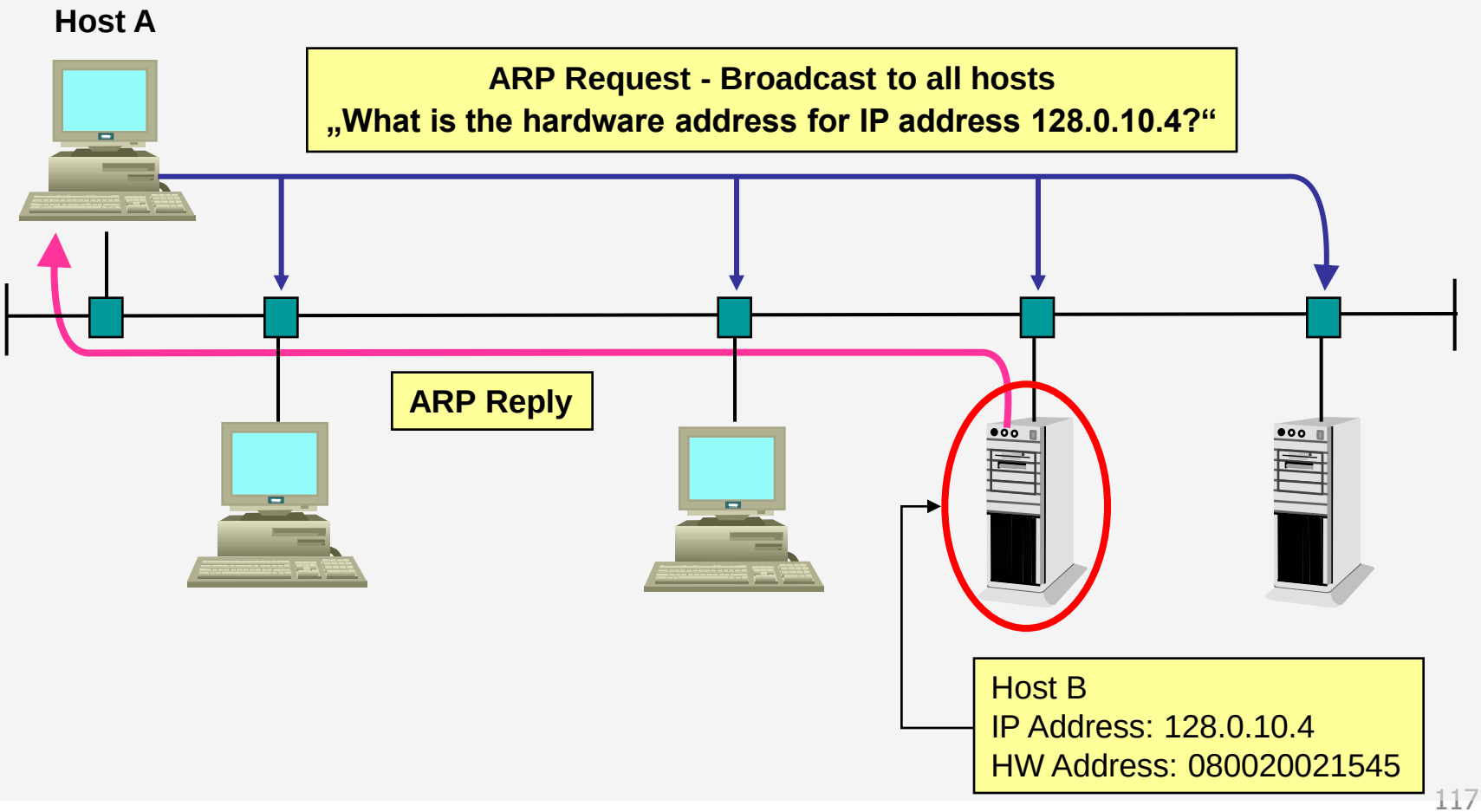
Lớp Internet

- IP: không quan tâm đến nội dung của các gói nhưng tìm kiếm đường dẫn cho gói tới đích.
- ICMP (Internet Control Message Protocol): đem đến khả năng điều khiển và chuyển thông điệp.
- ARP (Address Resolution Protocol): xác định địa chỉ lớp liên kết số liệu (MAC address) khi đã biết trước địa chỉ IP.
- RARP (Reverse Address Resolution Protocol): xác định các địa chỉ IP khi biết trước địa chỉ MAC.

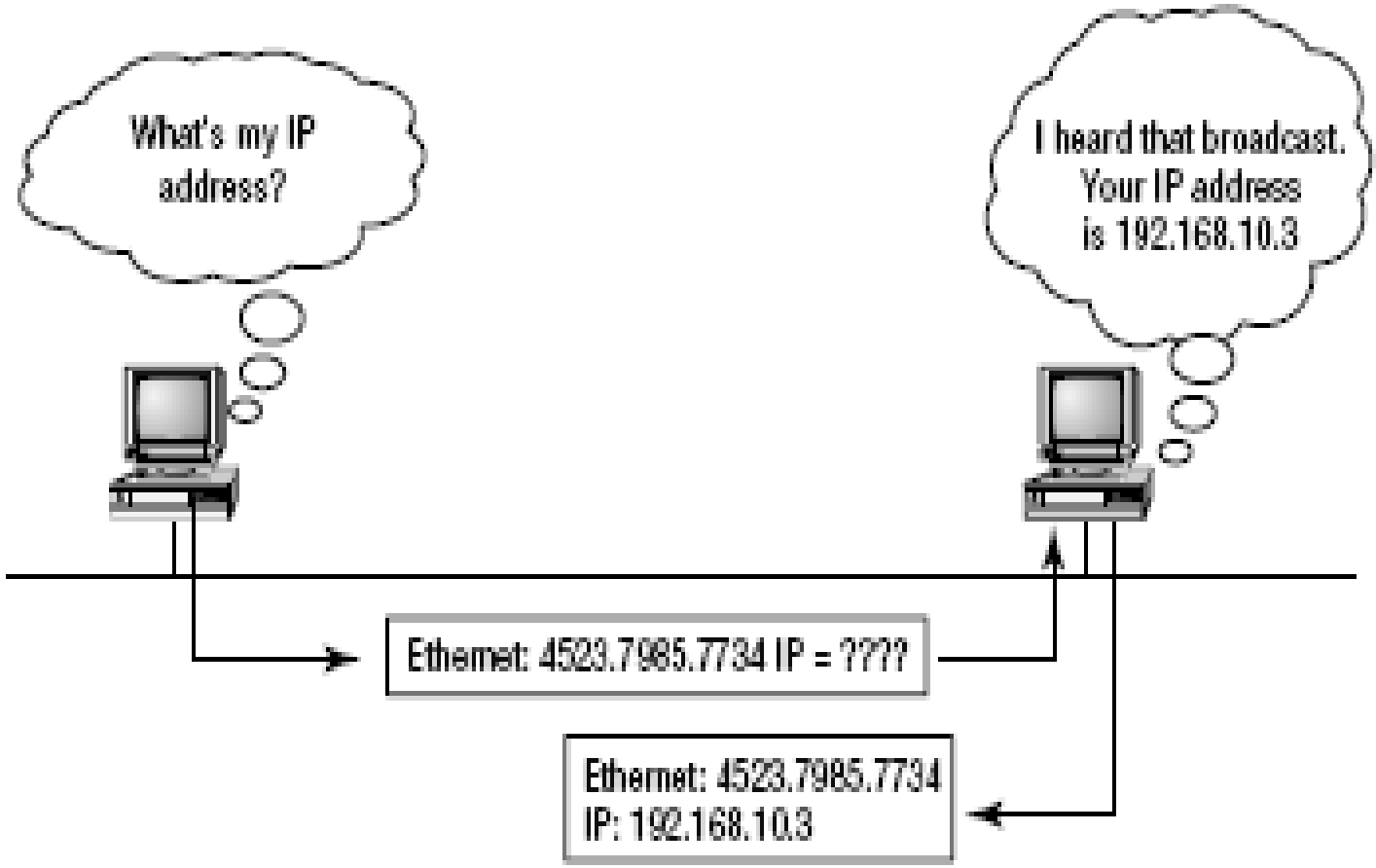
Khuôn dạng gói tin IP

<table>
<tr><td>VER</td><td>IHL</td><td>Type of services</td><td colspan="2">Total lenght</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment offset</td></tr>
<tr><td colspan="2">Time to live</td><td>Protocol</td><td colspan="2">Header checksum</td></tr>
<tr><td colspan="5">Source address</td></tr>
<tr><td colspan="5">Destination address</td></tr>
<tr><td colspan="5">Options + Padding</td></tr>
<tr><td colspan="5">Data</td></tr>
</table>

ARP


Host A
ARP Request - Broadcast to all hosts
„What is the hardware address for IP address 128.0.10.4?“
ARP Reply
Host B
IP Address: 128.0.10.4
HW Address: 080020021545

RARP


What's my IP address?
I heard that broadcast. Your IP address is 192.168.10.3
Ethernet: 4523.7985.7734 IP = ????
Ethernet: 4523.7985.7734
IP: 192.168.10.3

Lớp truy nhập mạng

- Ethernet
  - Là giao thức truy cập LAN phổ biến nhất.
  - Được hình thành bởi định nghĩa chuẩn 802.3 của IEEE (Institute of Electrical and Electronics Engineers).
  - Tốc độ truyền 10Mbps
- Fast Ethernet
- Gigabit Ethernet

Chuyển đổi giữa các hệ thống số

- Hệ 2 (nhị phân): gồm 2 ký số 0, 1
- Hệ 8 (bát phân): gồm 8 ký số 0, 1, …, 7
- Hệ 10 (thập phân): gồm 10 ký số 0, 1, …, 9
- Hệ 16 (thập lục phân): gồm các ký số 0, 1, …, 9 và các chữ cái A, B, C, D, E, F

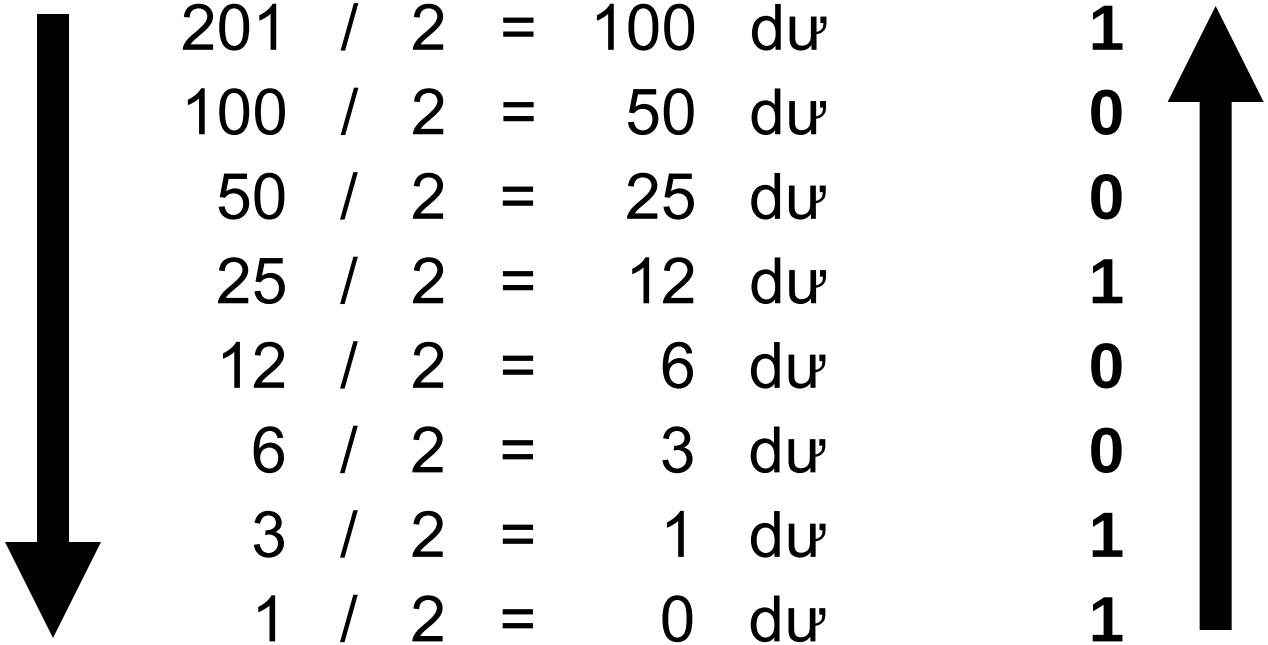
Chuyển đổi giữa hệ nhị phân sang hệ thập phân

$$10110_2 = (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) = 16 + 0 + 4 + 2 + 0 = 22$$

Place Value	128 64 32 16 8 4 2 1
Base^{Exponent}	$2^7 = 128$ $2^3 = 8$ $2^6 = 64$ $2^2 = 4$ $2^5 = 32$ $2^1 = 2$ $2^4 = 16$ $2^0 = 1$
Number of Symbols	2
Symbols	0, 1
Rationale	Two-state (discrete binary) voltage systems made from transistors can be diverse, powerful, inexpensive, tiny and relatively immune to noise.

Chuyển đổi giữa hệ thập phân sang hệ nhị phân

Đổi số 201_{10} sang nhị phân:



201	/	2	=	100	dư	**1**
100	/	2	=	50	dư	**0**
50	/	2	=	25	dư	**0**
25	/	2	=	12	dư	**1**
12	/	2	=	6	dư	**0**
6	/	2	=	3	dư	**0**
3	/	2	=	1	dư	**1**
1	/	2	=	0	dư	**1**

Khi thương số bằng 0, ghi các số dư theo thứ tự ngược với lúc xuất hiện, kết quả: **$201_{10} = 11001001_2$**

Chuyển đổi giữa hệ nhị phân sang hệ bát phân và thập lục phân

- Nhị phân sang bát phân:
  - Gom nhóm số nhị phân thành từng nhóm 3 chữ số tính từ phải sang trái. Mỗi nhóm tương ứng với một chữ số ở hệ bát phân.
  - Ví dụ: $1'101'100_{(2)} = 154_{(8)}$
- Nhị phân sang thập lục phân:
  - Tương tự như nhị phân sang bát phân nhưng mỗi nhóm có 4 chữ số.
  - Ví dụ: $110'1100_{(2)} = 6C_{(16)}$

Các phép toán làm việc trên bit

A	B	A and B
1	1	1
1	0	0
0	1	0
0	0	0

Địa chỉ IP và các lớp địa chỉ

- Địa chỉ IP là địa chỉ có cấu trúc với một con số có kích thước 32 bit, chia thành 4 phần mỗi phần 8 bit gọi là octet hoặc byte.
- Ví dụ:
  - 172.16.30.56
  - 10101100 00010000 00011110 00111000.
  - AC 10 1E 38

Địa chỉ IP và các lớp địa chỉ

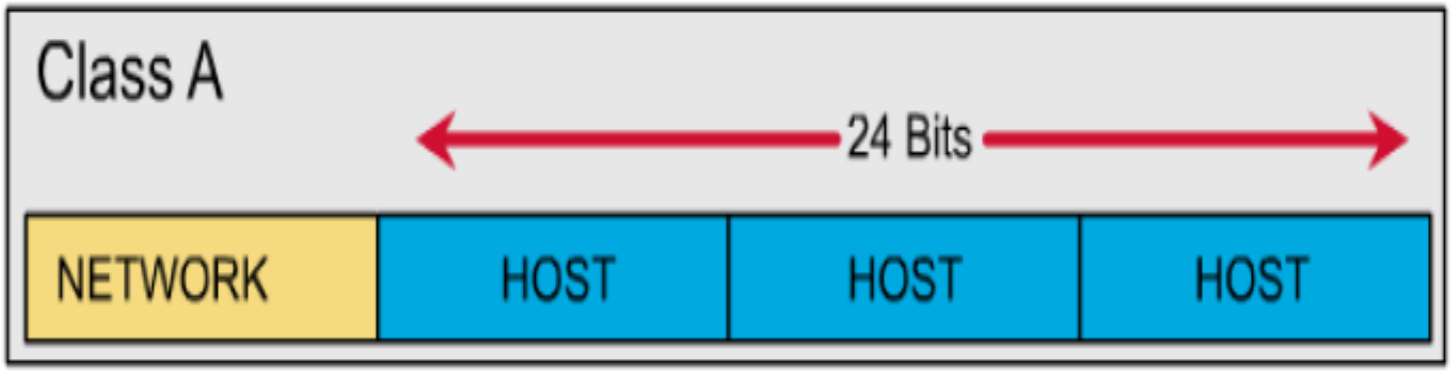
- Địa chỉ host là địa chỉ IP có thể dùng để đặt cho các interface của các host. Hai host nằm cùng một mạng sẽ có network_id giống nhau và host_id khác nhau.
- Khi cấp phát các địa chỉ host thì lưu ý không được cho tất cả các bit trong phần host_id bằng 0 hoặc tất cả bằng 1.
- Địa chỉ mạng (network address): là địa chỉ IP dùng để đặt cho các mạng. Phần host_id của địa chỉ chỉ chứa các bit 0. Ví dụ: 172.29.0.0
- Địa chỉ Broadcast: là địa chỉ IP được dùng để đại diện cho tất cả các host trong mạng. Phần host_id chỉ chứa các bit 1. Ví dụ: 172.29.255.255.

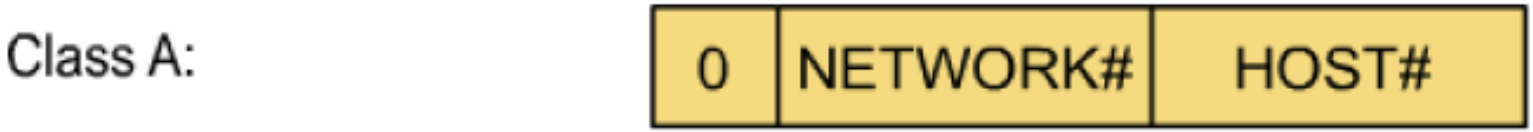
Các lớp địa chỉ IP

Không gian địa chỉ IP được chia thành 5 lớp (class) A, B, C, D và E. Các lớp A, B và C được triển khai để đặt cho các host trên mạng Internet, lớp D dùng cho các nhóm multicast, còn lớp E phục vụ cho mục đích nghiên cứu.

Lớp A (Class A)

Dành 1 byte cho phần network_id và 3 byte cho phần host_id.





Lớp A (Class A)

- Bit đầu tiên của byte đầu tiên phải là bit 0. Dạng nhị phân của octet này là 0xxxxxxx
- Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 0 (=$00000000_{(2)}$) đến 127 (=$01111111_{(2)}$) sẽ thuộc lớp A.
- Ví dụ: 50.14.32.8.

Lớp A (Class A)

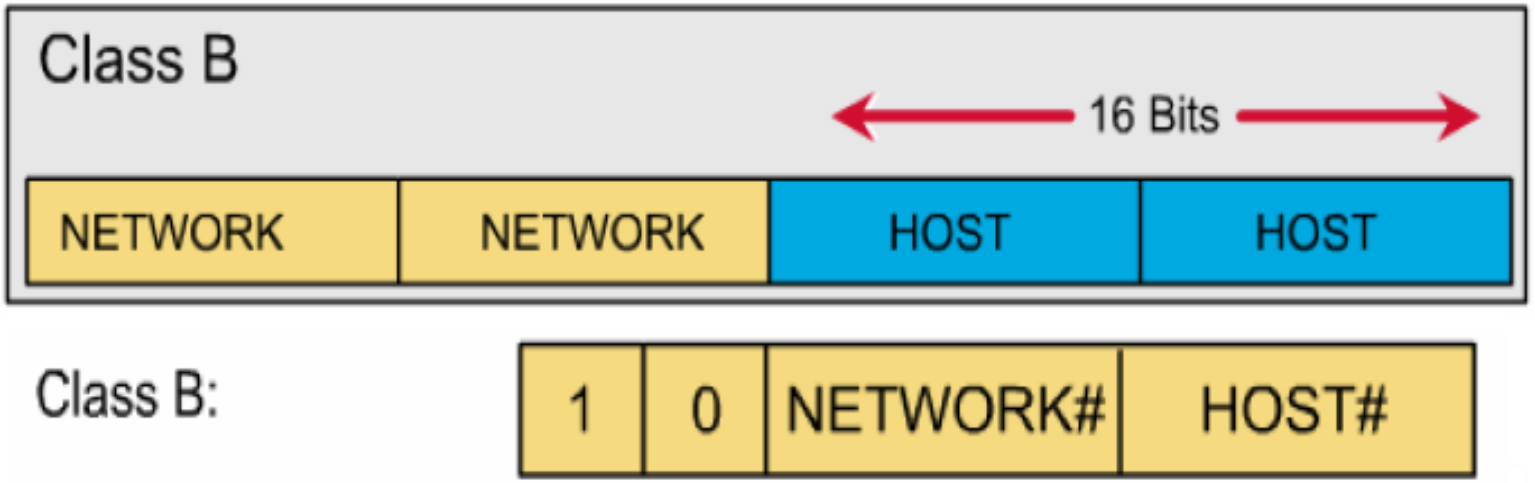
- Byte đầu tiên này cũng chính là network_id, trừ đi bit đầu tiên làm ID nhận dạng lớp A, còn lại 7 bit để đánh thứ tự các mạng, ta được 128 ($=2^7$) mạng lớp A khác nhau. Bỏ đi hai trường hợp đặc biệt là 0 và 127. Kết quả là lớp A chỉ còn 126 địa chỉ mạng, 1.0.0.0 đến 126.0.0.0.

Lớp A (Class A)

- Phần host_id chiếm 24 bit, nghĩa là có 2^{24} = 16777216 host khác nhau trong mỗi mạng. Bỏ đi hai trường hợp đặc biệt (phần host_id chứa toàn các bit 0 và bit 1). Còn lại: 16777214 host.
- Ví dụ đối với mạng 10.0.0.0 thì những giá trị host hợp lệ là 10.0.0.1 đến 10.255.255.254.

Lớp B (Class B)

Dành 2 byte cho phần network_id và 2 byte cho phần host_id.



Lớp B (Class B)

- Hai bit đầu tiên của byte đầu tiên phải là 10. Dạng nhị phân của octet này là 10xxxxxx
- Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 128 (=$10000000_{(2)}$) đến 191 (=$10111111_{(2)}$) sẽ thuộc về lớp B
- Ví dụ: 172.29.10.1 .

Lớp B (Class B)

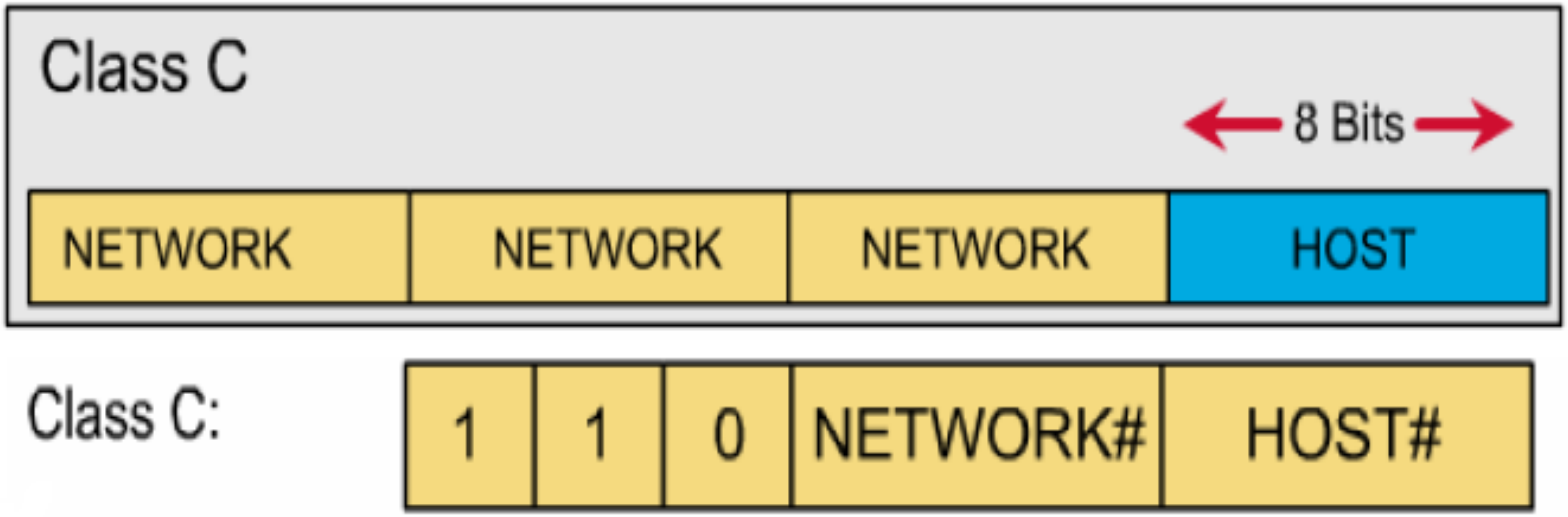
- Phần network_id chiếm 16 bit bỏ đi 2 bit làm ID cho lớp, còn lại 14 bit cho phép ta đánh thứ tự 16384 (=2^{14}) mạng khác nhau (128.0.0.0 đến 191.255.0.0).

Lớp B (Class B)

- Phần host_id dài 16 bit hay có 65536 ($=2^{16}$) giá trị khác nhau. Trừ đi 2 trường hợp đặc biệt còn lại 65534 host trong một mạng lớp B.
- Ví dụ đối với mạng 172.29.0.0 thì các địa chỉ host hợp lệ là từ 172.29.0.1 đến 172.29.255.254.

Lớp C (Class C)

Dành 3 byte cho phần network_id và 1 byte cho phần host_id.



Lớp C (Class C)

- Ba bit đầu tiên của byte đầu tiên phải là 110. Dạng nhị phân của octet này là 110xxxxx
- Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 192 (=$11000000_{(2)}$) đến 223 (=$11011111_{(2)}$) sẽ thuộc về lớp C.
- Ví dụ: 203.162.41.235

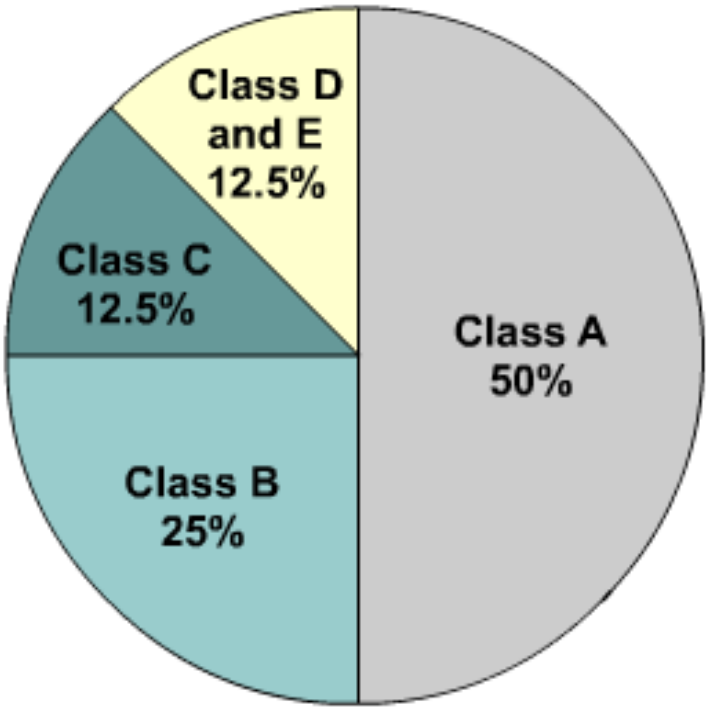
Các lớp địa chỉ IP

Address Class	Number of Networks	Number of Host per Network
A	126 *	16,777,216
B	16, 384	65,535
C	2,097,152	254
D (Multicast)	N/A	N/A

IP Address Class	High Order Bits	First Octet Address Range	Number of Bits in the Network Address
Class A	0	0 - 127 *	8
Class B	10	128 - 191	16
Class C	110	192 - 223	24
Class D	1110	224 - 239	28

Các lớp địa chỉ IP

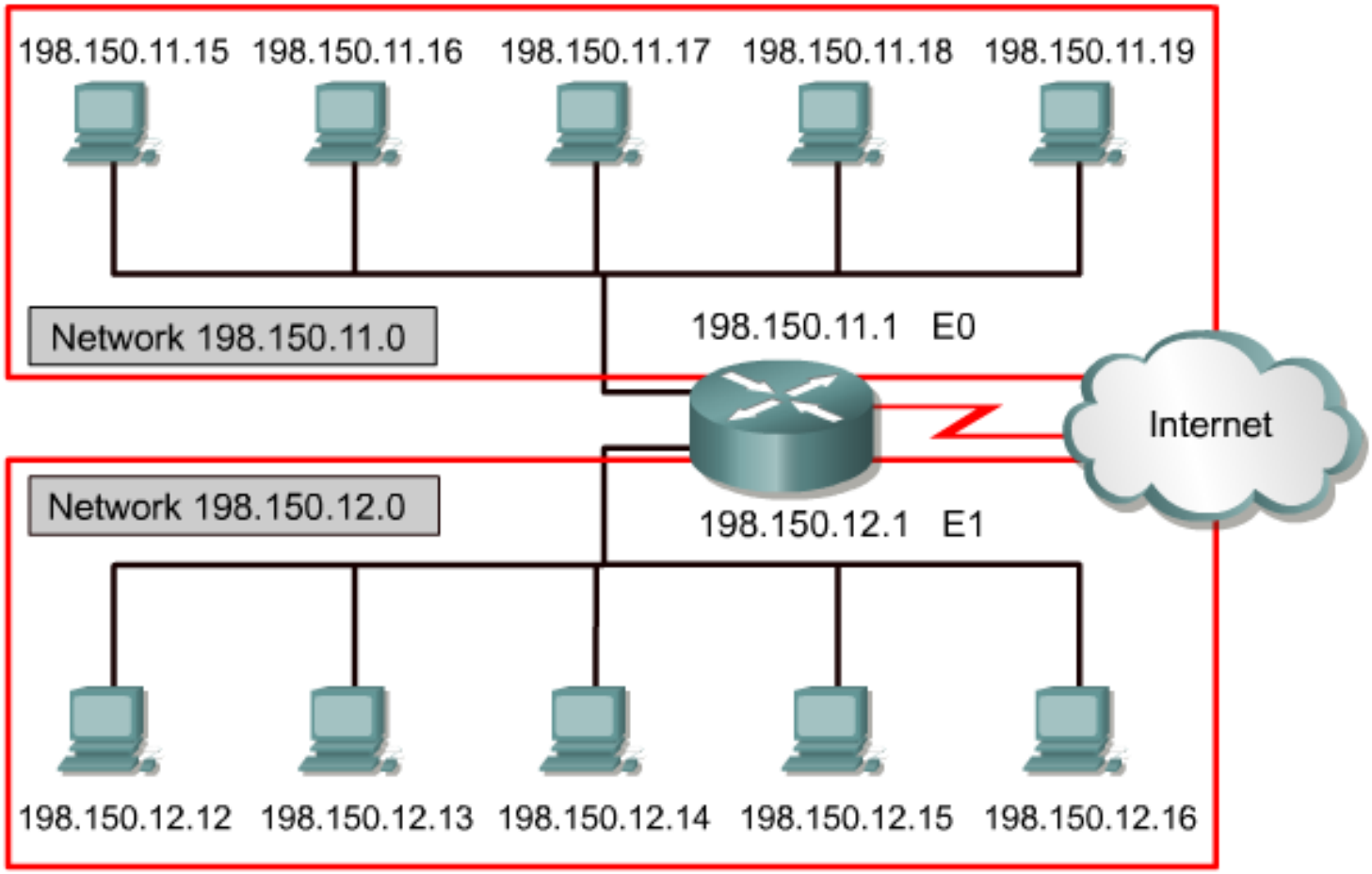
IP address class	IP address range (First Octet Decimal Value)
Class A	1-126 (00000001-01111110) *
Class B	128-191 (10000000-10111111)
Class C	192-223 (11000000-11011111)
Class D	224-239 (11100000-11101111)
Class E	240-255 (11110000-11111111)



Địa chỉ dành riêng

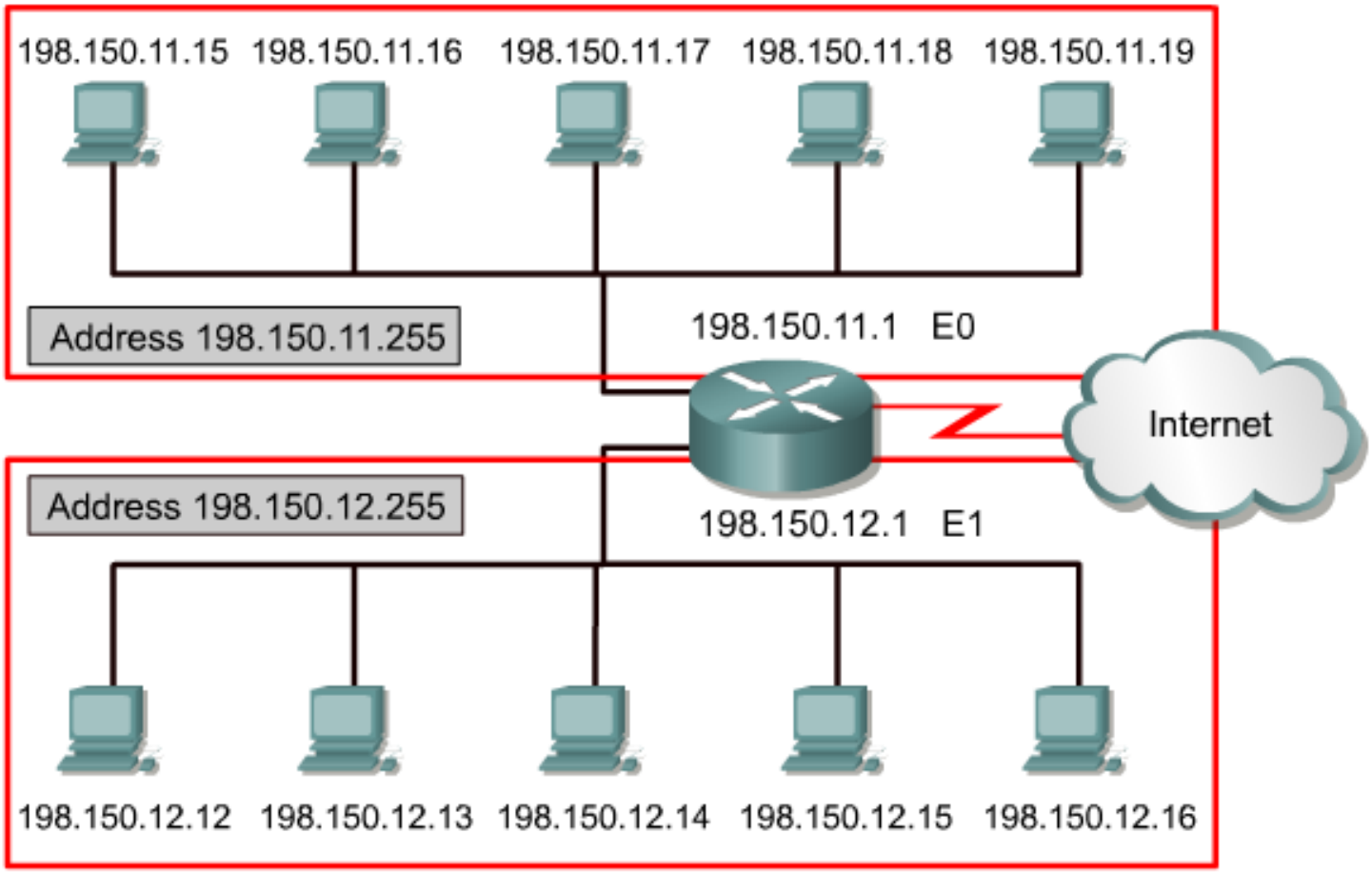
Class	RFC 1918 internal address range
A	10.0.0.0 to 10.255.255.255
B	172.16.0.0 to 172.31.255.255
C	192.168.0.0 to 192.168.255.255

Các lớp địa chỉ IP



Địa chỉ mạng

Các lớp địa chỉ IP



Địa chỉ broadcast

Các lớp địa chỉ IP

Lớp	Byte đầu tiên
A	0xxxxxxx
B	10xxxxxx
C	110xxxxx
D	1110xxxx
E	11110xxx

- 1.0.0.0 - 126.0.0.0 : Class A.
- 127.0.0.0 : Loopback network.
- 128.0.0.0 - 191.255.0.0 : Class B.
- 192.0.0.0 - 223.255.255.0 : Class C.

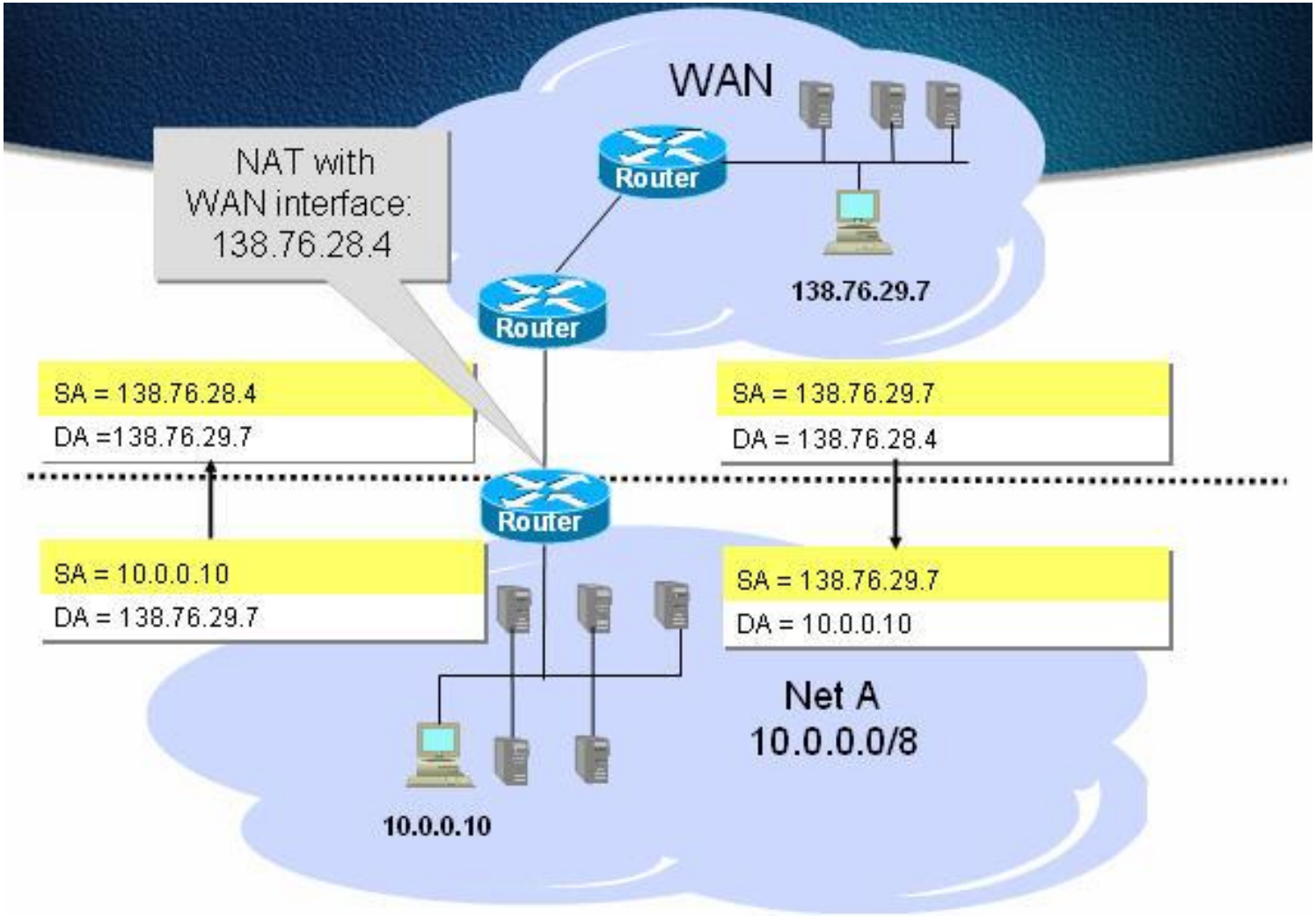
NAT: Network Address Translation

- Được thiết kế để tiết kiệm địa chỉ IP.
- Cho phép mạng nội bộ sử dụng địa chỉ IP riêng.
- Địa chỉ IP riêng sẽ được chuyển đổi sang địa chỉ công cộng định tuyến được.
- Mạng riêng được tách biệt và giấu kín IP nội bộ.
- Thường sử dụng trên router biên của mạng một cửa.

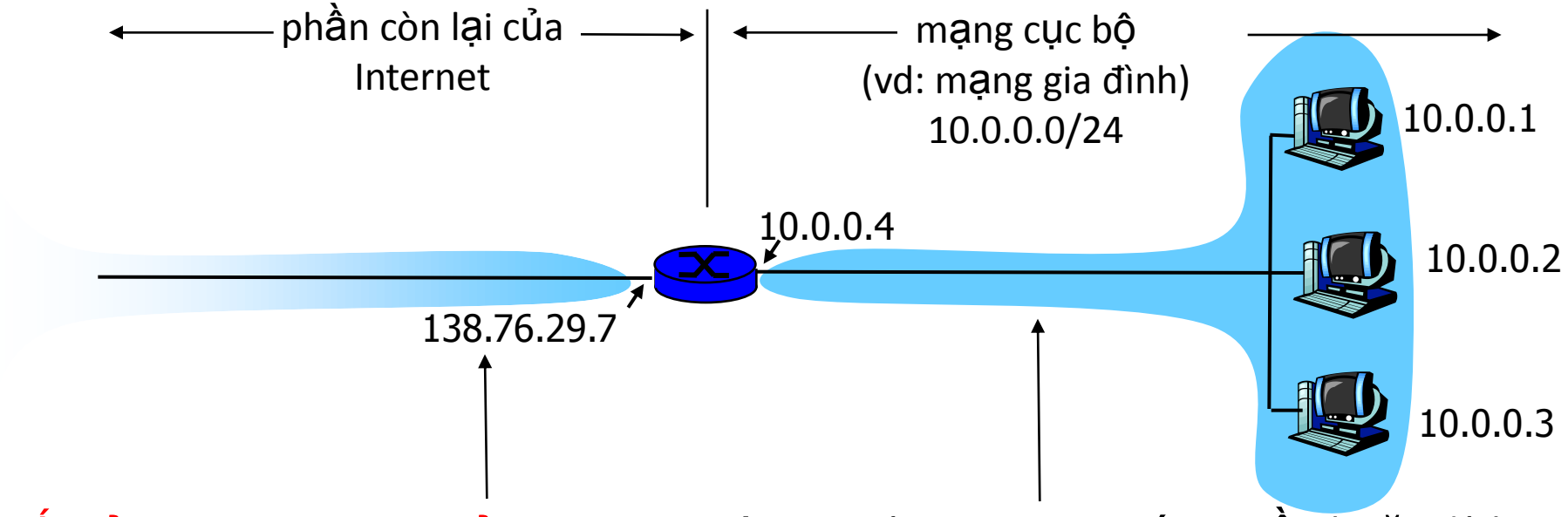
NAT

- Địa chỉ cục bộ bên trong (Inside local address): Địa chỉ được phân phối cho các host bên trong mạng nội bộ.
- Địa chỉ toàn cục bên trong (Inside global address): Địa chỉ hợp pháp được cung cấp bởi InterNIC (Internet Network Information Center) hoặc nhà cung cấp dịch vụ Internet, đại diện cho một hoặc nhiều địa chỉ nội bộ bên trong đối với thế giới bên ngoài.
- Địa chỉ cục bộ bên ngoài (Outside local address): Địa chỉ riêng của host nằm bên ngoài mạng nội bộ.
- Địa chỉ toàn cục bên ngoài (Outside global address): Địa chỉ công cộng hợp pháp của host nằm bên ngoài mạng nội bộ.

NAT


WAN
NAT with
WAN interface:
138.76.28.4
Router
Router
Router
138.76.29.7
SA = 138.76.28.4
DA =138.76.29.7
SA = 138.76.29.7
DA = 138.76.28.4
SA = 10.0.0.10
DA = 138.76.29.7
SA = 138.76.29.7
DA = 10.0.0.10
Net A
10.0.0.0/8
10.0.0.10

NAT



Tất cả datagram *đi ra khỏi* mạng cục bộ có cùng một địa chỉ IP NAT là: 138.76.29.7, với các số hiệu cổng nguồn khác nhau

các Datagram với nguồn hoặc đích trong mạng này có địa chỉ 10.0.0/24

NAT

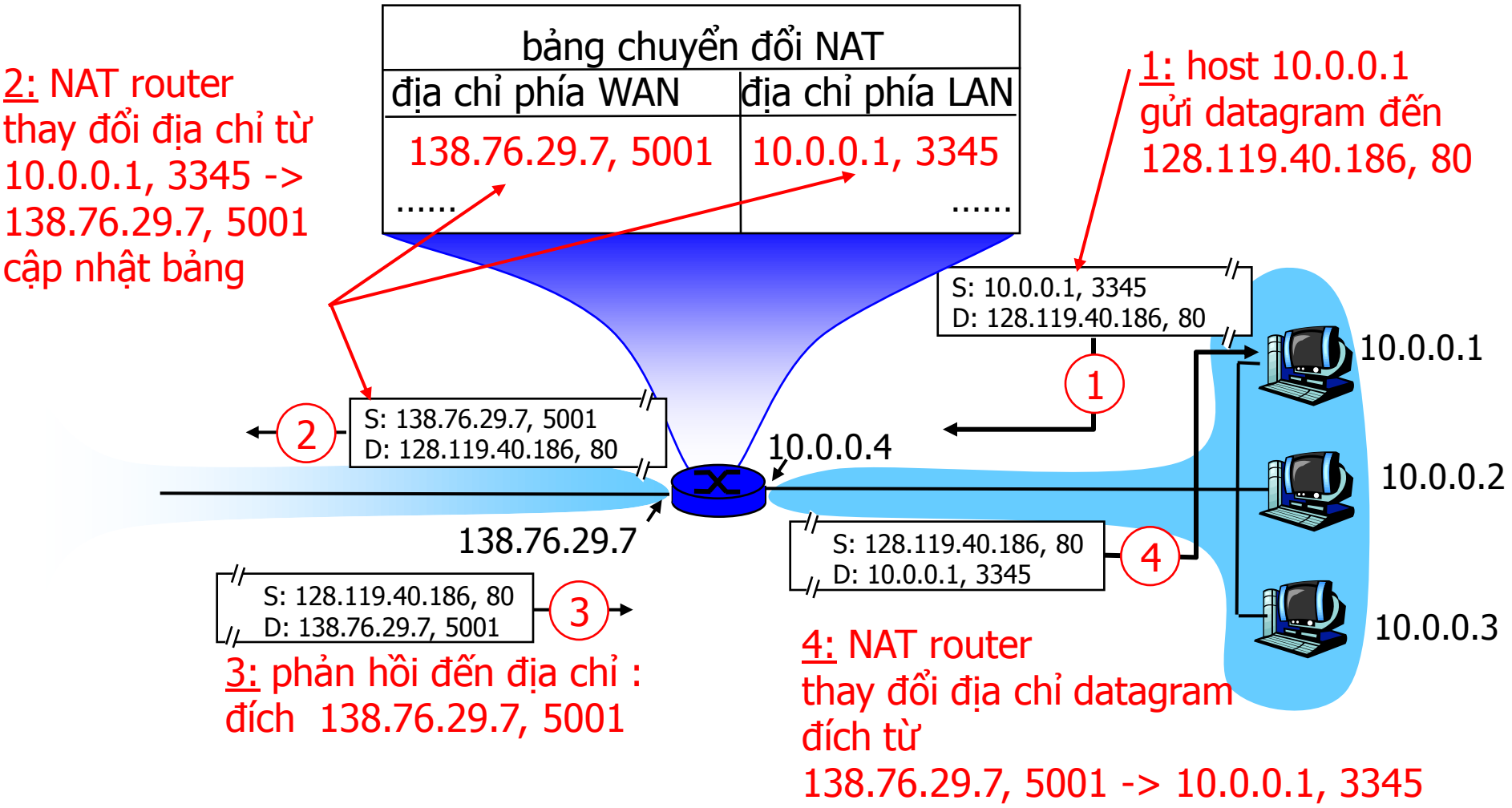
- Mạng cục bộ chỉ dùng 1 địa chỉ IP đối với bên ngoài:
  - không cần thiết dùng 1 vùng địa chỉ từ ISP: chỉ cần 1 cho tất cả các thiết bị
  - có thể thay đổi địa chỉ các thiết bị trong mạng cục bộ mà không cần thông báo với bên ngoài
  - có thể thay đổi ISP mà không cần thay đổi địa chỉ các thiết bị trong mạng cục bộ
  - các thiết bị trong mạng cục bộ không nhìn thấy, không định địa chỉ rõ ràng từ bên ngoài (tăng cường bảo mật)

NAT

Hiện thực: NAT router phải:

- *các datagram đi ra:* thay thế (địa chỉ IP và số hiệu cổng nguồn) mọi datagram đi ra bên ngoài bằng (địa chỉ NAT IP và số hiệu cổng nguồn mới)
  - . . . các clients/servers ở xa sẽ dùng (địa chỉ NAT IP và số hiệu cổng nguồn mới) đó như địa chỉ đích
- *ghi nhớ (trong bảng chuyển đổi NAT)* mọi cặp chuyển đổi (địa chỉ IP và số hiệu cổng nguồn) sang (địa chỉ NAT IP và số hiệu cổng nguồn mới)
- *các datagram đi đến:* thay thế (địa chỉ NAT IP và số hiệu cổng nguồn mới) trong các trường đích của mọi datagram đến với giá trị tương ứng (địa chỉ IP và số hiệu cổng nguồn) trong bảng NAT

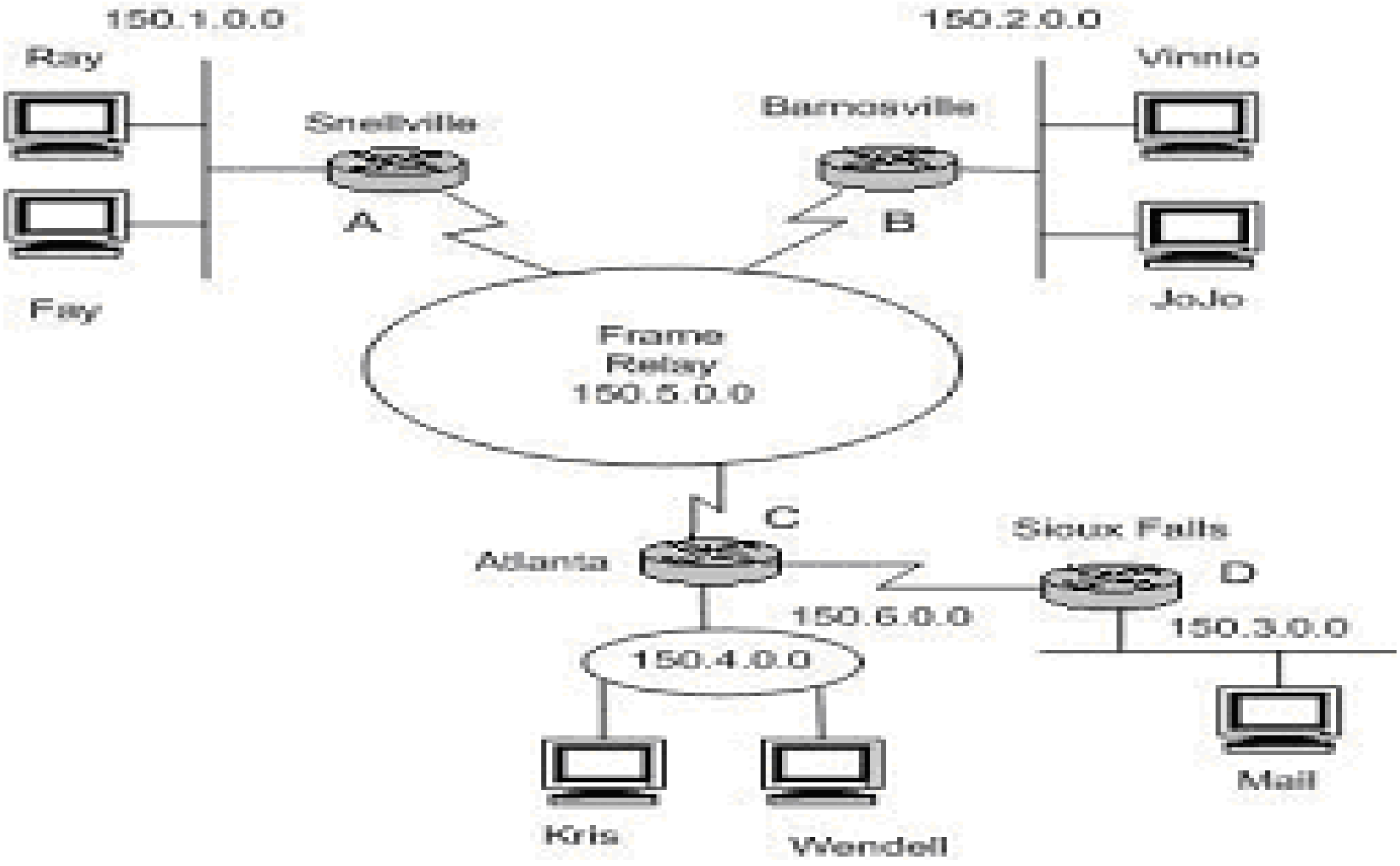
NAT


bảng chuyển đổi NAT
địa chỉ phía WAN
địa chỉ phía LAN
138.76.29.7, 5001
10.0.0.1, 3345
.....
.....
2: NAT router thay đổi địa chỉ từ 10.0.0.1, 3345 -> 138.76.29.7, 5001 cập nhật bảng
1: host 10.0.0.1 gửi datagram đến 128.119.40.186, 80
S: 10.0.0.1, 3345
D: 128.119.40.186, 80
1
2
S: 138.76.29.7, 5001
D: 128.119.40.186, 80
10.0.0.4
138.76.29.7
10.0.0.1
10.0.0.2
10.0.0.3
S: 128.119.40.186, 80
D: 10.0.0.1, 3345
4
S: 128.119.40.186, 80
D: 138.76.29.7, 5001
3
3: phản hồi đến địa chỉ : đích 138.76.29.7, 5001
4: NAT router thay đổi địa chỉ datagram đích từ 138.76.29.7, 5001 -> 10.0.0.1, 3345

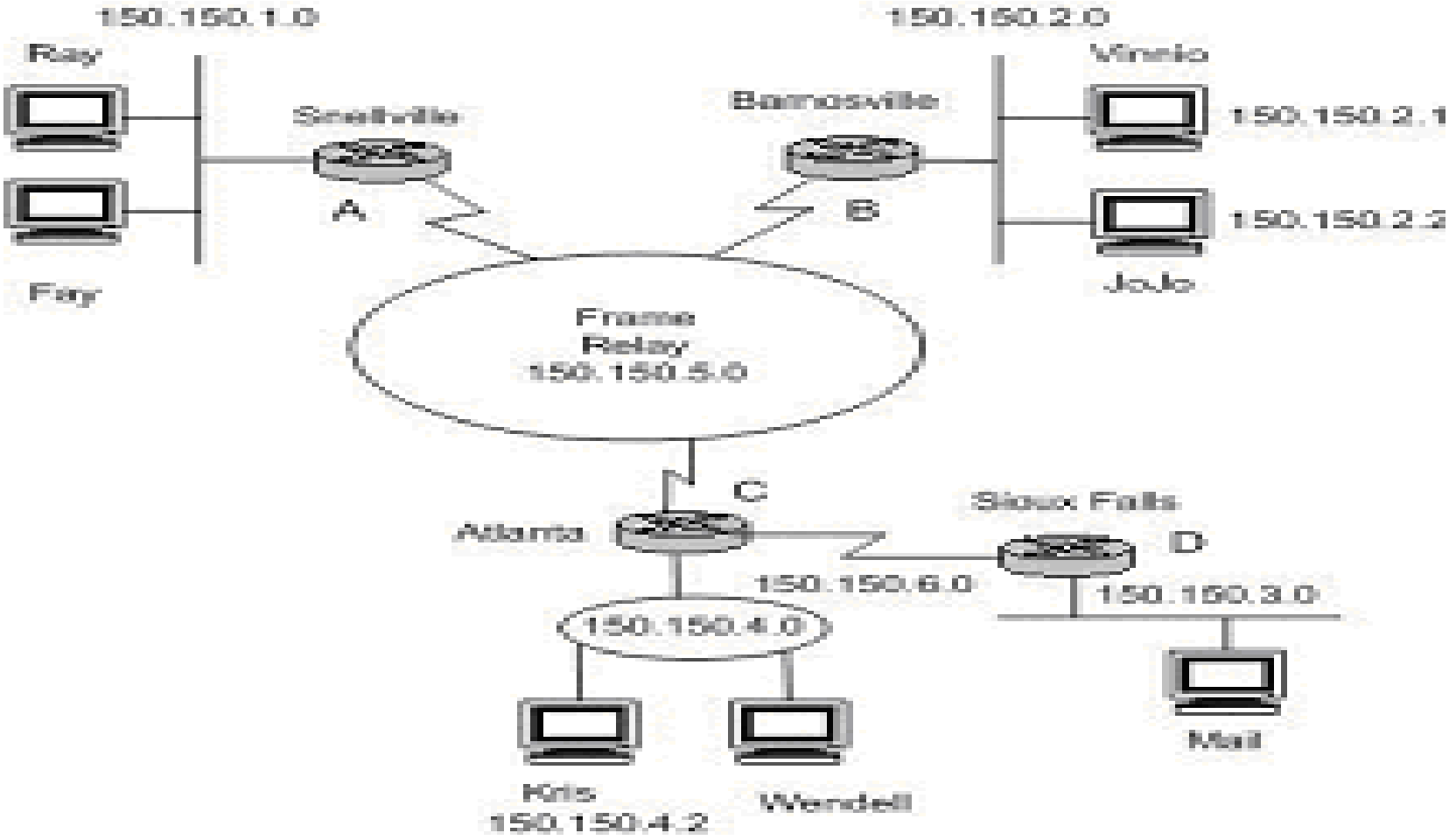
NAT

- Trường số hiệu cổng 16-bit:
  - Cho phép 60000 kết nối đồng thời chỉ với một địa chỉ phía WAN
- NAT còn có thể gây ra tranh luận:
  - các router chỉ xử lý đến lớp 3
  - vi phạm thỏa thuận end-to-end
    - những người thiết kế ứng dụng phải tính đến khả năng NAT, vd: ứng dụng P2P
  - sự thiếu thốn địa chỉ IP sẽ được giải quyết khi dùng IPv6

Mạng con

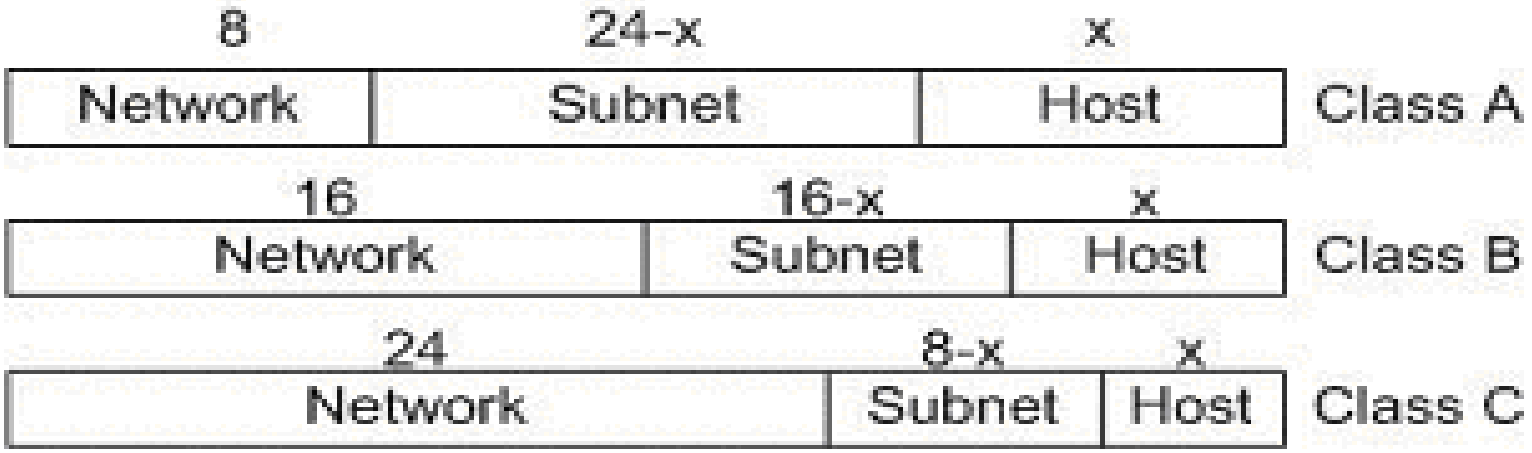

150.1.0.0
Ray
Fay
Snellville
A
150.2.0.0
Vinnie
JoJo
Barnesville
B
Frame
Relay
150.5.0.0
C
Atlanta
Sioux Falls
D
150.6.0.0
150.3.0.0
150.4.0.0
Mail
Kris
Wendell

Mạng con


150.150.1.0
Ray
Snellville
A
Fay
150.150.2.0
Vinnie
Barnesville
B
150.150.2.1
150.150.2.3
JoJo
Frame
Relay
150.150.5.0
C
Atlanta
Sioux Falls
D
150.150.6.0
150.150.3.0
150.150.4.0
Mail
Kris
150.150.4.2
Wendell

Kỹ thuật chia mạng con

- Mượn một số bit trong phần host_id ban đầu để đặt cho các mạng con
- Cấu trúc của địa chỉ IP lúc này sẽ gồm 3 phần: network_id, subnet_id và host_id.



Kỹ thuật chia mạng con

- Số bit dùng trong subnet_id tuỳ thuộc vào chiến lược chia mạng con. Tuy nhiên số bit tối đa có thể mượn phải tuân theo công thức:

  Subnet_id <= host_id - 2

- Số lượng bit tối đa có thể mượn:
  - Lớp A: 22 (= 24 – 2) bit -> chia được 2^{22} = 4194304 mạng con
  - Lớp B: 14 (= 16 – 2) bit -> chia được 2^{14} = 16384 mạng con
  - Lớp C: 06 (= 8 – 2) bit -> chia được 2^{6} = 64 mạng con

Kỹ thuật chia mạng con

- Số bit trong phần subnet_id xác định số lượng mạng con. Với số bit là x thì 2^x là số lượng mạng con có được.
- Ngược lại từ số lượng mạng con cần thiết theo nhu cầu, tính được phần subnet_id cần bao nhiêu bit. Nếu muốn chia 6 mạng con thì cần 3 bit (2^3=8), chia 12 mạng con thì cần 4 bit (2^4>=12).

Một số khái niệm mới

- Địa chỉ mạng con (địa chỉ đường mạng): gồm cả phần network_id và subnet_id, phần host_id chỉ chứa các bit 0
- Địa chỉ broadcast trong một mạng con: tất cả các bit trong phần host_id là 1.
- Mặt nạ mạng con (subnet mask): tất cả các bit trong phần host_id là 0, các phần còn lại là 1.

Quy ước ghi địa chỉ IP

- Nếu có địa chỉ IP như 172.29.8.230 thì chưa thể biết được host này nằm trong mạng nào, có chia mạng con hay không và có nếu chia thì dùng bao nhiêu bit để chia. Chính vì vậy khi ghi nhận địa chỉ IP của một host, phải cho biết subnet mask của nó
- Ví dụ: 172.29.8.230/255.255.255.0 hoặc 172.29.8.230/24 (có nghĩa là dùng 24 bit đầu tiên cho NetworkID).

Kỹ thuật chia mạng con

- Thực hiện 3 bước:
  - Bước 1: Xác định lớp (class) và subnet mask mặc nhiên của địa chỉ.
  - Bước 2: Xác định số bit cần mượn và subnet mask mới, tính số lượng mạng con, số host thực sự có được.
  - Bước 3: Xác định các vùng địa chỉ host và chọn mạng con muốn dùng

Bài tập 1

Cho địa chỉ IP sau: 172.16.0.0/16.

Hãy chia thành 8 mạng con và có tối thiểu 1000 host trên mỗi mạng con đó.

Bước 1: Xác định class và subnet mask mặc nhiên

Giải:

- Địa chỉ trên viết dưới dạng nhị phân
  10101100.00010000.00000000.00000000
- Xác định lớp của IP trên:
  [redacted]ớp B
- Xác định Subnet mask mặc nhiên:
  [redacted]55.255.0.0

Bước 2: Số bit cần mượn...

- Cần mượn bao nhiêu bit:
  - N = 3, bởi vì:
    - Số mạng con có thể: $2^3 = 8$.
    - Số host của mỗi mạng con có thể:
      $2^{(16-3)} - 2 = 2^{13} - 2 > 1000$.
- Xác định Subnet mask mới:

11111111.11111111.11100000.00000000

hay 255.255.224.0

Bước 3: Xác định vùng địa chỉ host

STT			
1	172.16.0.0	172.16.0.1 - 172.16.31.254	172.16.31.255
2	172.16.32.0	172.16.32.1 - 172.16.63.254	172.16.63.255
…	…		
7	172.16.192.0		
8	172.16.224.0	172.16.224.1 – 172.16.255.254	172.16.255.255

10101100.00010000.000**00000.00000001**
Đến
10101100.00010000.000**11111.11111111**

10101100.00010000.001**00000.00000001**
Đến
10101100.00010000.001**11111.11111110**

10101100.00010000.**001**00000.00000000

10101100.00010000.**001**11111.11111111

Bài tập 2

Cho 2 địa chỉ IP sau:

192.168.5.9/28

192.168.5.39/28

- Hãy cho biết các địa chỉ network, host của từng IP trên?
- Các máy trên có cùng mạng hay không ?
- Hãy liệt kê tất cả các địa chỉ IP thuộc các mạng vừa tìm được?

Địa chỉ IP thứ nhất: 192.168.5.9/28

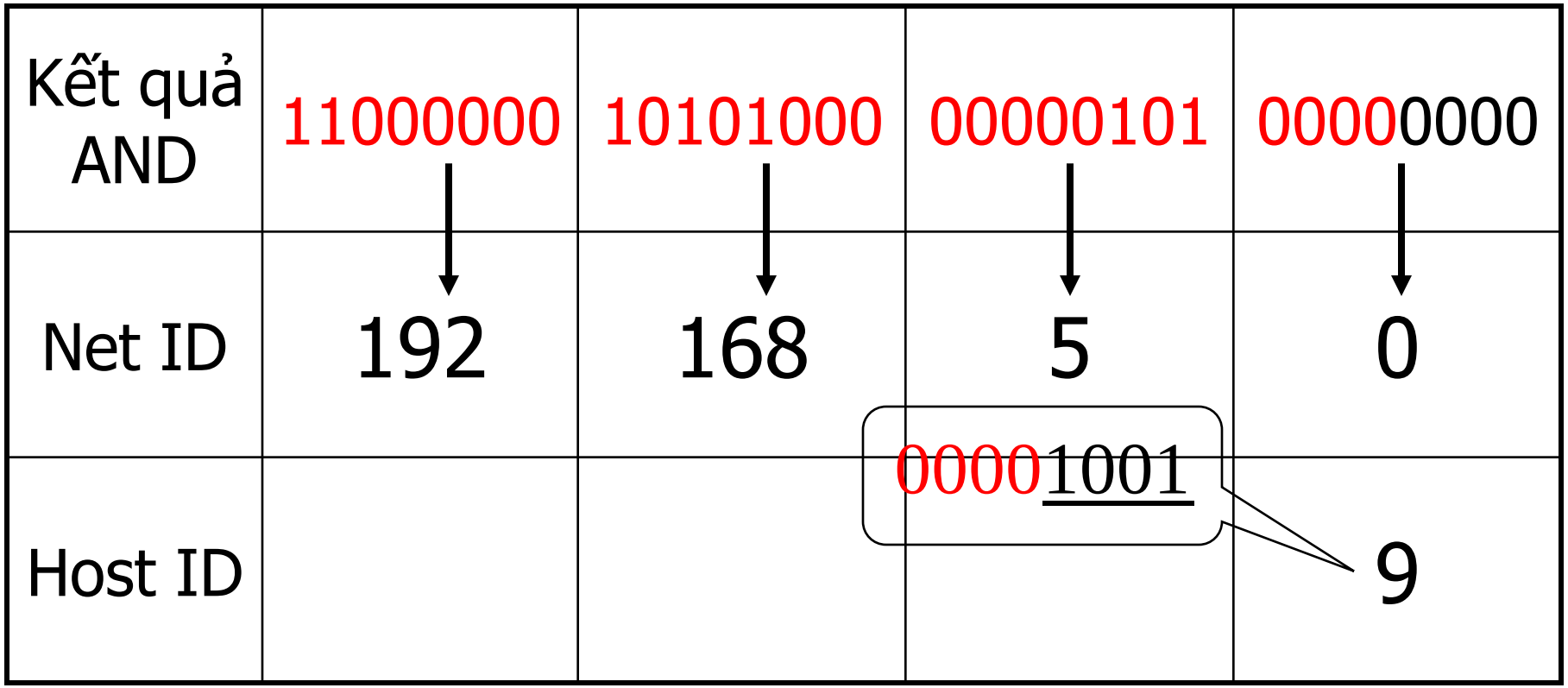
- Chú ý: 28 là số bit dành cho NetworkID
- Đây là IP thuộc lớp C
- Subnet mask mặc nhiên: 255.255.255.0

IP (thập phân)	192	168	5	9
IP (nhị phân)	11000000	10101000	00000101	00001001

Thực hiện AND địa chỉ IP với Subnet mask

IP	11000000	10101000	00000101	00001001
Subnet mask	11111111	11111111	11111111	11110000
Kết quả AND	11000000	10101000	00000101	00000000

Chuyển IP sang dạng thập phân



Địa chỉ IP thứ hai: 192.168.5.39/28

IP	192	168	5	39
IP (nhị phân)	11000000	10101000	00000101	00100111
Subnet Mask	11111111	11111111	11111111	11110000
AND	11000000	10101000	00000101	00100000
Network ID	192	168	5	32
HostID				7

Hai địa chỉ trên có cùng mạng?

- 192.168.5.9/28
- 192.168.5.39/28

Kết luận: Hai địa chỉ trên không cùng mạng

Net ID của địa chỉ thứ 1	192	168	5	0
Net ID của địa chỉ thứ 2	192	168	5	32

Liệt kê tất cả các địa chỉ IP

Mạng tương ứng với IP	Vùng địa chỉ HostID với dạng nhị phân	Vùng địa chỉ HostID với dạng thập phân
1	11000000.10101000.00000101.00000001 Đến 11000000.10101000.00000101.00001110	192.168.5.1/28 Đến 192.168.5.14/28
2	11000000.10101000.00000101.00100001 Đến 11000000.10101000.00000101.00101110	192.168.5.33/28 Đến 192.168.5.46/28

Bài tập 3

Hãy xét đến một địa chỉ IP class B, **139.12.0.0**, với subnet mask là **255.255.0.0**. Một Network với địa chỉ thế này có thể chứa 65534 nodes hay computers. Đây là một con số quá lớn, trên mạng sẽ có đầy broadcast traffic. Hãy chia network thành 5 mạng con.

Bước 1: Xác định Subnet mask

- Để chia thành 5 mạng con thì cần thêm 3 bit (vì $2^3 > 5$).
- Do đó Subnet mask sẽ cần: 16 (bits trước đây) + 3 (bits mới) = 19 bits
- Địa chỉ IP mới sẽ là **139.12.0.0/19** (để ý con số 19 thay vì 16 như trước đây).

Bước 2: Liệt kê ID của các Subnet mới

Subnet mask với dạng nhị phân	Subnet mask với dạng thập phân
11111111.11111111.11100000.00000000	255.255.224.0

NetworkID của bốn Subnets mới

TT	Subnet ID với dạng nhị phân	Subnet ID với dạng thập phân
1	10001011.00001100.00000000.00000000	139.12.0.0/19
2	10001011.00001100.00100000.00000000	139.12.32.0/19
3	10001011.00001100.01000000.00000000	139.12.64.0/19
4	10001011.00001100.01100000.00000000	139.12.96.0/19
5	10001011.00001100.10000000.00000000	139.12.128.0/19

Bước 3: Cho biết vùng địa chỉ IP của các HostID

TT	Dạng nhị phân	Dạng thập phân
1	10001011.00001100.00000000.00000001 10001011.00001100.00011111.11111110	139.12.0.1/19 - 139.12.31.254/19
2	10001011.00001100.00100000.00000001 10001011.00001100.00111111.11111110	139.12.32.1/19 - 139.12.63.254/19
3	10001011.00001100.01000000.00000001 10001011.00001100.01011111.11111110	139.12.64.1/19 - 139.12.95.254/19
4	10001011.00001100.01100000.00000001 10001011.00001100.01111111.11111110	139.12.96.1/19 - 139.12.127.254/19
5	10001011.00001100.10000000.00000001 10001011.00001100.10011111.11111110	139.12.128.1/19 - 139.12.159.254/19

Tính nhanh vùng địa chỉ IP

- n – số bit làm subnet
- Số mạng con: $S = 2^n$
- Số gia địa chỉ mạng con: $M = 2^{8-n}$ $(n \leq 8)$
- Byte cuối của IP địa chỉ mạng, ví dụ lớp C: (k-1)*M (với k=1,2,…)
- Byte cuối của IP host đầu tiên, ví dụ lớp C: (k-1)*M + 1 (với k=1,2,…)
- Byte cuối của IP host cuối cùng, ví dụ lớp C: k*M - 2 (với k=1,2,…)
- Byte cuối của IP broadcast, ví dụ lớp C: k*M - 1 (với k=1,2,…)

Ví dụ tính nhanh vùng địa chỉ IP

- Cho địa chỉ: 192.168.0.0/24
- Với n=4 → M= 16 (= 2^{8-4}) →
  - Network 1: 192.168.0.0. Host range: 192.168.0.1–192.168.0.14. Broadcast: 192.168.0.15
  - Network 2: 192.168.0.16. Host range: 192.168.0.17–192.168.0.30. Broadcast: 192.168.0.31
  - Network 3: 192.168.0.32. Host range: 192.168.0.33–192.168.0.46. Broadcast: 192.168.0.47
  - Network 4: 192.168.0.48. Host range: 192.168.0.49–192.168.0.62. Broadcast: 192.168.0.63

Bài tập 4

- Cho địa chỉ IP: 102.16.10.107/12
  - Tìm địa chỉ mạng con? Địa chỉ host
  - Dải địa chỉ host có cùng mạng với IP trên?
  - Broadcast của mạng mà IP trên thuộc vào?

Bước: Tính subnet mask

- 102.16.10.107/12 ➔
- Subnet mask: 11111111.1111**0000**.00000000.00000000
- Byte đầu tiên chắc chắn khi dùng phép toán AND ra kết quả bằng 102 → không cần đổi 102 sang nhị phân

Trả lời câu hỏi 1: Địa chỉ mạng con?

- Xét byte kế tiếp là: 16 (10) → 00010000 (2)
- Khi AND byte này với Subnet mask, ta được kết quả là: 00010000 (2)
- Như vậy địa chỉ mạng con sẽ là:

102.16.0.0/12

- Như vậy địa chỉ host sẽ là:

0.10.107

Trả lời câu hỏi 2: Dải địa chỉ host? Broadcast?

- Dải địa chỉ host sẽ từ:

01100110 00010000 00000000 00000001

(hay 102.16.0.1/12)

Đến:

01100110 00011111 11111111 11111110

(hay 102.31.255.254/12)

- Broadcast:

102.31.255.255/12

Bài tập 5: Cho IP 172.19.160.0/21

- Chia làm 4 mạng con
- Liệt kê các thông số gồm địa chỉ mạng, dãy địa chỉ host, địa chỉ broadcast của các mạng con đó

Giải BT 5

- Chia làm 4 mạng con nên phải mượn 2 bit
- Do /21 nên 2 byte đầu tiên của IP đã cho không thay đổi. Xét byte thứ 3
- 160 = 10100000$_{(2)}$
- Phần 2 bit 00 là nơi ta mượn làm subnet

Giải BT 5 (tt)

- Xét byte thứ 3
- Mạng con thứ 1: $10100000_{(2)}$
- Mạng con thứ 2: $10100010_{(2)}$
- Mạng con thứ 3: $10100100_{(2)}$
- Mạng con thứ 4: $10100110_{(2)}$

Giải BT 5 (tt)

Địa chỉ mạng	Dải địa chỉ host	Địa chỉ broadcast
172.19.160.0	172.19.160.1 đến 172.19.161.254	172.19.161.255
172.19.162.0	172.19.162.1 đến 172.19.163.254	172.19.163.255
172.19.164.0	172.19.164.1 đến 172.19.165.254	172.19.165.255
172.19.166.0	172.19.166.1 đến 172.19.167.254	172.19.167.255

Bài tập 6: Cho IP 172.16.192.0/18

- Chia làm 4 mạng con
- Liệt kê các thông số gồm địa chỉ mạng, dãy địa chỉ host, địa chỉ broadcast của các mạng con đó

Giải BT 6

- Chia làm 4 mạng con nên phải mượn 2 bit
- Do /18 nên 2 byte đầu tiên của IP đã cho không thay đổi. Xét byte thứ 3
- 192 = 11000000$_{(2)}$
- Phần 2 bit 00 là nơi ta mượn làm subnet

Giải BT 6 (tt)

- Xét byte thứ 3
- Mạng con thứ 1: $11000000_{(2)}$
- Mạng con thứ 2: $11010000_{(2)}$
- Mạng con thứ 3: $11100000_{(2)}$
- Mạng con thứ 4: $11110000_{(2)}$

Giải BT 6 (tt)

Địa chỉ mạng	Dải địa chỉ host	Địa chỉ broadcast
172.16.192.0	172.16.192.1 đến 172.16.207.254	172.16.207.255
172.16.208.0	172.16.208.1 đến 172.16.223.254	172.16.223.255
172.16.224.0	172.16.224.1 đến 172.16.239.254	172.16.239.255
172.16.240.0	172.16.240.1 đến 172.16.255.254	172.16.255.255

CHƯƠNG 6: BẢO MẬT MẠNG

- Hiểu các nguyên lý của bảo mật mạng:
  - mật mã
  - chứng thực
  - tính toàn vẹn
  - khóa phân bố
- Bảo mật trong thực tế:
  - các firewall
  - bảo mật trong các lớp application, transport, network, data-link

Bảo mật mạng là gì?

Sự bảo mật: chỉ có người gửi, người nhận mới "hiểu" được nội dung thông điệp

- người gửi mã hóa thông điệp
- người nhận giải mã thông điệp

Chứng thực: người gửi, người nhận xác định là nhận ra nhau

Sự toàn vẹn thông điệp: người gửi, người nhận muốn bảo đảm thông điệp không bị thay đổi (trên đường truyền hoặc sau khi nhận)

Truy cập & tính sẵn sàng: các dịch vụ phải có khả năng truy cập và sẵn sàng đối với các user

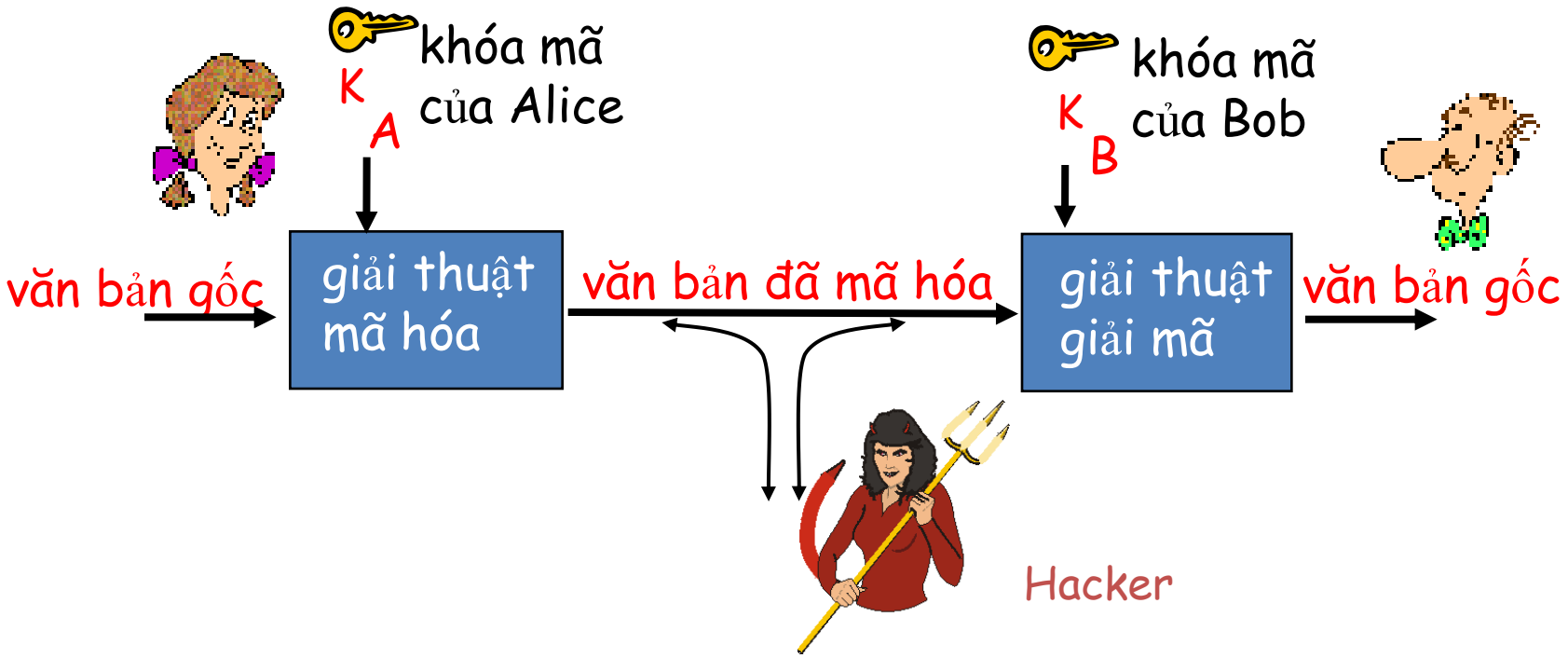
Các đối tượng cần bảo mật

- Trình duyệt Web/server cho các giao dịch điện tử
- Client/Server ngân hàng trực tuyến
- DNS servers
- Các router trao đổi thông tin cập nhật bảng routing
- .v.v.

Kẻ xấu có thể làm những việc gì?

- *nghe lén:* ngăn chặn các thông điệp
- kích hoạt *chèn* các thông điệp vào trong kết nối
- *giả danh:* có thể giả mạo địa chỉ nguồn trong gói (hoặc bất kỳ trường nào trong đó)
- *cướp:* "tiếp tục" kết nối hiện hành nhưng thay người gửi hoặc người nhận bằng chính họ
- *từ chối dịch vụ:* dịch vụ hiện tại bị người khác dùng (đồng nghĩa quá tải)
- .v.v.

Các nguyên lý mã hóa



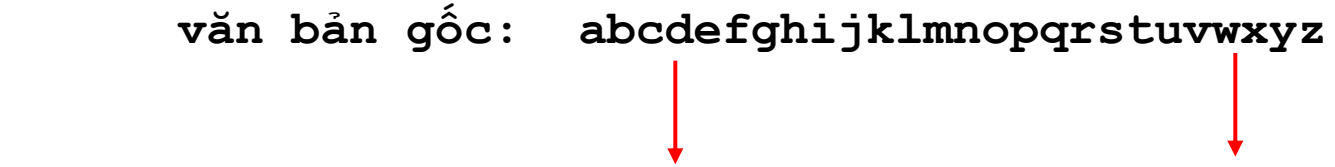
khóa đối xứng: khóa bên gửi và bên nhận giống nhau

khóa công cộng: khóa mã chung, khóa giải mã bí mật (riêng)

Mã hóa khóa đối xứng

mật mã thay thế: thay thứ này thành thứ khác

– mã hóa ký tự đơn: thay thế từng ký tự một



ví dụ: văn bản gốc: Bob. i love you. Alice

mã hóa thành: nko. s gktc wky. mgsbc

- Bẻ khóa kiểu mã hóa đơn giản này dễ không?
  - ❑ brute force (khó như thế nào?)
  - ❑ khác?

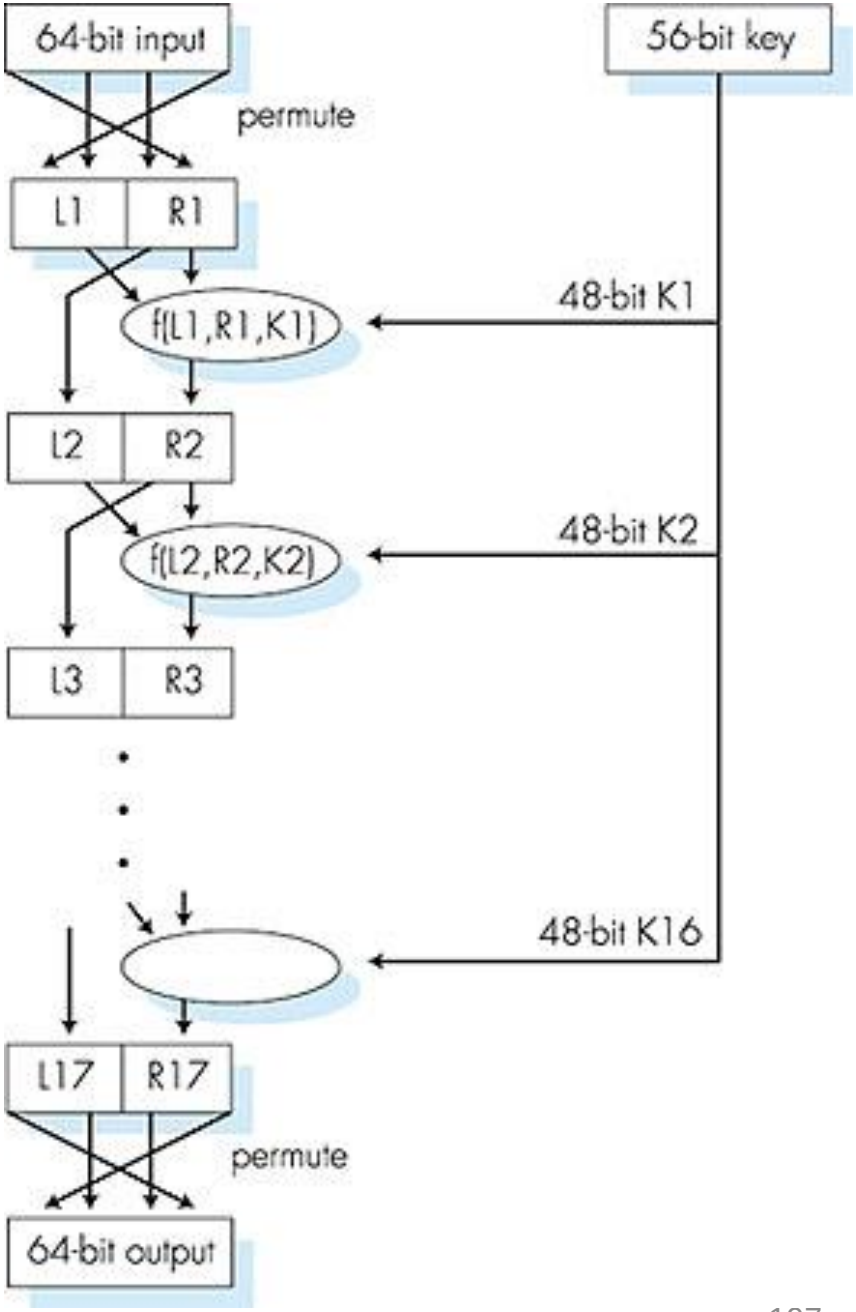
Mã hóa khóa đối xứng: DES

DES: Data Encryption Standard

- Chuẩn mã hóa của Hoa Kỳ [NIST 1993]
- Khóa đối xứng 56-bit, văn bản gốc vào 64-bit
- Bảo mật trong DES như thế nào?
  - chưa có cách tiếp cận "backdoor-cửa sau" để giải mã
- làm cho DES bảo mật hơn:
  - dùng 3 khóa tuần tự (3-DES) trong mỗi datum
  - dùng cơ chế liên kết khối mã

Mã hóa khóa đối xứng: DES

DES hoạt động



AES: Advanced Encryption Standard

- Chuẩn NIST khóa đối xứng mới (tháng 11-2001) thay thế cho DES
- Dữ liệu xử lý từng khối 128 bit
- Các khóa 128, 192 hoặc 256 bit
- Giải mã brute force (thử sai) tốn 1s với DES, tốn 149 tỷ tỷ năm với AES

Mã hóa khóa công cộng

khóa đối xứng

- yêu cầu người gửi, người nhận phải biết khóa công cộng
- Làm sao biết khóa công cộng đó trong lần đầu tiên (đặc biệt với những người chưa bao giờ gặp trước)?

Mã hóa khóa công cộng

- tiếp cận khác hoàn toàn
- người gửi, người nhận không chia sẻ khóa công cộng
- khóa công cộng cho mọi người đều biết
- khóa giải mã riêng chỉ có người nhận biết

Giải thuật mã hóa khóa công cộng

Yêu cầu:

1. cần $K_B^+(.)$ và $K_B^-(.)$ như sau:

$$K_B^-(K_B^+(m)) = m$$

2. cho khóa công cộng K_B^+, phải không thể tính toán ra được khóa riêng K_B^-

giải thuật RSA: Rivest, Shamir, Adelson

Sự chứng thực

Mục tiêu: Bob muốn Alice "chứng thực" nhân dạng của cô đối với anh ta

Mô tả cách thức hiện thực: Alice nói "Tôi là Alice"

Thất bại sẽ xảy ra??

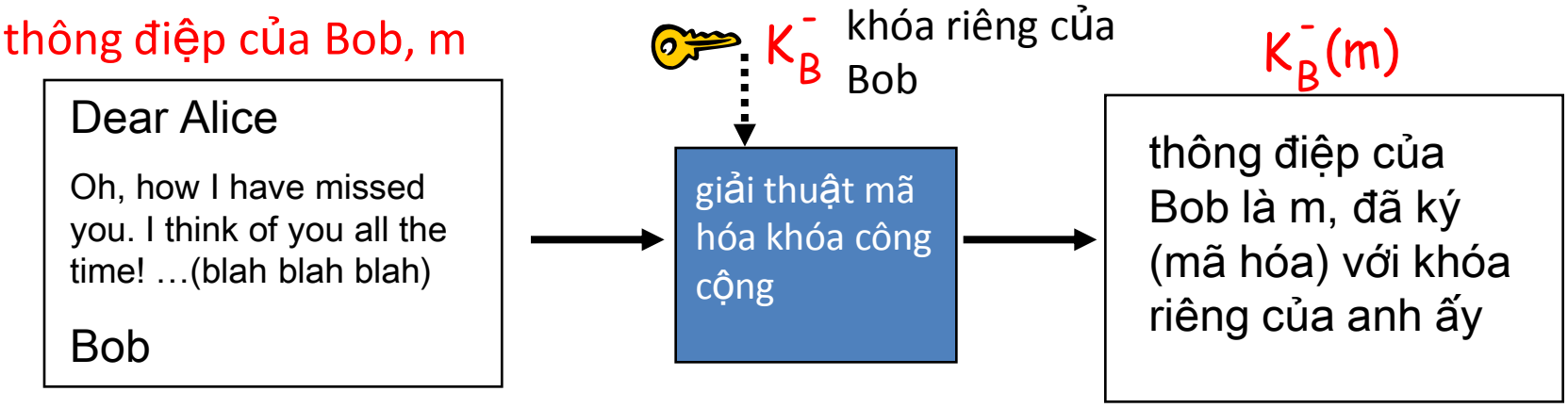
Sự toàn vẹn

- Chữ ký số: Kỹ thuật mã hóa tương tự như các chữ ký bằng tay.
  - người gửi (Bob) đánh dấu (số hóa) tài liệu, thiết lập thuộc tính là người sở hữu/tạo lập tài liệu.
  - có thể kiểm tra, không thể làm giả: người nhận (Alice) có thể chứng thực với người khác là chỉ có Bob chứ ngoài ra không có ai (kể cả Alice) đã ký trên tài liệu đó.

Chữ ký số

Chữ ký số đơn giản cho thông điệp m:

- Bob ký m bằng cách mã hóa với khóa riêng của anh ấy K_B^-, tạo thống điệp "đã được ký", $K_B^-(m)$



Chữ ký số (tt)

- Giả sử Alice nhận được m, với chữ ký số hóa là $K_B(m)$
- Alice kiểm tra m đã được ký bởi Bob bằng cách áp dụng khóa công cộng của Bob là K_B cho $K_B(m)$ sau đó kiểm tra $K_B(K_B(m)) = m$.
- Nếu $K_B(K_B(m)) = m$, bất cứ ai đã ký m phải dùng khóa riêng của Bob

Alice kiểm tra:

- ✓ Bob đã ký m.
- ✓ Không có ai khác đã ký m.
- ✓ Bob đã ký m và không ký m'.

Không thể phủ nhận:

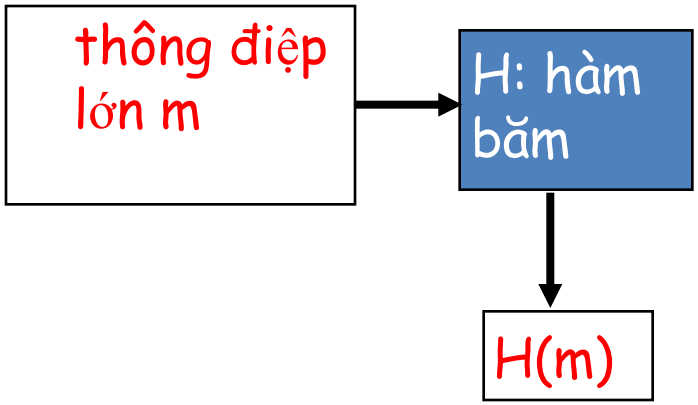
- ✓ Alice có thể giữ m và chữ ký $K_B(m)$ để chứng thực rằng Bob đã ký m.

Phân loại thông điệp

Tính toán các thông điệp dài có chi phí đắt

Mục tiêu: "dấu tay" số hóa có kích thước cố định, dễ tính toán được

- áp dụng hàm băm H vào m, tính được phân loại thông điệp kích thước cố định, *H(m).*



Các đặc tính hàm băm:

- nhiều-một
- sinh ra phân loại thông điệp kích thước cố định ("dấu tay")
- cho phân loại thông điệp x, không thể tính toán để tìm m dùng x = H(m)

Khóa phân bố và chứng chỉ

Vấn đề khóa đối xứng:

- Làm thế nào 2 thực thể cùng thiết lập khóa bí mật trên mạng?

Giải pháp:

- Trung tâm phân bố khóa (key distribution center-KDC) được tin cậy – hoạt động trung gian giữa các thực thể

Vấn đề khóa công cộng:

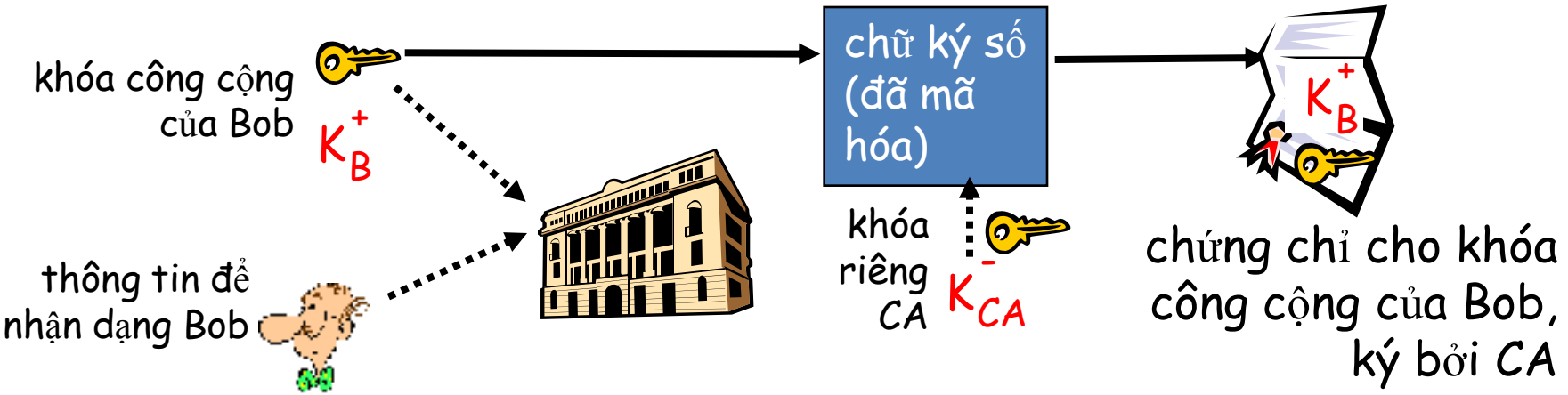
- Khi Alice lấy được khóa công cộng của Bob (từ web site, email, đĩa) làm sao biết khóa công cộng của Bob chứ không phải của Hacker?

Giải pháp:

- nơi cấp chứng chỉ (certification authority-CA) được tin cậy

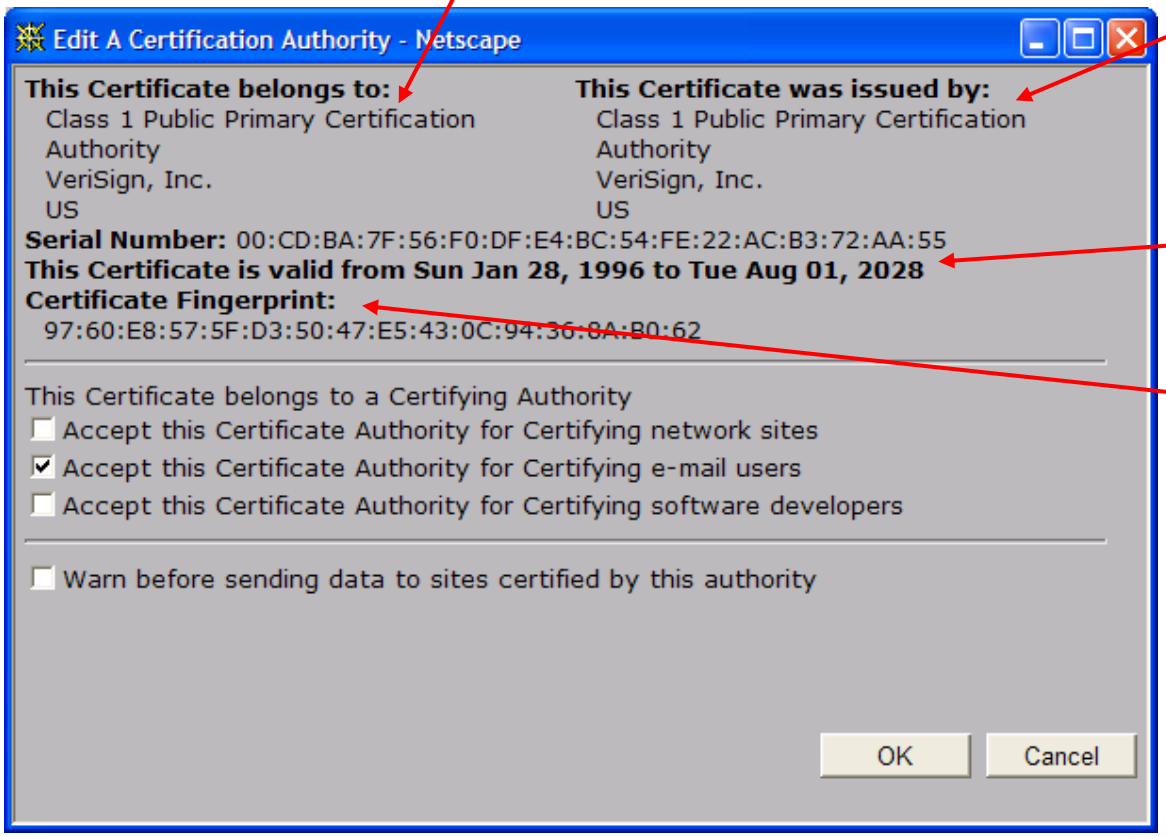
Cấp chứng chỉ

- Certification authority (CA): gắn kết khóa công cộng với thực thể E nào đó.
- E (người, router) đăng ký khóa công cộng của họ với CA.
  - E cung cấp "bằng chứng để nhận dạng" cho CA.
  - CA tạo ra chứng chỉ ràng buộc E với khóa công cộng của nó.
  - chứng chỉ chứa khóa công cộng của E được ký số bởi CA – CA nói "đây là khóa công cộng của E"



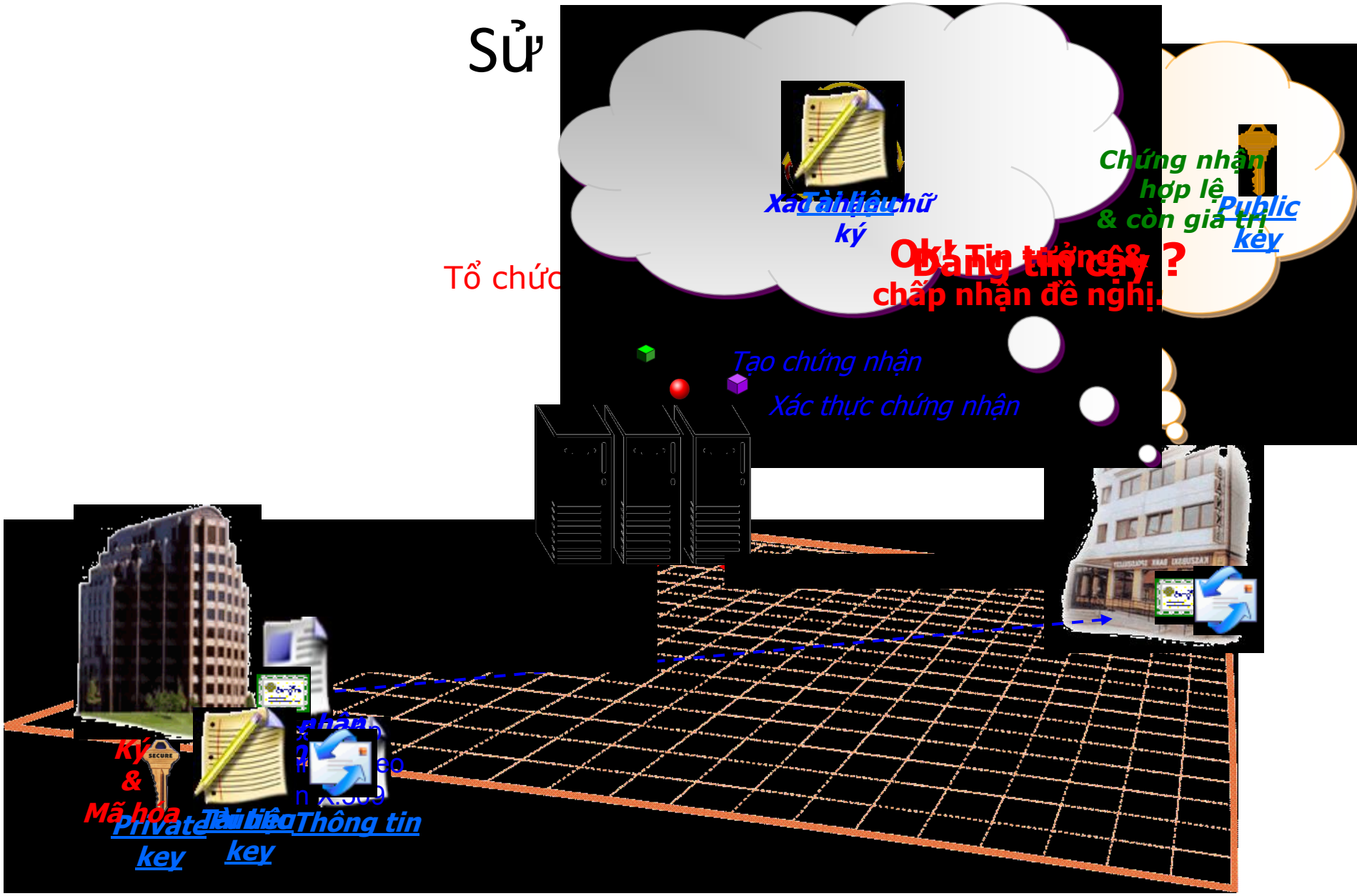
Mô tả chứng chỉ

- Số thứ tự (duy nhất)
- thông tin về người sở hữu chứng chỉ, bao gồm giải thuật và chính giá trị khóa (không hiển thị ra)

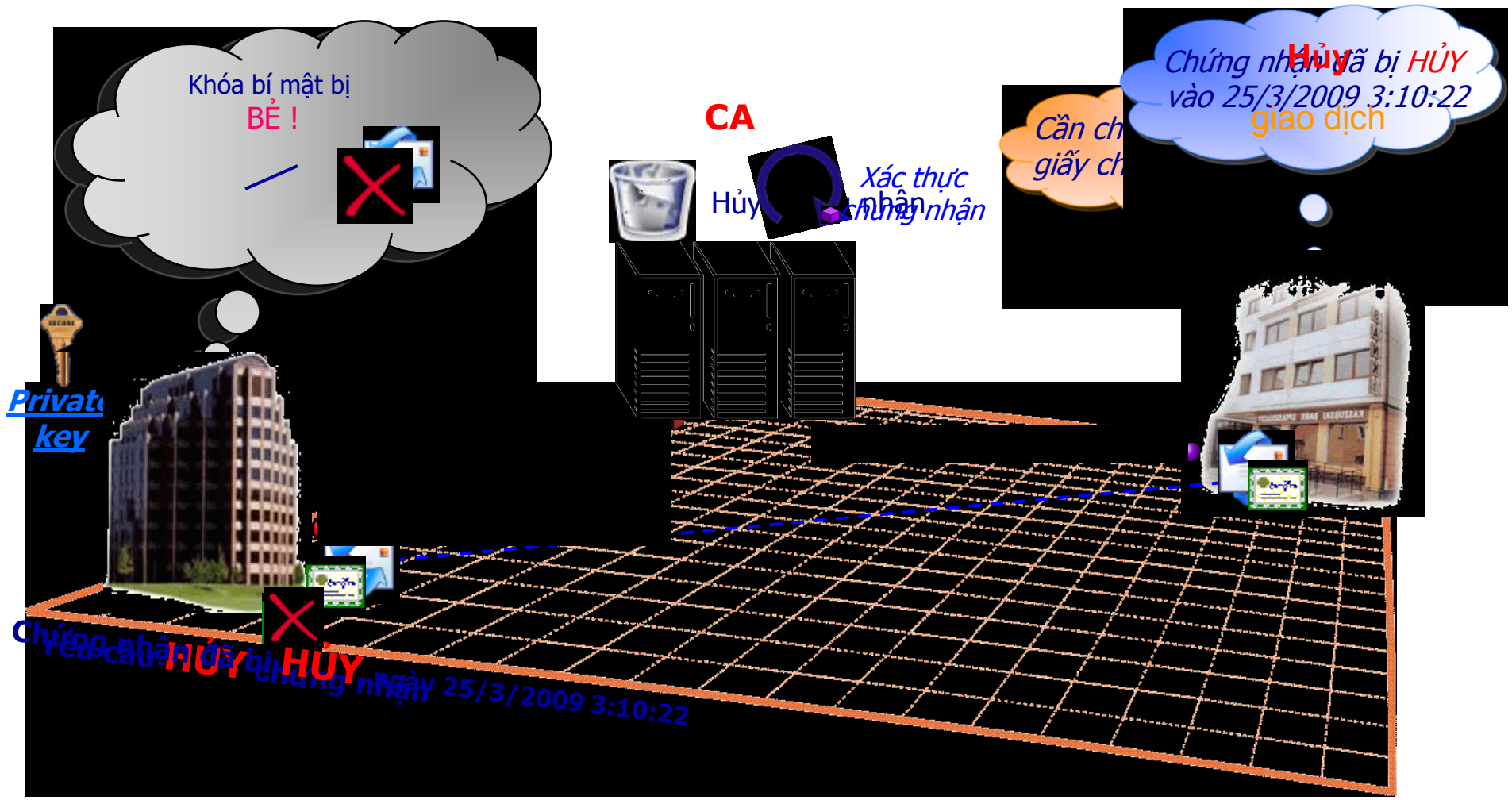


- thông tin về người phát hành chứng chỉ
- ngày kiểm tra tính hợp lệ
- chữ ký số bởi người phát hành chứng chỉ

Sử


Tổ chức
Tài liệu chữ ký
Xác nhận chữ ký
Chứng nhận hợp lệ & còn giá trị
Public key
Ok! Tin tưởng & chấp nhận đề nghị.
Bảng tin cậy ?
Tạo chứng nhận
Xác thực chứng nhận
Ký & Mã hóa
Private key
Thông tin

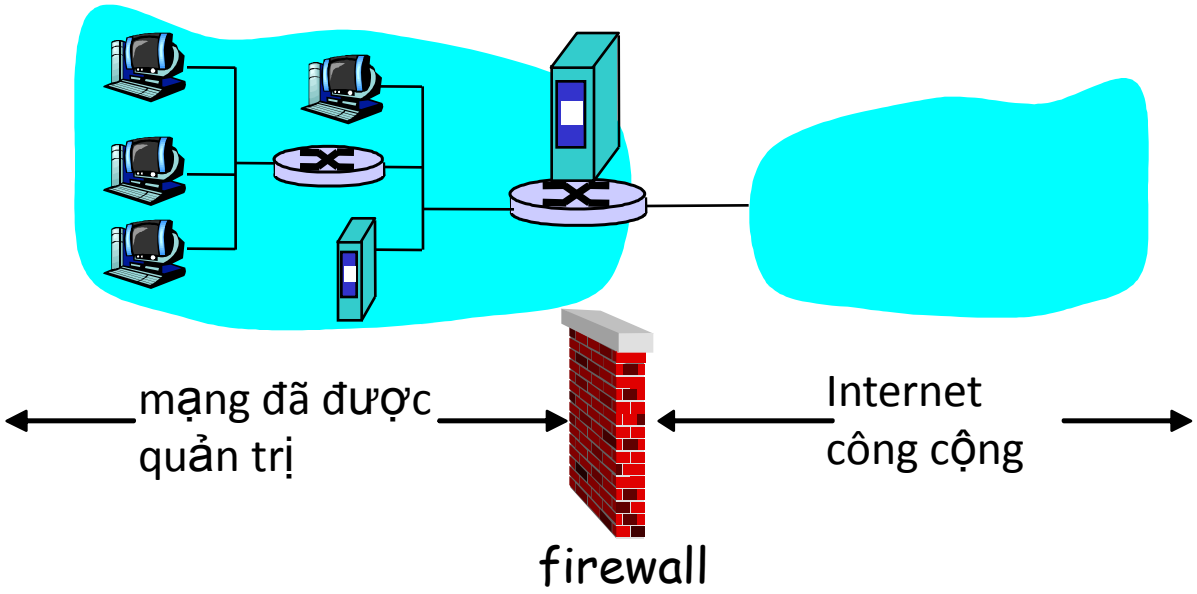
Sử dụng chứng chỉ



Các Firewall-Tường lửa

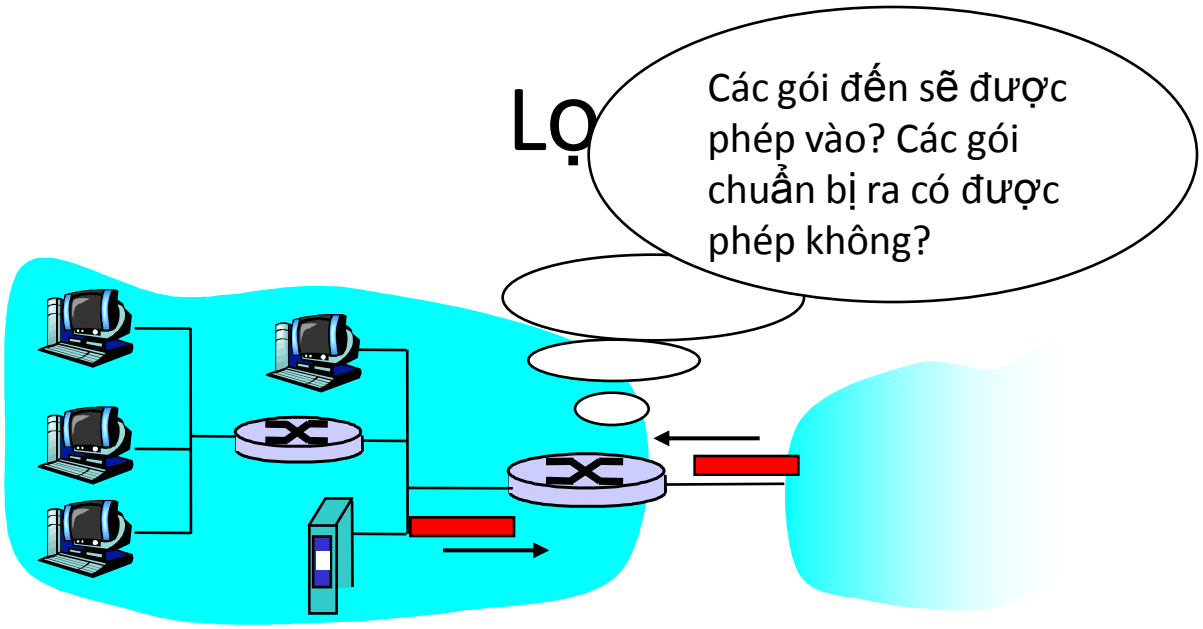
firewall

cô lập mạng nội bộ của tổ chức với Internet, cho phép một số gói được truyền qua, ngăn chặn các gói khác



Firewall: Tại sao phải dùng?

- Ngăn chặn các cuộc tấn công từ chối dịch vụ Denial Of Service (DoS):
  - SYN flooding: kẻ tấn công thiết lập nhiều kết nối TCP "ảo", không còn tài nguyên cho các kết nối "thật"
- Ngăn chặn việc sửa đổi/truy cập bất hợp pháp các dữ liệu nội bộ.
  - Ví dụ: kẻ tấn công thay thế trang chủ của CIA bằng trang nào đó
- Chỉ cho phép các truy cập hợp pháp vào bên trong mạng (tập hợp các host/user được chứng thực)
- 2 kiểu firewall:
  - mức ứng dụng
  - lọc gói tin

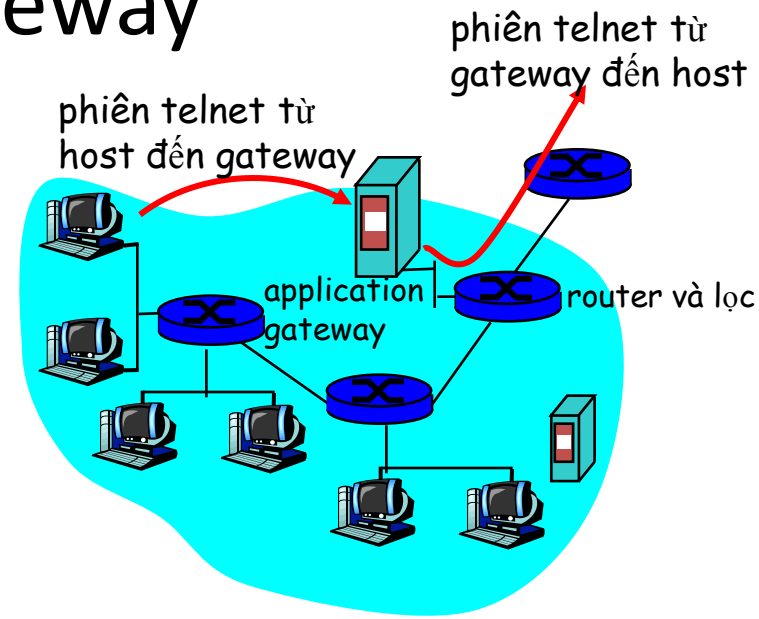


- mạng nội bộ kết nối với Internet thông qua router firewall
- router lọc từng gói một, xác định chuyển tiếp hoặc bỏ các gói dựa trên:
  - địa chỉ IP nguồn, địa chỉ IP đích
  - các số hiệu port TCP/UDP nguồn và đích
  - kiểu thông điệp ICMP
  - các bit TCP SYN và ACK

Lọc gói tin

- Ví dụ 1: chặn các datagram đến và đi với trường giao thức IP = 17 và port nguồn hoặc đích = 23.
  - Tất cả các dòng UDP đến/đi và các kết nối telnet đều bị chặn lại.
- Ví dụ 2: chặn các đoạn Block TCP với ACK=0.
  - Ngăn chặn các client bên ngoài tạo các kết nối TCP với các client bên trong, nhưng cho phép các client bên trong kết nối ra ngoài.

Các ứng dụng gateway



- Lọc các gói trên dữ liệu ứng dụng cũng như các trường IP/TCP/UDP.
- Ví dụ: cho phép chọn các user bên trong được telnet ra ngoài.

1. yêu cầu tất cả các user phải telnet thông qua gateway
2. với các user đã được cấp phép, gateway thiết lập kết nối với host đích. gateway tiếp vận dữ liệu giữa 2 kết nối.
3. Router lọc và chặn tất cả các kết nối telnet không xuất phát từ gateway.

Các hạn chế của các firewall và gateway

- giả mạo IP: router không thể biết dữ liệu có thực sự đến từ nguồn tin cậy hay không
- nếu nhiều ứng dụng cần đối xử đặc biệt, mỗi cái sở hữu gateway riêng…
- phần mềm client phải biết cách tiếp xúc với gateway.
  - ví dụ: phải thiết lập địa chỉ IP của proxy trong trình duyệt Web
- các lọc thường dùng tất cả hoặc không có chính sách nào dành cho UDP
- sự cân bằng: mức độ truyền thông với bên ngoài và sự an toàn
- nhiều site bảo vệ mức cao vẫn phải chịu đựng sự tấn công

Các loại tấn công và cách phòng chống

Phương thức:

- Trước khi tấn công: hacker tìm hiểu các dịch vụ đã hiện thực/hoạt động trên mạng
- Dùng `ping` để xác định các host nào có địa chỉ trên mạng
- Quét port: liên tục thử thiết lập các kết nối TCP với mỗi port (xem thử chuyện gì xảy ra)

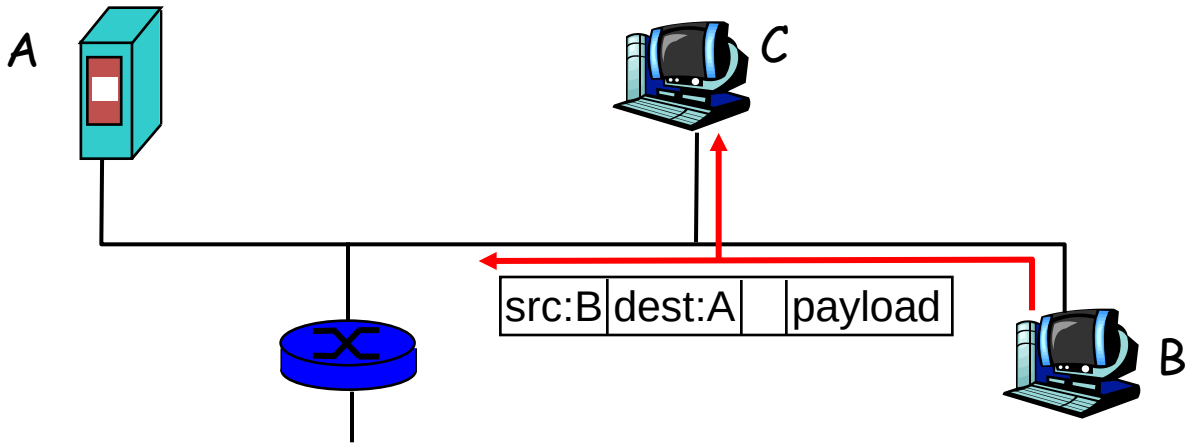
Biện pháp đối phó?

- Ghi nhận lưu thông vào mạng
- Quan tâm các hành vi nghi ngờ (các địa chỉ IP, port bị quét liên tục)

Các mối đe dọa bảo mật Internet

Packet sniffing: Nghe ngóng gói

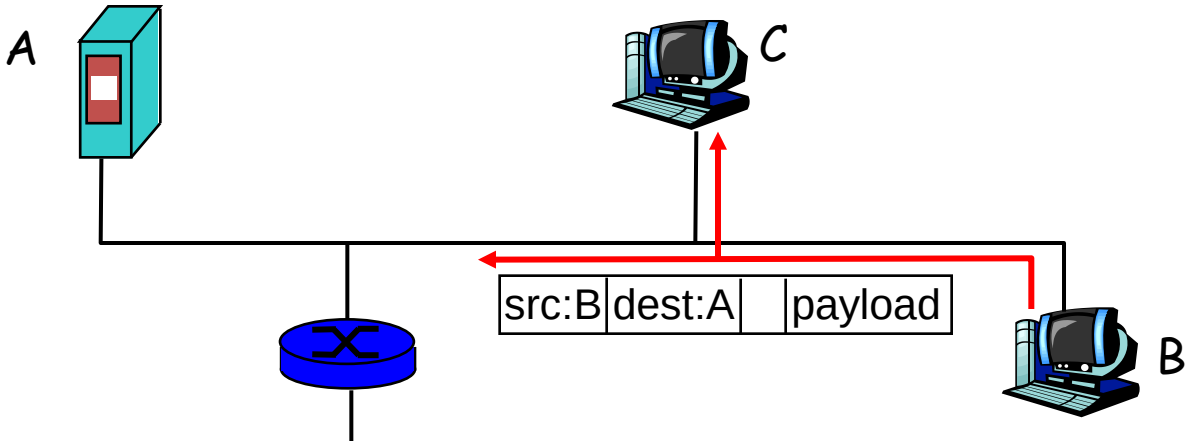
- NIC promiscuous (hỗn tạp) đọc tất cả các gói chuyển qua nó
- Có thể đọc tất cả các dữ liệu được mã hóa (như mật khẩu)
- Ví dụ: C nghe ngóng các gói của B



Các mối đe dọa bảo mật Internet

Packet sniffing: Biện pháp đối phó

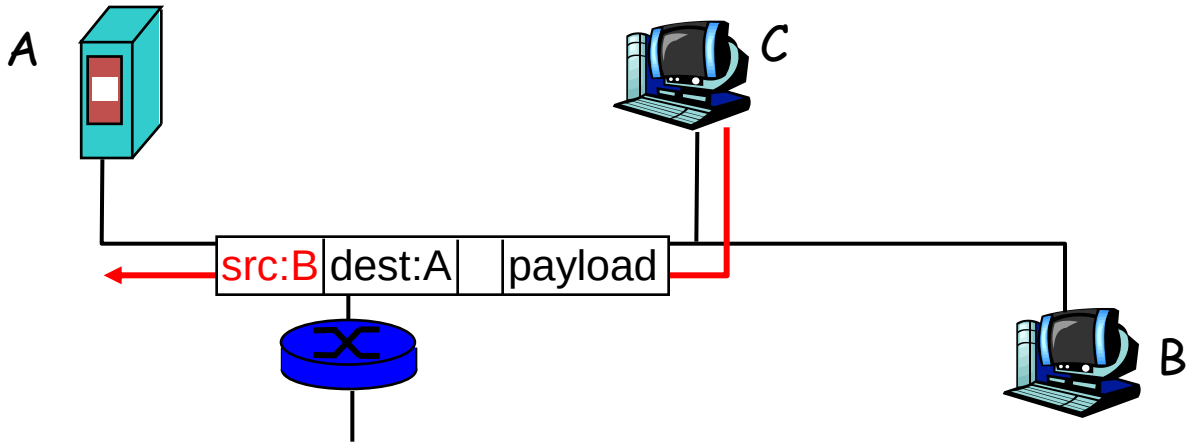
- Tất cả các host trong tổ chức chạy phần mềm kiểm tra định kỳ xem host có ở chế độ promiscuous
- 1 host mỗi đoạn của phương tiện truyền thông



Các mối đe dọa bảo mật Internet

IP Spoofing (giả mạo IP):

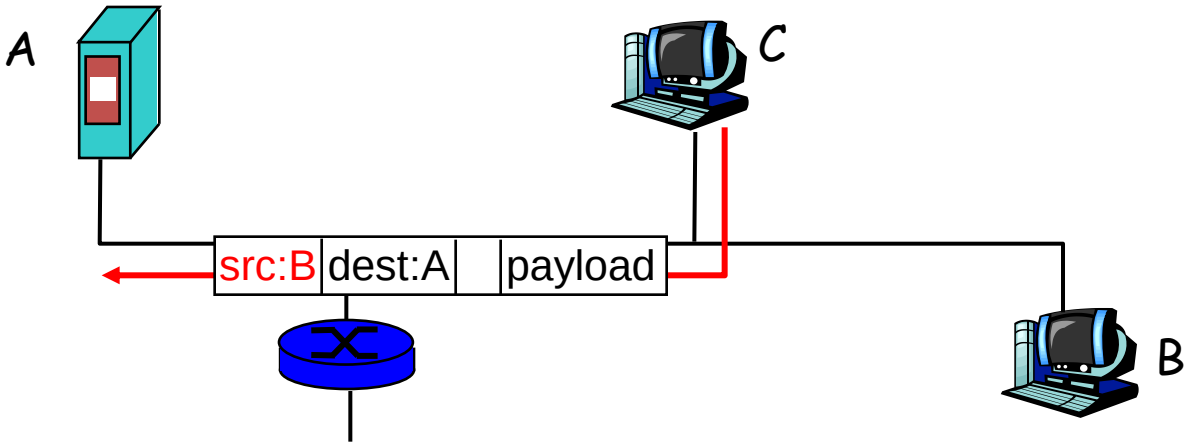
- Có thể sinh ra các gói IP "thô" trực tiếp từ ứng dụng, gán giá trị bất kỳ vào trường địa chỉ IP nguồn
- Bên nhận không thể xác định nguồn bị giả mạo
- Ví dụ: C giả mạo là B



Các mối đe dọa bảo mật Internet

IP Spoofing: lọc quyền vào

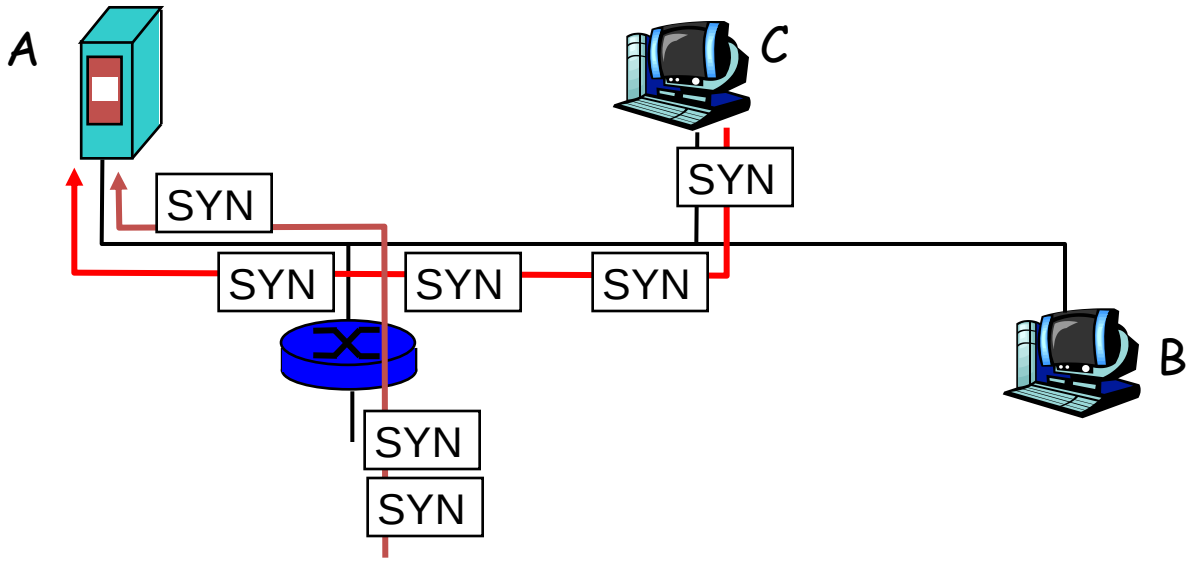
- Router sẽ không chuyển tiếp các gói đi với trường hợp các địa chỉ nguồn không hợp lệ
- Tuyệt vời, nhưng lọc như thế không thể áp dụng cho tất cả các mạng



Các mối đe dọa bảo mật Internet

Denial of Service (DoS):

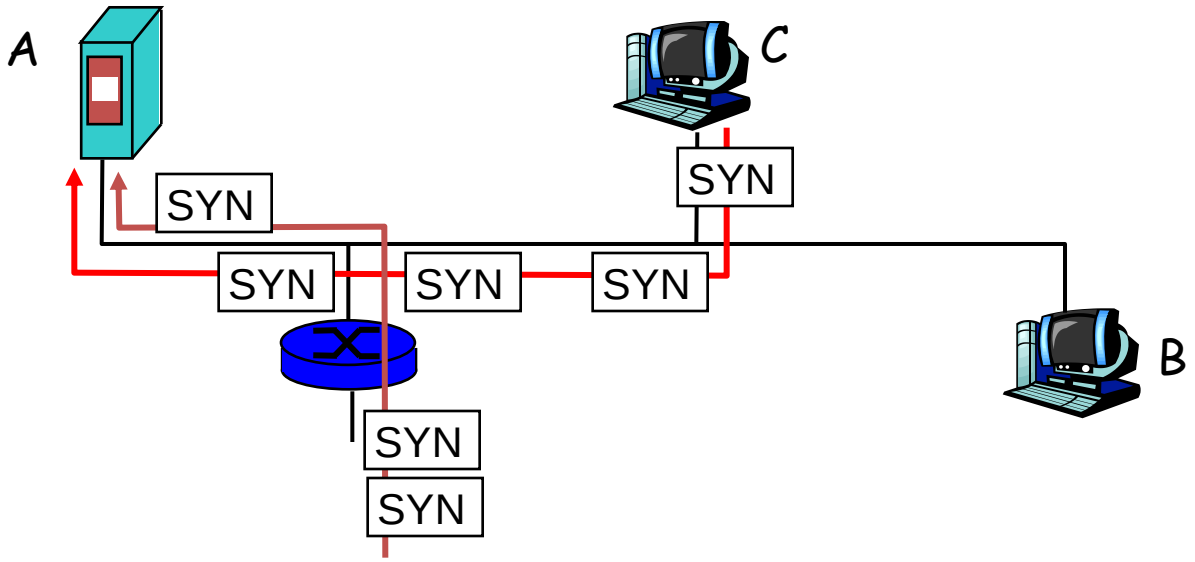
- Gây ra "ngập lụt" bằng các gói sinh ra bởi ý đồ xấu cho bên nhận
- Distributed DOS (DDoS): nhiều nguồn phối hợp làm "ngập lụt" bên nhận
- Ví dụ: C và các host ở xa tấn công SYN A



Các mối đe dọa bảo mật Internet

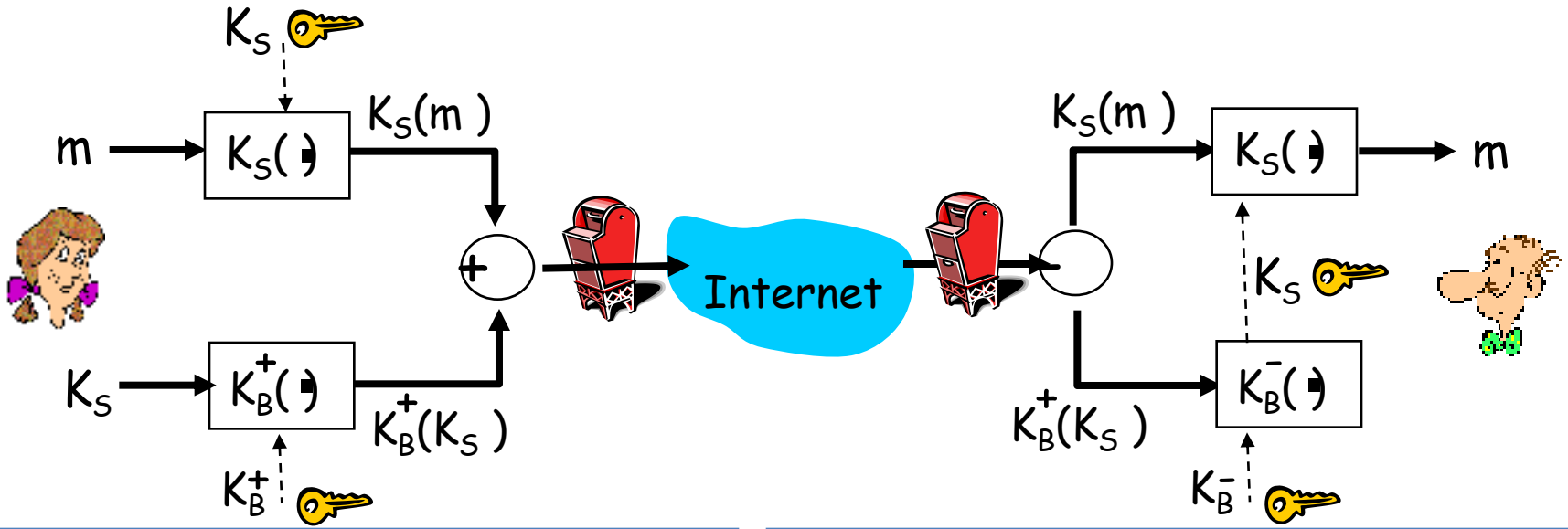
Denial of Service (DoS): Biện pháp đối phó?

- Lọc ra trước các gói dùng làm "ngập lụt" (ví dụ: SYN)
- Theo dõi ngược lại nguồn gây ra "ngập lụt" (cơ chế giống máy phát hiện nói dối của Mỹ)



Bảo mật e-mail

- Alice muốn gửi 1 e-mail bí mật, m, đến Bob.



Alice:

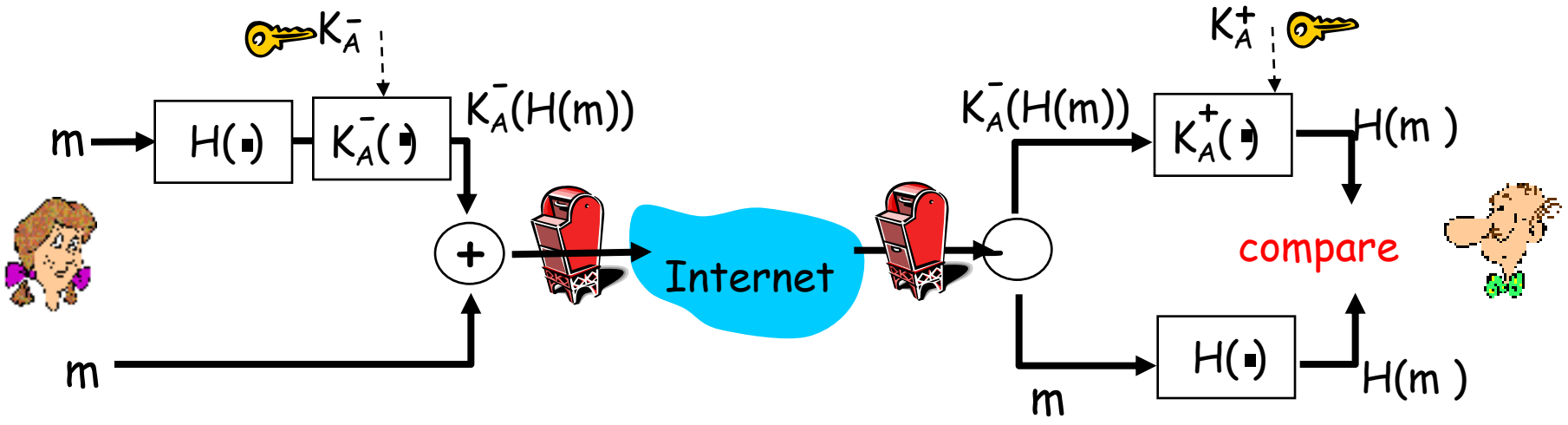
- sinh ra khóa riêng đối xứng ngẫu nhiên, K_S.
- mã hóa thông điệp với K_S
- cũng mã hóa K_S với khóa công cộng của Bob.
- gửi cả $K_S(m)$ và $K_B(K_S)$ cho Bob.

Bob:

- dùng khóa riêng của anh ấy để giải mã và phục hồi K_S
- dùng K_S để giải mã $K_S(m)$ và phục hồi m

Bảo mật e-mail

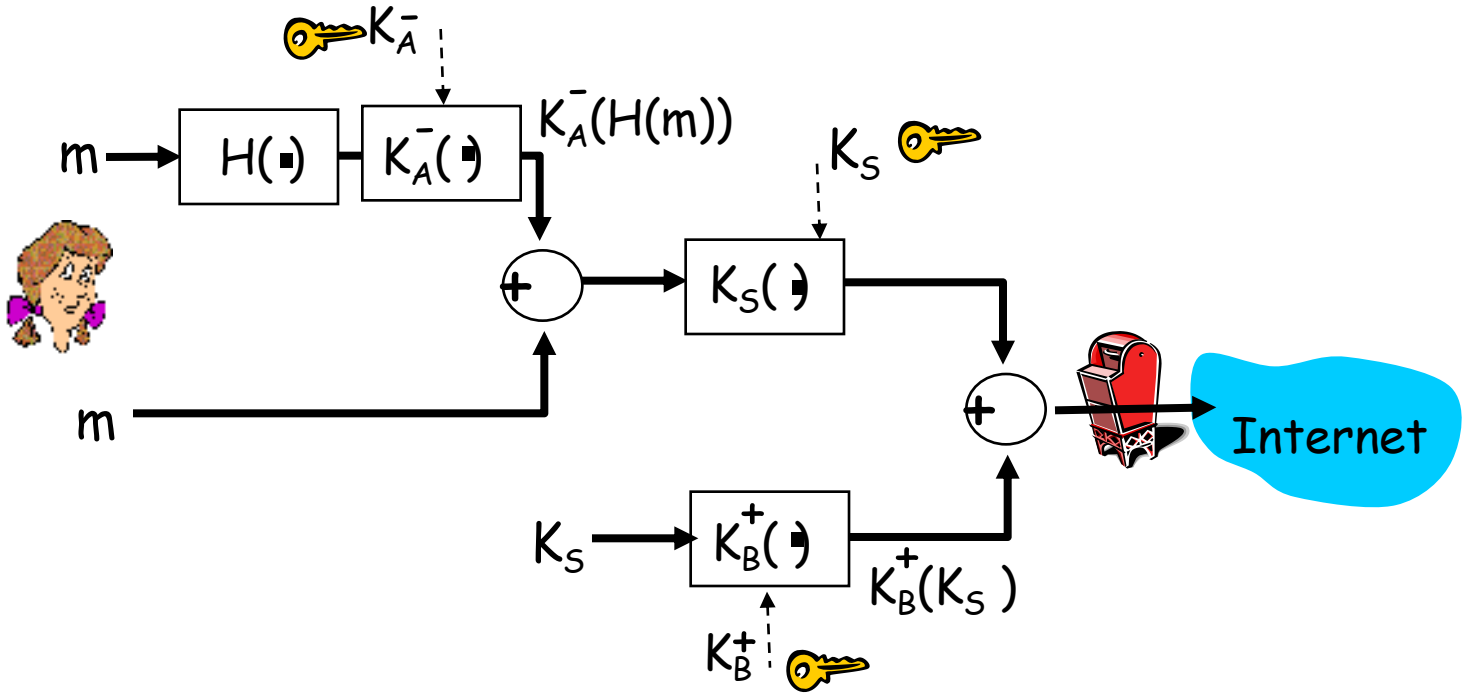
- Alice muốn cung cấp sự toàn vẹn thông điệp chứng thực người gửi.



- Alice ký số trên thông điệp.
- gửi cả thông điệp (dạng rõ ràng) và chữ ký số.

Bảo mật e-mail

- Alice muốn cung cấp sự toàn vẹn thông điệp chứng thực người gửi → sự bí mật



Alice dùng 3 khóa: khóa riêng của cô ấy, khóa công cộng của Bob, khóa đối xứng vừa mới tạo

Pretty good privacy (PGP)

- Chuẩn trên thực tế để mã hóa email Internet.
- Dùng mã hóa khóa đối xứng, khóa công cộng, hàm băm và chữ ký số như đã trình bày ở trước.
- Hỗ trợ đồng nhất, chứng thực người gửi, bí mật
- Người phát minh: Phil Zimmerman.

Một thông điệp đã được ký bằng PGP

```
---BEGIN PGP SIGNED MESSAGE---
Hash: SHA1

Bob:My husband is out of town
   tonight.Passionately yours, A

---BEGIN PGP SIGNATURE---
Version: PGP 5.0
Charset: noconv
yhHJRHhGJGhgg/12EpJ+lo8gE4vB3mqJ
   hFEvZP9t6n7G6m5Gw2
---END PGP SIGNATURE---
```

Secure sockets layer (SSL)

- Bảo mật lớp transport với bất kỳ ứng dụng nào dựa trên TCP dùng các dịch vụ SSL
- Dùng giữa trình duyệt Web, các server trong thương mại điện tử
- Các dịch vụ bảo mật:
  - Chứng thực server
  - Mã hóa dữ liệu
  - Chứng thực client (tùy chọn)

- Chứng thực server:
  - Trình duyệt cho phép SSL chứa các khóa công cộng cho các CA được tin cậy
  - Trình duyệt yêu cầu chứng chỉ server, phát ra bởi CA được tin cậy
  - Trình duyệt dùng khóa công cộng của CA để trích ra khóa công cộng của server từ chứng chỉ
- Kiểm tra trong trình duyệt của bạn để thấy các CA được tin cậy

SSL (tt)

Mã hóa phiên làm việc SSL :

- Trình duyệt sinh ra *khóa phiên đối xứng*, mã hóa nó với khóa công cộng của server, gửi khóa (đã mã hóa) cho server.
- Dùng khóa riêng, server giải mã khóa phiên
- Trình duyệt, server biết khóa phiên
  - Tất cả dữ liệu gửi vào trong TCP socket (do client hoặc server) được mã hóa bởi khóa phiên.

- SSL: cơ sở của IETF Transport Layer Security (TLS).
- SSL có thể dùng cho các ứng dụng không Web, như IMAP.
- Chứng thực client có thể hoàn thành với các chứng chỉ client

IPSec: bảo mật lớp Network

- Bảo mật lớp Network:
  - host gửi mã hóa dữ liệu trong IP datagram
  - các đoạn TCP & UDP; các thông điệp ICMP & SNMP.
- Chứng thực lớp Network:
  - host đích có thể chứng thực địa chỉ IP nguồn
- 2 giao thức cơ bản:
  - authentication header (AH)
  - encapsulation security payload (ESP)
- Với cả AH và ESP, nguồn – đích bắt tay nhau:
  - tạo kênh logic lớp network gọi là một security association (SA)
- Mỗi SA theo 1 chiều duy nhất
- duy nhất xác định bởi:
  - giao thức bảo mật (AH hoặc ESP)
  - địa chỉ IP nguồn
  - ID của kết nối 32-bit

Giao thức AH

- Hỗ trợ chứng thực nguồn, toàn vẹn dữ liệu, không tin cậy
- AH header được chèn vào giữa IP header, trường dữ liệu.
- Trường giao thức: 51
- Trung gian xử lý các datagram như bình thường

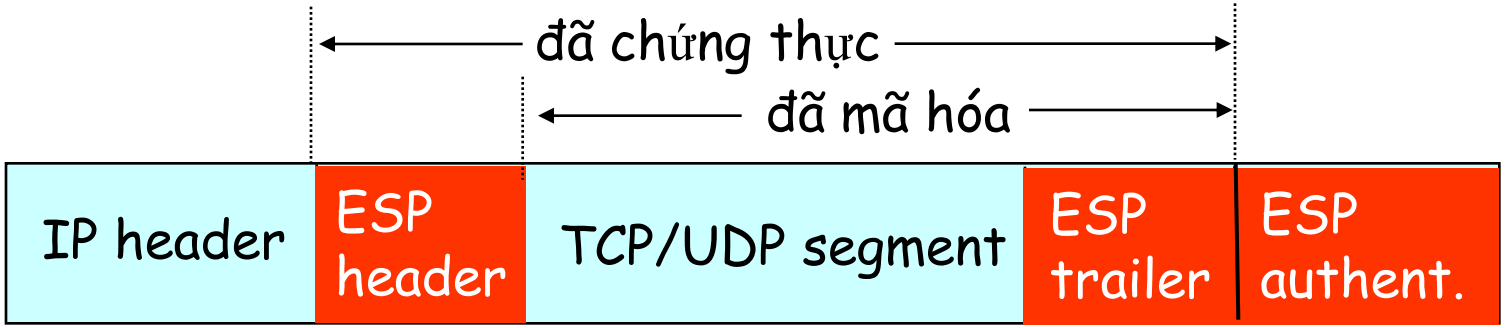
AH header chứa:

- Nhận dạng kết nối
- Dữ liệu chứng thực: thông điệp đã được ký từ nguồn được tính toán dựa trên IP datagram gốc
- Trường header kế tiếp: xác định kiểu của dữ liệu (vd: TCP, UDP, ICMP)

Giao thức ESP

- Hỗ trợ toàn vẹn dữ liệu, chứng thực host, tính bí mật
- Mã hóa dữ liệu, ESP trailer
- Trường header kế tiếp nằm trong ESP trailer.

- Trường chứng thực ESP tương tự như của AH
- Protocol = 50.



Bảo mật IEEE 802.11

- *Khảo sát:*
  - 85% việc sử dụng mà không có mã hóa/chứng thực
  - Dễ dàng bị phát hiện/nghe ngóng và nhiều loại tấn công khác!
- Bảo mật 802.11
  - Mã hóa, chứng thực
  - Thử nghiệm bảo mật 802.11 đầu tiên là Wired Equivalent Privacy (WEP): có thiếu sót
  - Thử nghiệm hiện tại: 802.11i

Wired Equivalent Privacy (WEP):

- Chứng thực như trong giao thức *ap4.0*
  - host yêu cầu chứng thực từ access point
  - access point gửi 128 bit
  - host mã hóa dùng khóa đối xứng chia sẻ
  - access point giải mã, chứng thực host
- Không có cơ chế phân bố khóa
- Chứng thực: chỉ cần biết khóa chia sẻ

Wi-Fi Protected Access (WPA)

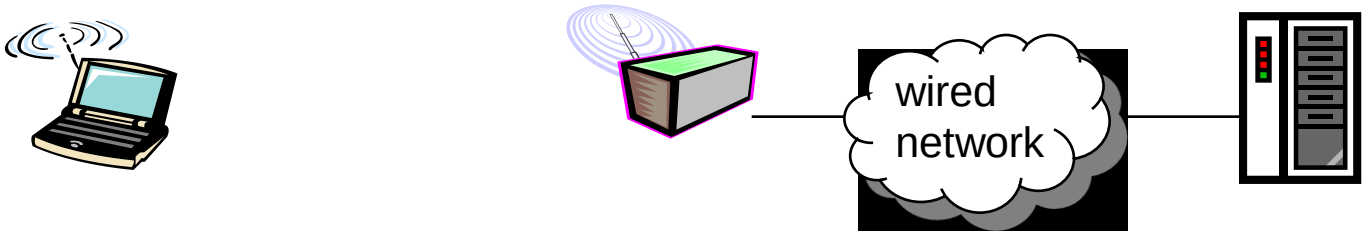
- Hai sự cải tiến chính so với WEP:
  - Mã hóa dữ liệu cải tiến thông qua giao thức Temporal Key Integrity Protocol (TKIP). TKIP scrambles key sử dụng thuật toán hashing và bằng đặc tính kiểm tra số nguyên, đảm bảo rằng Key sẽ không bị giả mạo.
  - Chứng thực người dùng, thông qua EAP.
- WPA là tiêu chuẩn tạm thời mà sẽ được thay thế với chuẩn IEEE 802.11i

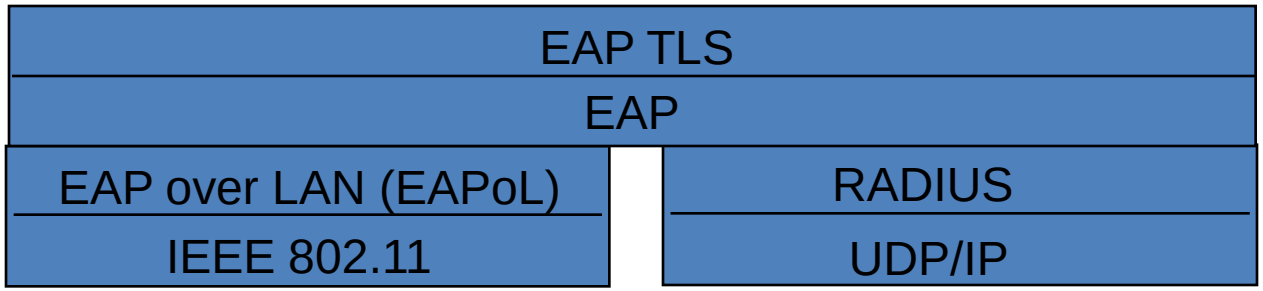
802.11i: cải tiến sự bảo mật

- Rất nhiều (và chắc chắn hơn) dạng mã hóa có thể
- Hỗ trợ phân bố khóa
- Dùng chứng thực server tách riêng khỏi AP

EAP: Extensible Authentication Protocol

- EAP được gửi trên các "link" riêng biệt
  - mobile-đến-AP (EAP trên LAN)
  - AP đến server chứng thực (RADIUS trên UDP)





TÀI LIỆU THAM KHẢO, ĐỊA CHỈ LIÊN LẠC

- Giáo trình Mạng máy tính, KS. Nguyễn Bình Dương, TS. Đàm Quang Hồng Hải
- Giáo trình hệ thống Mạng máy tính CCNA, Nguyễn Hồng Sơn
- CCNA: Cisco Certified Network Associate – Study Guide, Todde Lammle - 2007
- Computer Networking: A Top Down Approach Featuring the Internet, 3rd edition. Jim Kurose, Keith Ross. 2004.
- Computer Networks, 4th edition. Andrew S. Tanenbaum. 2003

- Địa chỉ liên lạc: Trần Bá Nhiệm – Khoa Mạng máy tính & Truyền thông – ĐH CNTT – 34 Trương Định, Q3, Tp.HCM. Email: tranbanhiem@yahoo.com

Giáo trình nhập môn mạng máy tính

MỤC LỤC

Chương I Những khái niệm cơ bản của mạng máy tính ...6

I. Định nghĩa mạng máy tính 6

II. Phân loại mạng máy tính 7

II.1. Dựa theo vị trí địa lý 7

II.2. Dựa theo cấu trúc mạng 7

II.2.1 Kiểu điểm - điểm (point - to - point) 7

II.2.2 Kiểu khuyếch tán 8

II.3. Dựa theo phương pháp chuyển mạch 8

II.3.1 Mạng chuyển mạch kênh (Line switching network) 8

II.3.2 Mạng chuyển mạch thông điệp (Message switching network) 8

II.3.3 Mạng chuyển mạch gói (Packet switching network) 9

III. So sánh giữa mạng cục bộ và mạng diện rộng 9

IV. Các thành phần của mạng máy tính 11

IV.1. Một số bộ giao thức kết nối mạng 11

IV.2. Hệ điều hành mạng - NOS (*Network Operating System)* 11

V. Các lợi ích của mạng máy tính 12

V.1. Mạng tạo khả năng dùng chung tài nguyên cho các người dùng. 12

V.2. Mạng cho phép nâng cao độ tin cậy. 13

V.3. Mạng giúp cho công việc đạt hiệu suất cao hơn. 13

V.4. Tiết kiệm chi phí. 13

V.5. Tăng cường tính bảo mật thông tin. 13

V.6. Việc phát triển mạng máy tính đã tạo ra nhiều ứng dụng mới 13

VI. Các dịch vụ phổ biến trên mạng máy tính 13

Chương II Mô hình truyền thông 15

I. Sự cần thiết phải có mô hình truyền thông 15

II. Các nhu cầu về chuẩn hóa đối với mạng 16

III. Mô hình OSI (Open Systems Interconnection) 17

III.1. Nguyên tắc sử dụng khi định nghĩa các tầng hệ thống mở 17

III.2. Các giao thức trong mô hình OSI 18

III.3. Các chức năng chủ yếu của các tầng của mô hình OSI. 19

III.3.1 Tầng 1: Vật lý (Physical) 19

III.3.2 Tầng 2: Liên kết dữ liệu (Data link) 20

III.3.3 Tầng 3: Mạng (Network) 20
III.3.4 Tầng 4: Vận chuyển (Transport) 22
III.3.5 Tầng 5: Giao dịch (Session) 23
III.3.6 Tầng 6: Trình bày (Presentation) 23
III.3.7 Tầng 7: Ứng dụng (Application) 24
IV. Quá trình chuyển vận gói tin 24
IV.1. Quá trình đóng gói dữ liệu (tại máy gửi) 24
IV.2. Quá trình truyền dữ liệu từ máy gửi đến máy nhận. 26
IV.3. Chi tiết quá trình xử lý tại máy nhận 26
V. Phương thức truyền tín hiệu 27
VI. Mô hình TCP/IP 27
VI.1. Tổng quan về bộ giao thức TCP/IP 27
VI.2. So sánh TCP/IP với OSI 29
VII. Các giao thức truy cập đường truyền trên mạng LAN 30
VII.1. Giao thức chuyển mạch (yêu cầu và chấp nhận) 30
VII.2. Giao thức đường dây đa truy cập với cảm nhận va chạm 30
VII.3. Giao thức dùng thẻ bài vòng (Token ring) 31
VII.4. Giao thức dung thẻ bài cho dạng đường thẳng (Token bus) 31
VIII. Các phương tiện kết nối mạng liên khu vực (WAN) 31
Chương III Địa chỉ IP 33
I. Giao thức TCP/IP 33
II. Địa chỉ IP 33
II.1. Tổng quát 33
II.2. Cấu trúc của các địa chỉ IP 33
III. Một số khái niệm và thuật ngữ liên quan 37
III.1. Các giao thức trong mạng IP 38
III.2. Các bước hoạt động của giao thức IP 38
IV. Giao thức điều khiển truyền dữ liệu TCP 39
IV.1. Các bước thực hiện để thiết lập một liên kết TCP/IP: 40
IV.2. Các bước thực hiện khi truyền và nhận dữ liệu 41
IV.3. Các bước thực hiện khi đóng một liên kết 42
IV.4. Một số hàm khác của TCP 42
V. Giao thức UDP (User Datagram Protocol) 44
VI. Địa chỉ IPv4 46

VI.1. Thành phần và hình dạng của địa chỉ IP 46
VI.2. Các lớp địa chỉ IP 46
VII. IPv6 48
VII.1. Giao thức liên mạng thế hệ mới (IPv6) 48
VII.2. Một số đặc điểm mới của IPv6: 48
VII.3. Kiến trúc địa chỉ trong IPv6: 49
VII.3.1 Không gian địa chỉ: 49
VII.3.2 Cú pháp địa chỉ: 50
Chương IV Thiết bị mạng 51
I. Môi trường truyền dẫn 51
I.1. Khái niệm 51
I.2. Tần số truyền thông 51
I.3. Các đặc tính của phương tiện truyền dẫn 51
I.4. Các kiểu truyền dẫn. 52
II. Đường cáp truyền mạng 52
II.1. Cáp xoắn cặp 52
II.2. Cáp đồng trục 53
II.3. Cáp sợi quang (Fiber - Optic Cable) 54
II.4. Các yêu cầu cho một hệ thống cáp 54
III. Đường truyền vô tuyến 55
III.1. Sóng vô tuyến (radio) 55
III.2. Sóng viba 55
III.3. Hồng ngoại 55
IV. Các kỹ thuật bấm cáp mạng 56
V. Các thiết bị liên kết mạng 57
V.1. Repeater (Bộ tiếp sức) 57
V.2. Bridge (Cầu nối) 58
V.3. Router (Bộ tìm đường) 61
V.3.1 Các phương thức hoạt động của Router 64
V.3.2 Một số giao thức hoạt động chính của Router 64
V.4. Gateway (cổng nối) 64
V.5. Hub (Bộ tập trung) 65
V.6. Bộ chuyển mạch (*switch*) 66
Chương V Mô hình mạng 67

I. Kiến trúc mạng (Topology) 67
II. Những cấu trúc chính của mạng cục bộ 67
II.1. Dạng đường thẳng (Bus) 67
II.2. Dạng vòng tròn (Ring) 68
II.3. Dạng hình sao (Star) 68
II.4. Mạng dạng kết hợp 70
Chương VI Các dịch vụ của mạng diện rộng (WAN) 71
I. Mạng chuyển mạch (Circuit Swiching Network) 71
II. Mạng thuê bao (Leased line Network) 73
III. Mạng chuyển gói tin (Packet Switching NetWork) 74
IV. Mạng X25 75
V. Mạng Frame Relay 76
VI. Mạng ATM (Cell relay) 76
Chương VII CÁC DỊCH VỤ MẠNG THÔNG DỤNG 78
I. DỊCH VỤ WEB 78
I.1. Một số thuật ngữ cơ bản. 78
I.2. Giới thiệu mô hình hoạt động của Web. 80
II. DỊCH VỤ FTP 81
II.1. Mô hình hoạt động của FTP 81
II.2. Tập hợp các lệnh FTP 81
III. E-MAIL. 83
III.1. Mô hình hoạt động 83
III.2. Các loại mail. 83
III.3. Sử dụng WebMail. 83

Chương I Những khái niệm cơ bản của mạng máy tính

Mạng máy tính ngày nay đã phát triển một cách nhanh chóng và đa dạng. Hệ điều hành cùng các ứng dụng của mạng ngày càng phong phú, các lợi ích của mạng ngày càng được khẳng định. Mạng máy tính bao gồm rất nhiều loại, nhiều mô hình triển khai. Trong một mạng máy tính lại có nhiều thành phần cấu thành. Trước khi đi chi tiết về mạng máy tính, chúng ta sẽ tìm hiểu các khái niệm cơ bản của mạng máy tính.

I. Định nghĩa mạng máy tính

Mạng máy tính là một tập hợp các máy tính được nối với nhau bởi đường truyền theo một cấu trúc nào đó và thông qua đó các máy tính trao đổi thông tin qua lại với nhau.

Đường truyền là hệ thống các thiết bị truyền dẫn có dây hay không dây dùng để chuyển các tín hiệu điện tử từ máy tính này đến máy tính khác. Các tín hiệu điện tử đó biểu thị các giá trị dữ liệu dưới dạng các xung nhị phân (on - off). Tất cả các tín hiệu được truyền giữa các máy tính đều thuộc một dạng sóng điện từ. Tùy theo tần số của sóng điện từ có thể dùng các đường truyền vật lý khác nhau để truyền các tín hiệu. Ở đây đường truyền được kết nối có thể là dây cáp đồng trục, cáp xoắn, cáp quang, dây điện thoại, sóng vô tuyến ... Các đường truyền dữ liệu tạo nên cấu trúc của mạng.

Hình I-1 Mạng máy tính

Với sự trao đổi qua lại giữa máy tính này với máy tính khác đã phân biệt mạng máy tính với các hệ thống thu phát một chiều như truyền hình, phát thông tin từ vệ tinh xuống các trạm thu thụ động... vì tại đây chỉ có thông tin một chiều từ nơi phát đến nơi thu mà không quan tâm đến có bao nhiêu nơi thu, có thu tốt hay không.

Đặc trưng cơ bản của đường truyền vật lý là giải thông. Giải thông của một đường chuyền chính là độ đo phạm vi tần số mà nó có thể đáp ứng được. Tốc độ truyền dữ liệu trên đường truyền còn được gọi là thông lượng của đường truyền - thường được tính bằng số lượng bit được truyền đi trong một giây (Bps). Thông lượng còn được đo bằng đơn vị khác là Baud (lấy từ tên nhà bác học - Emile Baudot). Baud biểu thị số lượng thay đổi tín hiệu trong một giây.

Ở đây Baud và Bps không phải bao giờ cũng đồng nhất. Ví dụ: nếu trên đường dây có 8 mức tín hiệu khác nhau thì mỗi mức tín hiệu tương ứng với 3 bit hay là 1 Baud tương ứng với 3 bit. Chỉ khi có 2 mức tín hiệu trong đó mỗi mức tín hiệu tương ứng với 1 bit thì 1 Baud mới tương ứng với 1 bit.

II. Phân loại mạng máy tính

Có nhiều cách để phân biệt mạng máy tính nhưng người ta thường phân biệt mạng máy tính theo vị trí địa lý, cấu trúc mạng, phương pháp chuyển mạch.

II.1. Dựa theo vị trí địa lý

Dựa vào phạm vi phân bổ của mạng người ta có thể phân ra các loại mạng như sau:

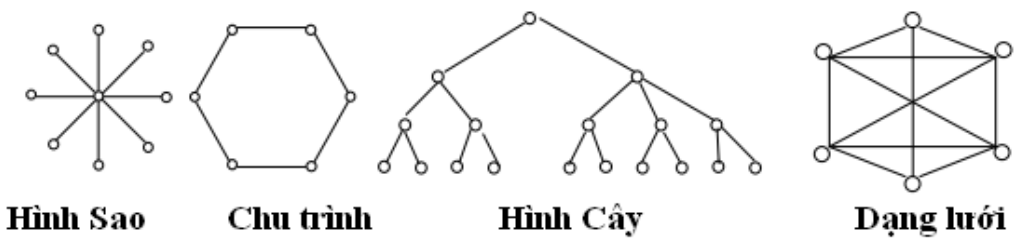
- **GAN (Global Area Network)** - Kết nối máy tính từ các châu lục khác nhau. Thông thường kết nối này được thực hiện thông qua mạng viễn thông và vệ tinh.
- **WAN (Wide Area Network)** - Mạng diện rộng, kết nối máy tính trong nội bộ các quốc gia hay giữa các quốc gia trong cùng một châu lục. Thông thường kết nối này được thực hiện thông qua mạng viễn thông. Các WAN có thể được kết nối với nhau thành GAN hay tự nó đã là GAN.
- **MAN (Metropolitan Area Network)** - Kết nối các máy tính trong phạm vi một thành phố hay giữa các thành phố với nhau.
- **LAN (Local Area Network)** - Mạng cục bộ, kết nối các máy tính trong một khu vực bán kính hẹp thông thường khoảng vài trăm mét. Kết nối được thực hiện thông qua các môi trường truyền thông tốc độ cao ví dụ cáp đồng trục thay cáp quang. LAN thường được sử dụng trong nội bộ một cơ quan/tổ chức... Các LAN có thể được kết nối với nhau thành WAN.

Trong các khái niệm nói trên, thường được sử dụng nhất hiện này là khái niệm Mạng diện rộng WAN và mạng cục bộ LAN.

II.2. Dựa theo cấu trúc mạng

II.2.1 Kiểu điểm - điểm (point - to - point)

Đường truyền nối từng cặp nút mạng với nhau. Thông tin đi từ nút nguồn qua nút trung gian rồi gởi tiếp nếu đường truyền không bị bận. Do đó, còn có tên là mạng lưu trữ và chuyển tiếp (store and forward).



Hình I-2 Cấu trúc điểm – điểm

II.2.2 Kiểu khuyếch tán

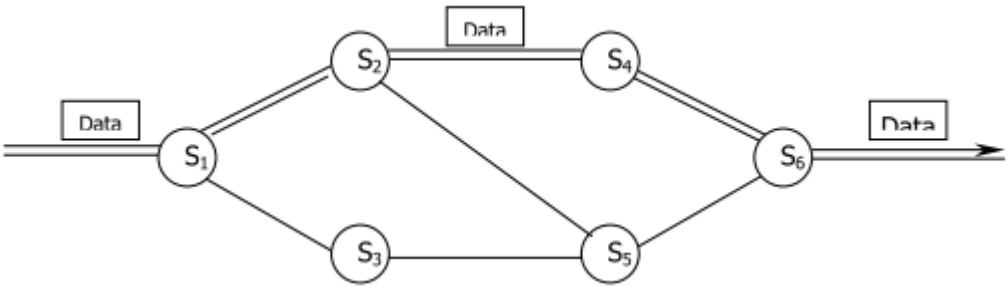
Bản tin được gởi đi từ một nút sẽ được tiếp nhận bởi các nút còn lại (còn gọi là broadcasting hay point to multipoint). Trong bản tin phải có vùng địa chỉ cho phép mỗi nút kiểm xem có phải tin của mình không và xử lý nếu đúng bản tin được gởi đến.

Hình I-3 Cấu trúc kiểu khuyếch tán

II.3. Dựa theo phương pháp chuyển mạch

II.3.1 Mạng chuyển mạch kênh (Line switching network)

Chuyển mạch kênh dùng trong mạng điện thoại. Một kênh cố định được thiết lập giữa cặp thực thể cần liên lạc với nhau. Mạng này có hiệu suất không cao vì có lúc kênh bỏ không.

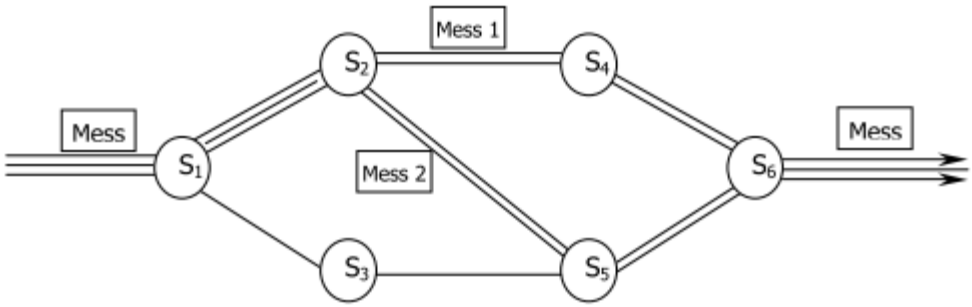


Hình I-4 Mạng chuyển mạch kênh

II.3.2 Mạng chuyển mạch thông điệp (Message switching network)

Các nút của mạng căn cứ vào địa chỉ đích của "thông điệp" để chọn nút kế tiếp. Như vậy các nút cần lưu trữ và đọc tin nhận được, quản lý việc truyền tin. Trong trường hợp bản tin quá dài và nếu sai phải truyền lại. Phương pháp này giống như cách gởi thư thông thường.

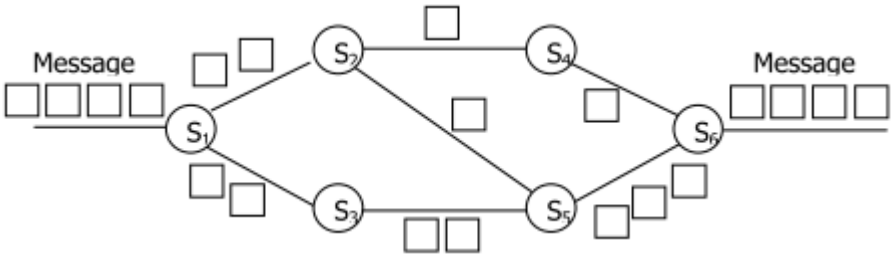
Mạng chuyển mạch thông báo thích hợp với các dịch vụ thông tin kiểu thư điện tử (Email) hơn là đối với các ứng dụng có tính thời gian thực vì tồn tại độ trễ nhất định do lưu trữ và xử lý thông tin điều khiển tại mỗi nút.



Hình I-5 Mạng chuyển mạch thông điệp

II.3.3 Mạng chuyển mạch gói (Packet switching network)

Bản tin được chia thành nhiều gói tin (packet) có độ dài 512 bytes, phần đầu của gói tin thường là địa chỉ đích, mã để tập hợp các gói. Các gói tin của các thông điệp khác nhau có thể được truyền độc lập trên cùng một đường truyền. Vấn đề phức tạp ở đây là tạo lại bản tin ban đầu, đặc biệt là khi truyền trên các con đường khác nhau. Chuyển mạch gói mềm dẻo, hiệu suất cao. Sử dụng hai kĩ thuật chuyển mạch kênh và chuyển mạch gói trong cùng một mạng thống nhất gọi là mạng ISDN (Integrated Services Digital Network – Mạng thông tin số đa dịch vụ)



Hình I-6 Mạng chuyển mạch gói

III. So sánh giữa mạng cục bộ và mạng diện rộng

Mạng cục bộ và mạng diện rộng có thể được phân biệt bởi: địa phương hoạt động, tốc độ đường truyền và tỷ lệ lỗi trên đường truyền, chủ quản của mạng, đường đi của thông tin trên mạng, dạng chuyển giao thông tin.

Địa phương hoạt động

Liên quan đến khu vực địa lý thì mạng cục bộ sẽ là mạng liên kết các máy tính nằm ở trong một khu vực nhỏ. Khu vực có thể bao gồm một tòa nhà hay là một khu nhà... Điều đó hạn chế bởi khoảng cách đường dây cáp được dùng để liên kết các máy tính của mạng cục bộ (Hạn chế đó còn là hạn chế của khả năng kỹ thuật của đường truyền dữ liệu). Ngược lại mạng diện rộng là mạng có khả năng liên kết các máy tính trong một vùng rộng lớn như là một thành phố, một miền, một đất nước, mạng diện rộng được xây dựng để nối hai hoặc nhiều khu vực địa lý riêng biệt.

Tốc độ đường truyền và tỷ lệ lỗi trên đường truyền

Do các đường cáp của mạng cục bộ đươc xây dựng trong một khu vực nhỏ cho nên nó ít bị ảnh hưởng bởi tác động của thiên nhiên (như là sấm chớp, ánh sáng...). Điều đó cho phép mạng cục bộ có thể truyền dữ liệu với tốc độ cao mà chỉ chịu một tỷ lệ lỗi nhỏ. Ngược lại với mạng diện rộng do phải truyền ở những khoảng cách khá xa với những đường truyền dẫn dài có khi lên tới hàng ngàn km. Do vậy mạng diện rộng không thể truyền với tốc độ quá cao vì khi đó tỉ lệ lỗi sẽ trở nên khó chấp nhận được.

Mạng cục bộ thường có tốc độ truyền dữ liệu từ 4 đến 16 Mbps và đạt tới 100 Mbps nếu dùng cáp quang. Còn phần lớn các mạng diện rộng cung cấp đường truyền có tốc độ thấp hơn nhiều như T1 với 1.544 Mbps hay E1 với 2.048 Mbps.

Đơn vị bps (Bit Per Second) là một đơn vị trong truyền thông tương đương với 1 bit được truyền trong một giây, ví dụ như tốc độ đường truyền là 1 Mbps tức là có thể truyền tối đa 1 Megabit trong 1 giây trên đường truyền đó.

Thông thường trong mạng cục bộ tỷ lệ lỗi trong truyền dữ liệu vào khoảng 1/107-108 còn trong mạng diện rộng thì tỷ lệ đó vào khoảng 1/106 - 107

Chủ quản và điều hành của mạng

Do sự phức tạp trong việc xây dựng, quản lý, duy trì các đường truyền dẫn nên khi xây dựng mạng diện rộng người ta thường sử dụng các đường truyền được thuê từ các công ty viễn thông hay các nhà cung cấp dịch vụ truyền số liệu. Tùy theo cấu trúc của mạng những đường truyền đó thuộc cơ quan quản lý khác nhau như các nhà cung cấp đường truyền nội hạt, liên tỉnh, liên quốc gia. Các đường truyền đó phải tuân thủ các quy định của chính phủ các khu vực có đường dây đi qua như: tốc độ, việc mã hóa.

Còn đối với mạng cục bộ thì công việc đơn giản hơn nhiều, khi một cơ quan cài đặt mạng cục bộ thì toàn bộ mạng sẽ thuộc quyền quản lý của cơ quan đó.

Đường đi của thông tin trên mạng

Trong mạng cục bộ thông tin được đi theo con đường xác định bởi cấu trúc của mạng. Khi người ta xác định cấu trúc của mạng thì thông tin sẽ luôn luôn đi theo cấu trúc đã xác định đó. Còn với mạng diện rộng dữ liệu cấu trúc có thể phức tạp hơn nhiều do việc sử dụng các dịch vụ truyền dữ liệu. Trong quá trình hoạt động các điểm nút có thể thay đổi đường đi của các thông tin khi phát hiện ra có trục trặc trên đường truyền hay khi phát hiện có quá nhiều thông tin cần truyền giữa hai điểm nút nào đó. Trên mạng diện rộng thông tin có thể có các con đường đi khác nhau, điều đó cho phép có thể sử dụng tối đa các năng lực của đường truyền hay nâng cao điều kiện an toàn trong truyền dữ liệu.

Dạng chuyển giao thông tin

Phần lớn các mạng diện rộng hiện nay được phát triển cho việc truyền đồng thời trên đường truyền nhiều dạng thông tin khác nhau như: video, tiếng nói, dữ liệu... Trong khi đó các mạng cục bộ chủ yếu phát triển trong việc truyền dữ liệu thông thường. Điều này có thể giải thích do việc truyền các dạng thông tin như video, tiếng nói trong một khu vực nhỏ ít được quan tâm hơn như khi truyền qua những khoảng cách lớn.

Các hệ thống mạng hiện nay ngày càng phức tạp về chất lượng, đa dạng về chủng loại và phát triển rất nhanh về chất. Trong sự phát triển đó số lượng những nhà sản xuất từ phần mềm, phần cứng máy tính, các sản phẩm viễn thông cũng tăng nhanh với nhiều sản phẩm đa dạng. Chính vì vậy vai trò chuẩn hóa cũng mang những ý nghĩa quan trọng. Tại các nước các cơ quan chuẩn quốc gia đã đưa ra các những chuẩn về phần cứng và các quy định về giao tiếp nhằm giúp cho các nhà sản xuất có thể làm ra các sản phẩm có thể kết nối với các sản phẩm do hãng khác sản xuất.

IV. Các thành phần của mạng máy tính

Mạng máy tính bao gồm các thiết bị phần cứng, các giao thức và các phần mềm mạng. Khi nghiên cứu về mạng máy tính, các vấn đề quan trọng cần được xem xét là giao thức mạng, cấu hình kết nối của mạng và các dịch vụ mạng.

IV.1. Một số bộ giao thức kết nối mạng

1. TCP/IP

- Ưu thế chính của bộ giao thức này là khả năng liên kết hoạt động của nhiều loại máy tính khác nhau.
- TCP/IP đã trở thành tiêu chuẩn thực tế cho kết nối liên mạng cũng như kết nối Internet toàn cầu.

2. NetBEUI

- Bộ giao thức nhỏ, nhanh và hiệu quả được cung cấp theo các sản phẩm của hãng IBM, cũng như sự hỗ trợ của Microsoft.
- Bất lợi chính của bộ giao thức này là không hỗ trợ định tuyến và sử dụng giới hạn ở mạng dựa vào Microsoft.

3. IPX/SPX

- Đây là bộ giao thức sử dụng trong mạng Novell.
- Ưu thế: nhỏ, nhanh và hiệu quả trên các mạng cục bộ đồng thời hỗ trợ khả năng định tuyến.

4. DECnet

- Đây là bộ giao thức độc quyền của hãng Digital Equipment Corporation.
- DECnet định nghĩa mô hình truyền thông qua mạng LAN, mạng MAN và WAN. Hỗ trợ khả năng định tuyến

IV.2. Hệ điều hành mạng - NOS (*Network Operating System)*

Cùng với sự nghiên cứu và phát triển mạng máy tính, hệ điều hành mạng đã được nhiều công ty đầu tư nghiên cứu và đã công bố nhiều phần mềm quản lý và điều hành mạng có hiệu quả như: *NetWare* của công ty NOVELL, LAN *Manager* của *Microsoft* dùng cho các máy *server* chạy hệ điều hành OS/2, LAN server của IBM (gần như đồng nhất với LAN *Manager*), *Vines* của *Banyan Systems* là hệ điều hành mạng dùng cho *server* chạy hệ điều hành UNIX, *Promise* LAN của *Mises Computer* chạy trên *card* điều hợp mạng độc quyền, *Windows for Workgroups* của *Microsoft*, LANtastic của *Artisoft, NetWare Lite* của *Novell*,....

Một trong những sự lựa chọn cơ bản mà ta phải quyết định trước là hệ điều hành mạng nào sẽ làm nền tảng cho mạng của ta, việc lựa chọn tuỳ thuộc vào kích cỡ của mạng hiện tại và

sự phát triển trong tương lai, còn tuỳ thuộc vào những ưu điểm và nhược điểm của từng hệ điều hành.

Một số hệ điều hành mạng phổ biến hiện nay:

- Hệ điều hành mạng UNIX: Đây là hệ điều hành do các nhà khoa học xây dựng và được dùng rất phổ biến trong giới khoa học, giáo dục. Hệ điều hành mạng UNIX là hệ điều hành đa nhiệm, đa người sử dụng, phục vụ cho truyền thông tốt. Nhược điểm của nó là hiện nay có nhiều *Version* khác nhau, không thống nhất gây khó khăn cho người sử dụng. Ngoài ra hệ điều hành này khá phức tạp lại đòi hỏi cấu hình máy mạnh (trước đây chạy trên máy *mini*, gần đây có SCO UNIX chạy trên máy vi tính với cấu hình mạnh).
- Hệ điều hành mạng *Windows* NT: Đây là hệ điều hành của hãng *Microsoft*, cũng là hệ điều hành đa nhiệm, đa người sử dụng. Đặc điểm của nó là tương đối dễ sử dụng, hỗ trợ mạnh cho phần mềm WINDOWS. Do hãng *Microsoft* là hãng phần mềm lớn nhất thế giới hiện nay, hệ điều hành này có khả năng sẽ được ngày càng phổ biến rộng rãi. Ngoài ra, *Windows* NT có thể liên kết tốt với máy chủ *Novell Netware*. Tuy nhiên, để chạy có hiệu quả, *Windows* NT cũng đòi hỏi cấu hình máy tương đối mạnh.
- Hệ điều hành mạng *Windows for Workgroup*: Đây là hệ điều hành mạng ngang hàng nhỏ, cho phép một nhóm người làm việc (khoảng 3-4 người) dùng chung ổ đĩa trên máy của nhau, dùng chung máy in nhưng không cho phép chạy chung một ứng dụng. Hệ dễ dàng cài đặt và cũng khá phổ biến.
- Hệ điều hành mạng *NetWare của Novell*: Đây là hệ điều hành phổ biến nhất hiện nay ở nước ta và trên thế giới trong thời gian cuối, nó có thể dùng cho các mạng nhỏ (khoảng từ 5-25 máy tính) và cũng có thể dùng cho các mạng lớn gồm hàng trăm máy tính. Trong những năm qua, *Novell* đã cho ra nhiều phiên bản của *Netware*: Netware 2.2, 3.11. 4.0 và hiện có 4.1. *Netware* là một hệ điều hành mạng cục bộ dùng cho các máy vi tính theo chuẩn của IBM hay các máy tính *Apple Macintosh*, chạy hệ điều hành MS-DOS hoặc OS/2.

  Hệ điều hành này tương đối gọn nhẹ, dễ cài đặt (máy chủ chỉ cần thậm chí AT386) do đó phù hợp với hoàn cảnh trang thiết bị hiện tại của nước ta. Ngoài ra, vì là một phần mềm phổ biến nên Novell Netware được các nhà sản xuất phần mềm khác hỗ trợ (theo nghĩa các phân mềm do các hãng phần mềm lớn trên thế giới làm đều có thể chạy tốt trên hệ điều hành mạng này).

V. Các lợi ích của mạng máy tính

V.1. Mạng tạo khả năng dùng chung tài nguyên cho các người dùng.

Vấn đề là làm cho các tài nguyên trên mạng như chương trình, dữ liệu và thiết bị, đặc biệt là các thiết bị đắt tiền, có thể sẵn dùng cho mọi người trên mạng mà không cần quan tâm đến vị trí thực của tài nguyên và người dùng.

Về mặt thiết bị, các thiết bị chất lượng cao thường đắt tiền, chúng thường được dùng chung cho nhiều người nhằm giảm chi phí và dễ bảo quản.

Về mặt chương trình và dữ liệu, khi được dùng chung, mỗi thay đổi sẽ sẵn dùng cho mọi thành viên trên mạng ngay lập tức. Điều này thể hiện rất rõ tại các nơi như ngân hàng, các đại lý bán vé máy bay...

V.2. Mạng cho phép nâng cao độ tin cậy.

Khi sử dụng mạng, có thể thực hiện một chương trình tại nhiều máy tính khác nhau, nhiều thiết bị có thể dùng chung. Điều này tăng độ tin cậy trong công việc vì khi có máy tính hoặc thiết bị bị hỏng, công việc vẫn có thể tiếp tục với các máy tính hoặc thiết bị khác trên mạng trong khi chờ sửa chữa.

V.3. Mạng giúp cho công việc đạt hiệu suất cao hơn.

Khi chương trình và dữ liệu đã dùng chung trên mạng, có thể bỏ qua một số khâu đối chiếu không cần thiết. Việc điều chỉnh chương trình (nếu có) cũng tiết kiệm thời gian hơn do chỉ cần cài đặt lại trên một máy.

Về mặt tổ chức, việc sao chép dữ liệu phòng hờ tiện lợi hơn do có thể giao cho chỉ một người thay vì mọi người phải tự sao chép phần của mình.

V.4. Tiết kiệm chi phí.

Việc dùng chung các thiết bị ngoại vi cho phép giảm chi phí trang bị tính trên số người dùng. Về phần mềm, nhiều nhà sản xuất phần mềm cung cấp cả những ấn bản cho nhiều người dùng, với chi phí thấp hơn tính trên mỗi người dùng.

V.5. Tăng cường tính bảo mật thông tin.

Dữ liệu được lưu trên các máy phục vụ tập tin (file server) sẽ được bảo vệ tốt hơn so với đặt tại các máy cá nhân nhờ cơ chế bảo mật của các hệ điều hành mạng.

V.6. Việc phát triển mạng máy tính đã tạo ra nhiều ứng dụng mới

Một số ứng dụng có ảnh hưởng quan trọng đến toàn xã hội: khả năng truy xuất các chương trình và dữ liệu từ xa, khả năng thông tin liên lạc dễ dàng và hiệu quả, tạo môi trường giao tiếp thuận lợi giữa những người dùng khác nhau, khả năng tìm kiếm thông tin nhanh chóng trên phạm vi toàn thế giới,...

VI. Các dịch vụ phổ biến trên mạng máy tính

- **Dịch vụ tập tin (File services)**

Cho phép chia sẻ tài nguyên thông tin chung, chuyển giao các tập tin dữ liệu từ máy này sang máy khác. Tìm kiếm thông tin và điều khiển truy nhập. Dịch vụ thư điện tử E_Mail (Electronic mail) cung cấp cho người sử dụng phương tiện trao đổi, tranh luận bằng thư điện tử. Dịch vụ thư điện tử giá thành hạ, chuyển phát nhanh, an toàn và nội dung có thể tích hợp các loại dữ liệu.

- **Dịch vụ in ấn**

Có thể dùng chung các máy in đắt tiền trên mạng. Cung cấp khả năng đa truy nhập đến máy in, phục vụ đồng thời cho nhiều nhu cầu in khác nhau. Cung cấp các dịch vụ FAX và quản lý được các trang thiết bị in chuyên dụng.

- **Các dịch vụ ứng dụng hướng đối tượng**

Sử dụng các dịch vụ thông điệp (Message) làm trung gian tác động đến các đối tượng truyền thông. Đối tượng chỉ bàn giao dữ liệu cho tác nhân (Agent) và tác nhân sẽ bàn giao dữ liệu cho đối tượng đích.

- **Các dịch vụ ứng dụng quản trị luồng công việc trong nhóm làm việc:**

Định tuyến các tài liệu điện tử giữa những người trong nhóm. Khi chữ ký điện tử được xác nhận trong các phiên giao dịch thì có thể thay thế được nhiều tiến trình mới hiệu quả và nhanh chóng hơn.

- **Dịch vụ cơ sở dữ liệu**

Là dịch vụ phổ biến về các dịch vụ ứng dụng, là các ứng dụng theo mô hình Client/Server. Dịch vụ xử lý phân tán lưu trữ dữ liệu phân tán trên mạng, người dùng trong suốt và dễ sử dụng, đáp ứng các nhu cầu truy nhập của người sử dụng.

Chương II Mô hình truyền thông

I. Sự cần thiết phải có mô hình truyền thông

Để một mạng máy tính trở một môi trường truyền dữ liệu thì nó cần phải có những yếu tố sau:

- Mỗi máy tính cần phải có một địa chỉ phân biệt trên mạng.
- Việc chuyển dữ liệu từ máy tính này đến máy tính khác do mạng thực hiện thông qua những quy định thống nhất gọi là giao thức của mạng.

Khi các máy tính trao đổi dữ liệu với nhau thì một quá trình truyền giao dữ liệu đã được thực hiện hoàn chỉnh. Ví dụ như để thực hiện việc truyền một file giữa một máy tính với một máy tính khác cùng được gắn trên một mạng các công việc sau đây phải được thực hiện:

- Máy tính cần truyền cần biết địa chỉ của máy nhận.
- Máy tính cần truyền phải xác định được máy tính nhận đã sẵn sàng nhận thông tin
- Chương trình gửi file trên máy truyền cần xác định được rằng chương trình nhận file trên máy nhận đã sẵn sàng tiếp nhận file.
- Nếu cấu trúc file trên hai máy không giống nhau thì một máy phải làm nhiệm vụ chuyển đổi file từ dạng này sang dạng kia.
- Khi truyền file máy tính truyền cần thông báo cho mạng biết địa chỉ của máy nhận để các thông tin được mạng đưa tới đích.

Điều trên đó cho thấy giữa hai máy tính đã có một sự phối hợp hoạt động ở mức độ cao. Bây giờ thay vì chúng ta xét cả quá trình trên như là một quá trình chung thì chúng ta sẽ chia quá trình trên ra thành một số công đoạn và mỗi công đoạn con hoạt động một cách độc lập với nhau. Ở đây chương trình truyền nhận file của mỗi máy tính được chia thành ba module là: Module truyền và nhận File, Module truyền thông và Module tiếp cận mạng. Hai module tương ứng sẽ thực hiện việc trao đổi với nhau trong đó:

- Module truyền và nhận file cần được thực hiện tất cả các nhiệm vụ trong các ứng dụng truyền nhận file. Ví dụ: truyền nhận thông số về file, truyền nhận các mẫu tin của file, thực hiện chuyển đổi file sang các dạng khác nhau nếu cần. Module truyền và nhận file không cần thiết phải trực tiếp quan tâm tới việc truyền dữ liệu trên mạng như thế nào mà nhiệm vụ đó được giao cho Module truyền thông.
- Module truyền thông quan tâm tới việc các máy tính đang hoạt động và sẵn sàng trao đổi thông tin với nhau. Nó còn kiểm soát các dữ liệu sao cho những dữ liệu này có thể trao đổi một cách chính xác và an toàn giữa hai máy tính. Điều đó có nghĩa là phải truyền file trên nguyên tắc đảm bảo an toàn cho dữ liệu, tuy nhiên ở đây có thể có một vài mức độ an toàn khác nhau được dành cho từng ứng dụng. Ở đây việc trao đổi dữ liệu giữa hai máy tính không phụ thuộc vào bản chất của mạng đang liên kết chúng. Những yêu cầu liên quan đến mạng đã được thực hiện ở module thứ ba là module tiếp cận mạng và nếu mạng thay đổi thì chỉ có module tiếp cận mạng bị ảnh hưởng.
- Module tiếp cận mạng được xây dựng liên quan đến các quy cách giao tiếp với mạng và phụ thuộc vào bản chất của mạng. Nó đảm bảo việc truyền dữ liệu từ máy tính này đến máy tính khác trong mạng.

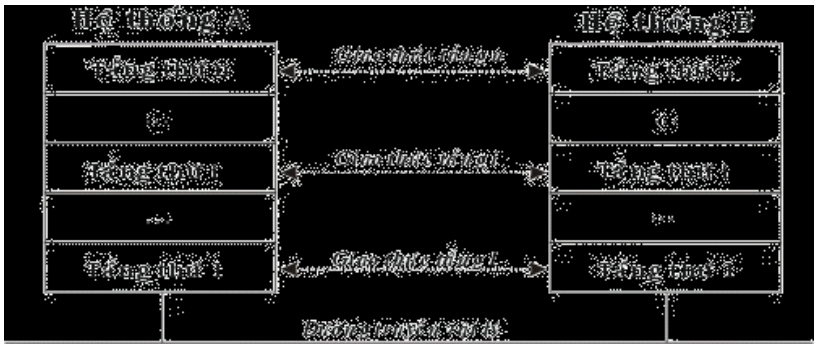
Như vậy thay vì xét cả quá trình truyền file với nhiều yêu cầu khác nhau như một tiến trình phức tạp thì chúng ta có thể xét quá trình đó với nhiều tiến trình con phân biệt dựa trên việc trao đổi

giữa các Module tương ứng trong chương trình truyền file. Cách này cho phép chúng ta phân tích kỹ quá trình file và dễ dàng trong việc viết chương trình.

Việc xét các module một cách độc lập với nhau như vậy cho phép giảm độ phức tạp cho việc thiết kế và cài đặt. Phương pháp này được sử dụng rộng rãi trong việc xây dựng mạng và các chương trình truyền thông và được gọi là phương pháp phân tầng (layer).

Nguyên tắc của phương pháp phân tầng là:

- Mỗi hệ thống thành phần trong mạng được xây dựng như một cấu trúc nhiều tầng và đều có cấu trúc giống nhau như: số lượng tầng và chức năng của mỗi tầng.
- Các tầng nằm chồng lên nhau, dữ liệu được chỉ trao đổi trực tiếp giữa hai tầng kề nhau từ tầng trên xuống tầng dưới và ngược lại.
- Cùng với việc xác định chức năng của mỗi tầng chúng ta phải xác định mối quan hệ giữa hai tầng kề nhau. Dữ liệu được truyền đi từ tầng cao nhất của hệ thống truyền lần lượt đến tầng thấp nhất sau đó truyền qua đường nối vật lý dưới dạng các bit tới tầng thấp nhất của hệ thống nhận, sau đó dữ liệu được truyền ngược lên lần lượt đến tầng cao nhất của hệ thống nhận.
- Chỉ có hai tầng thấp nhất có liên kết vật lý với nhau còn các tầng trên cùng thứ tư chỉ có các liên kết logic với nhau. Liên kết logic của một tầng được thực hiện thông qua các tầng dưới và phải tuân theo những quy định chặt chẽ, các quy định đó được gọi giao thức của tầng.



Hình II-1 Mô hình phân tầng

II. Các nhu cầu về chuẩn hóa đối với mạng

Trong thực tế việc phân chia các tầng như trong mô hình trên thực sự chưa đủ. Trên thế giới hiện có một số cơ quan định chuẩn, họ đưa ra hàng loạt chuẩn về mạng tuy các chuẩn đó có tính chất khuyến nghị chứ không bắt buộc nhưng chúng rất được các cơ quan chuẩn quốc gia coi trọng.

Hai trong số các cơ quan chuẩn quốc tế là:

- **ISO (The International Standards Organization)** - Là tổ chức tiêu chuẩn quốc tế hoạt động dưới sự bảo trợ của Liên hợp Quốc với thành viên là các cơ quan chuẩn quốc gia với số lượng khoảng hơn 100 thành viên với mục đích hỗ trợ sự phát triển các chuẩn trên phạm vi toàn thế giới. Một trong những thành tựu của ISO trong lãnh vực truyền thông là mô hình hệ thống mở (Open Systems Interconnection - gọi tắt là OSI).

- **CCITT (Commité Consultatif International pour le Telegraphe et la Téléphone)** - Tổ chức tư vấn quốc tế về điện tín và điện thoại làm việc dưới sự bảo trợ của Liên Hiệp Quốc có trụ sở chính tại Geneva - Thụy sỹ. Các thành viên chủ yếu là các cơ quan bưu chính viễn thông các quốc gia. Tổ chức này có vai trò phát triển các khuyến nghị trong các lãnh vực viễn thông.

III. Mô hình OSI (Open Systems Interconnection)

Mô hình OSI là một cơ sở dành cho việc chuẩn hoá các hệ thống truyền thông, nó được nghiên cứu và xây dựng bởi ISO. Việc nghiên cứu về mô hình OSI được bắt đầu tại ISO vào năm 1971 với mục tiêu nhằm tới việc nối kết các sản phẩm của các hãng sản xuất khác nhau và phối hợp các hoạt động chuẩn hoá trong các lĩnh vực viễn thông và hệ thống thông tin. Theo mô hình OSI chương trình truyền thông được chia ra thành 7 tầng với những chức năng phân biệt cho từng tầng. Hai tầng đồng mức khi liên kết với nhau phải sử dụng một giao thức chung.

Việc nghiên cứu về OSI được bắt đầu tại ISO vào năm 1971 với các mục tiêu nhằm nối kết các sản phẩm của các hãng sản xuất khác. Ưu điểm chính của OSI là ở chỗ nó hứa hẹn giải pháp cho vấn đề truyền thông giữa các máy tính không giống nhau. Hai hệ thống, dù có khác nhau đều có thể truyền thông với nhau một các hiệu quả nếu chúng đảm bảo những điều kiện chung sau đây:

- Chúng cài đặt cùng một tập các chức năng truyền thông.
  Các chức năng đó được tổ chức thành cùng một tập các tầng. các tầng đồng mức phải cung cấp các chức năng như nhau.
- Các tầng đồng mức khi trao đổi với nhau sử dụng chung một giao thức

Mô hình OSI tách các mặt khác nhau của một mạng máy tính thành bảy tầng theo mô hình phân tầng. Mô hình OSI là một khung mà các tiêu chuẩn lập mạng khác nhau có thể khớp vào. Mô hình OSI định rõ các mặt nào của hoạt động của mạng có thể nhằm đến bởi các tiêu chuẩn mạng khác nhau. Vì vậy, theo một nghĩa nào đó, mô hình OSI là một loại tiêu chuẩn của các chuẩn.

III.1. Nguyên tắc sử dụng khi định nghĩa các tầng hệ thống mở

Sau đây là các nguyên tắc mà ISO quy định dùng trong quá trình xây dựng mô hình OSI

- Không định nghĩa quá nhiều tầng để việc xác định và ghép nối các tầng không quá phức tạp.
- Tạo các ranh giới các tầng sao cho việc giải thích các phục vụ và số các tương tác qua lại hai tầng là nhỏ nhất.
- Tạo các tầng riêng biệt cho các chức năng khác biệt nhau hoàn toàn về kỹ thuật sử dụng hoặc quá trình thực hiên.
- Các chức năng giống nhau được đặt trong cùng một tầng.
  Lựa chọn ranh giới các tầng tại các điểm mà những thử nghiệm trong quá khứ thành công.
- Các chức năng được xác định sao cho chúng có thể dễ dàng xác định lại, và các nghi thức của chúng có thể thay đổi trên mọi hướng.
- Tạo ranh giới các tầng mà ở đó cần có những mức độ trừu tượng khác nhau trong việc sử dụng số liệu.
- Cho phép thay đổi các chức năng hoặc giao thức trong tầng không ảnh hưởng đến các tầng khác.

- Tạo các ranh giới giữa mỗi tầng với tầng trên và dưới nó.

III.2. Các giao thức trong mô hình OSI

Trong mô hình OSI có hai loại giao thức chính được áp dụng: giao thức có liên kết (connection - oriented) và giao thức không liên kết (connectionless).

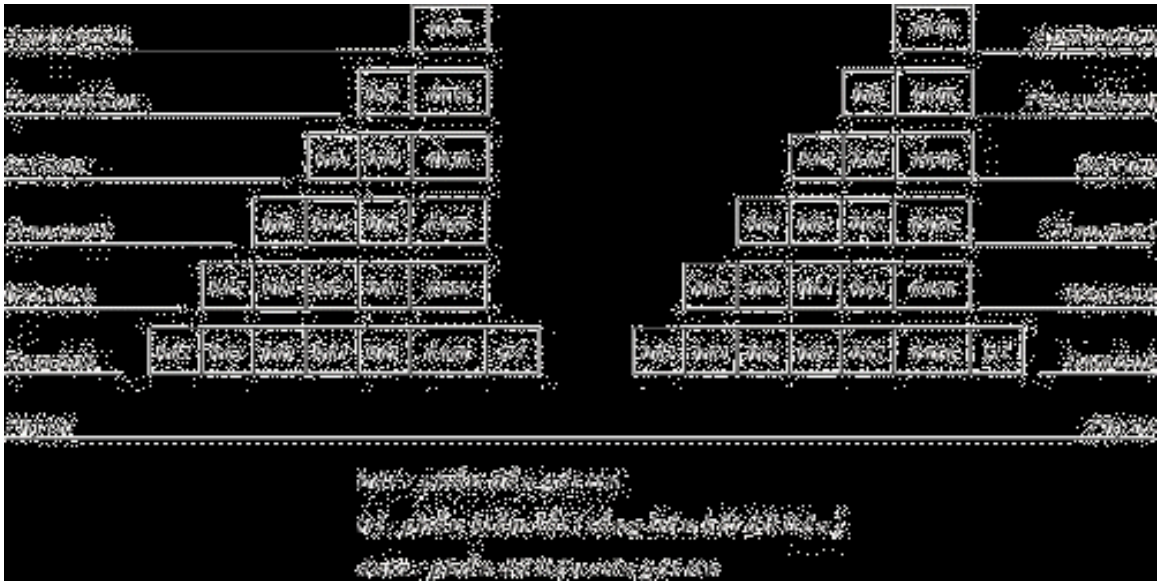
- *Giao thức có liên kết*: trước khi truyền dữ liệu hai tầng đồng mức cần thiết lập một liên kết logic và các gói tin được trao đổi thông qua liên kết náy, việc có liên kết logic sẽ nâng cao độ an toàn trong truyền dữ liệu.
- *Giao thức không liên kết*: trước khi truyền dữ liệu không thiết lập liên kết logic và mỗi gói tin được truyền độc lập với các gói tin trước hoặc sau nó.

Như vậy với giao thức có liên kết, quá trình truyền thông phải gồm 3 giai đoạn phân biệt:

- *Thiết lập liên kết (logic):* hai thực thể đồng mức ở hai hệ thống thương lượng với nhau về tập các tham số sẽ sử dụng trong giai đoạn sau (truyền dữ liệu).
- *Truyền dữ liệu*: dữ liệu được truyền với các cơ chế kiểm soát và quản lý kèm theo (như kiểm soát lỗi, kiểm soát luồng dữ liệu, cắt/hợp dữ liệu...) để tăng cường độ tin cậy và hiệu quả của việc truyền dữ liệu.
- *Hủy bỏ liên kết (logic):* giải phóng tài nguyên hệ thống đã được cấp phát cho liên kết để dùng cho liên kết khác.

Đối với giao thức không liên kết thì chỉ có duy nhất một giai đoạn truyền dữ liệu mà thôi.

Gói tin của giao thức: Gói tin (Packet) được hiểu như là một đơn vị thông tin dùng trong việc liên lạc, chuyển giao dữ liệu trong mạng máy tính. Những thông điệp (message) trao đổi giữa các máy tính trong mạng, được tạo dạng thành các gói tin ở máy nguồn. Và những gói tin này khi đích sẽ được kết hợp lại thành thông điệp ban đầu. Một gói tin có thể chứa đựng các yêu cầu phục vụ, các thông tin điều khiển và dữ liệu.



Hình II-2 Phương thức xác lập các gói tin trong mô hình OSI

Trên quan điểm mô hình mạng phân tầng tầng mỗi tầng chỉ thực hiện một chức năng là nhận dữ liệu từ tầng bên trên để chuyển giao xuống cho tầng bên dưới và ngược lại. Chức năng này thực chất là gắn thêm và gỡ bỏ phần đầu (header) đối với các gói tin trước khi chuyển nó đi. Nói cách khác, từng gói tin bao gồm phần đầu (header) và phần dữ liệu. Khi đi đến một tầng mới gói tin sẽ được đóng thêm một phần đầu đề khác và được xem như là gói tin của tầng mới, công việc trên tiếp diễn cho tới khi gói tin được truyền lên đường dây mạng để đến bên nhận.

Tại bên nhận các gói tin được gỡ bỏ phần đầu trên từng tầng tướng ứng và đây cũng là nguyên lý của bất cứ mô hình phân tầng nào.

Chú ý: Trong mô hình OSI phần kiểm lỗi của gói tin tầng liên kết dữ liệu đặt ở cuối gói tin

III.3. Các chức năng chủ yếu của các tầng của mô hình OSI.

III.3.1 Tầng 1: Vật lý (Physical)

Tầng vật lý (Physical layer) là tầng dưới cùng của mô hình OSI là. Nó mô tả các đặc trưng vật lý của mạng: Các loại cáp được dùng để nối các thiết bị, các loại đầu nối được dùng , các dây cáp có thể dài bao nhiêu v.v... Mặt khác các tầng vật lý cung cấp các đặc trưng điện của các tín hiệu được dùng để khi chuyển dữ liệu trên cáp từ một máy này đến một máy khác của mạng, kỹ thuật nối mạch điện, tốc độ cáp truyền dẫn.

Tầng vật lý không qui định một ý nghĩa nào cho các tín hiệu đó ngoài các giá trị nhị phân 0 và 1. Ở các tầng cao hơn của mô hình OSI ý nghĩa của các bit được truyền ở tầng vật lý sẽ được xác định.

Ví dụ: Tiêu chuẩn Ethernet cho cáp xoắn đôi 10 baseT định rõ các đặc trưng điện của cáp xoắn đôi, kích thước và dạng của các đầu nối, độ dài tối đa của cáp.

Khác với các tầng khác, tầng vật lý là không có gói tin riêng và do vậy không có phần đầu (header) chứa thông tin điều khiển, dữ liệu được truyền đi theo dòng bit. Một giao thức tầng vật lý tồn tại giữa các tầng vật lý để quy định về phương thức truyền (đồng bộ, phi đồng bộ), tốc độ truyền.

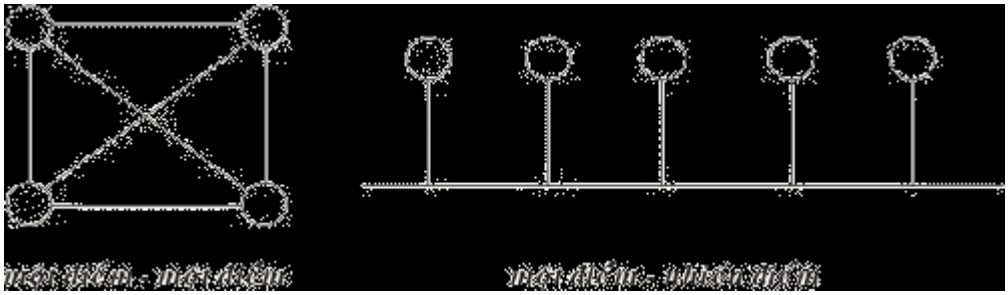
Các giao thức được xây dựng cho tầng vật lý được phân chia thành phân chia thành hai loại giao thức sử dụng phương thức truyền thông dị bộ (asynchronous) và phương thức truyền thông đồng bộ (synchronous).

- *Phương thức truyền dị bộ*: Không có một tín hiệu quy định cho sự đồng bộ giữa các bit giữa máy gửi và máy nhận, trong quá trình gửi tín hiệu máy gửi sử dụng các bit đặc biệt START và STOP được dùng để tách các xâu bit biểu diễn các ký tự trong dòng dữ liệu cần truyền đi. Nó cho phép một ký tự được truyền đi bất kỳ lúc nào mà không cần quan tâm đến các tín hiệu đồng bộ trước đó.
- *Phương thức truyền đồng bộ*: Sử dụng phương thức truyền cần có đồng bộ giữa máy gửi và máy nhận, nó chèn các ký tự đặc biệt như SYN (Synchronization), EOT (End Of Transmission) hay đơn giản hơn, một cái "cờ " (flag) giữa các dữ liệu của máy gửi để báo hiệu cho máy nhận biết được dữ liệu đang đến hoặc đã đến.

III.3.2 Tầng 2: Liên kết dữ liệu (Data link)

Tầng liên kết dữ liệu (data link layer) là tầng mà ở đó ý nghĩa được gán cho các bít được truyền trên mạng. Tầng liên kết dữ liệu phải quy định được các dạng thức, kích thước, địa chỉ máy gửi và nhận của mỗi gói tin được gửi đi. Nó phải xác định cơ chế truy nhập thông tin trên mạng và phương tiện gửi mỗi gói tin sao cho nó được đưa đến cho người nhận đã định.

Tầng liên kết dữ liệu có hai phương thức liên kết dựa trên cách kết nối các máy tính, đó là phương thức "một điểm - một điểm" và phương thức "một điểm - nhiều điểm". Với phương thức "một điểm - một điểm" các đường truyền riêng biệt được thiết lập để nối các cặp máy tính lại với nhau. Phương thức "một điểm - nhiều điểm " tất cả các máy phân chia chung một đường truyền vật lý.



Hình II-3 : Các đường truyền kết nối kiểu "một điểm - một điểm" và "một điểm - nhiều điểm"

Tầng liên kết dữ liệu cũng cung cấp cách phát hiện và sửa lỗi cơ bản để đảm bảo cho dữ liệu nhận được giống hoàn toàn với dữ liệu gửi đi. Nếu một gói tin có lỗi không sửa được, tầng liên kết dữ liệu phải chỉ ra được cách thông báo cho nơi gửi biết gói tin đó có lỗi để nó gửi lại.

Các giao thức tầng liên kết dữ liệu chia làm 2 loại chính là các giao thức hướng ký tư và các giao thức hướng bit. Các giao thức hướng ký tự được xây dựng dựa trên các ký tự đặc biệt của một bộ mã chuẩn nào đó (như ASCII hay EBCDIC), trong khi đó các giao thức hướng bit lại dùng các cấu trúc nhị phân (xâu bit) để xây dựng các phần tử của giao thức (đơn vị dữ liệu, các thủ tục.) và khi nhận, dữ liệu sẽ được tiếp nhận lần lượt từng bit một.

III.3.3 Tầng 3: Mạng (Network)

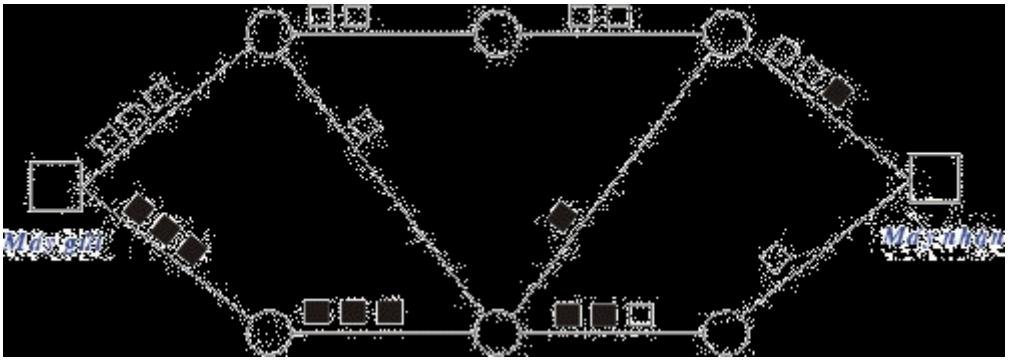
Tầng mạng (network layer) nhắm đến việc kết nối các mạng với nhau bằng cách tìm đường (routing) cho các gói tin từ một mạng này đến một mạng khác. Nó xác định việc chuyển hướng, vạch đường các gói tin trong mạng, các gói này có thể phải đi qua nhiều chặng trước khi đến được đích cuối cùng. Nó luôn tìm các tuyến truyền thông không tắc nghẽn để đưa các gói tin đến đích.

Tầng mạng cung các các phương tiện để truyền các gói tin qua mạng, thậm chí qua một mạng của mạng (network of network). Bởi vậy nó cần phải đáp ứng với nhiều kiểu mạng và nhiều kiểu dịch vụ cung cấp bởi các mạng khác nhau. hai chức năng chủ yếu của tầng mạng là chọn đường (routing) và chuyển tiếp (relaying). Tầng mạng là quan trọng nhất khi liên kết hai loại mạng khác nhau như mạng Ethernet với mạng Token Ring khi đó phải dùng một bộ tìm đường (quy định bởi tầng mạng) để chuyển các gói tin từ mạng này sang mạng khác và ngược lại.

Đối với một mạng chuyển mạch gói (packet - switched network) - gồm tập hợp các nút chuyển mạch gói nối với nhau bởi các liên kết dữ liệu. Các gói dữ liệu được truyền từ một hệ thống mở tới một hệ thống mở khác trên mạng phải được chuyển qua một chuỗi các nút. Mỗi nút nhận gói dữ liệu từ một đường vào (incoming link) rồi chuyển tiếp nó tới một đường ra (outgoing link) hướng đến đích của dữ liệu. Như vậy ở mỗi nút trung gian nó phải thực hiện các chức năng chọn đường và chuyển tiếp.

Việc chọn đường là sự lựa chọn một con đường để truyền một đơn vị dữ liệu (một gói tin chẳng hạn) từ trạm nguồn tới trạm đích của nó. Một kỹ thuật chọn đường phải thực hiện hai chức năng chính sau đây:

- Quyết định chọn đường tối ưu dựa trên các thông tin đã có về mạng tại thời điểm đó thông qua những tiêu chuẩn tối ưu nhất định.
- Cập nhật các thông tin về mạng, tức là thông tin dùng cho việc chọn đường, trên mạng luôn có sự thay đổi thường xuyên nên việc cập nhật là việc cần thiết.



Hình II-4 Mô hình chuyển vận các gói tin trong mạng chuyển mạch gói

Người ta có hai phương thức đáp ứng cho việc chọn đường là phương thức xử lý tập trung và xử lý tại chỗ.

- *Phương thức chọn đường xử lý tập trung* được đặc trưng bởi sự tồn tại của một (hoặc vài) trung tâm điều khiển mạng, chúng thực hiện việc lập ra các bảng đường đi tại từng thời điểm cho các nút và sau đó gửi các bảng chọn đường tới từng nút dọc theo con đường đã được chọn đó. Thông tin tổng thể của mạng cần dùng cho việc chọn đường chỉ cần cập nhập và được cất giữ tại trung tâm điều khiển mạng.
- *Phương thức chọn đường xử lý tại chỗ* được đặc trưng bởi việc chọn đường được thực hiện tại mỗi nút của mạng. Trong từng thời điểm, mỗi nút phải duy trì các thông tin của mạng và tự xây dựng bảng chọn đường cho mình. Như vậy các thông tin tổng thể của mạng cần dùng cho việc chọn đường cần cập nhập và được cất giữ tại mỗi nút.

Thông thường các thông tin được đo lường và sử dụng cho việc chọn đường bao gồm:

- Trạng thái của đường truyền.
- Thời gian trễ khi truyền trên mỗi đường dẫn.
- Mức độ lưu thông trên mỗi đường.
- Các tài nguyên khả dụng của mạng.

Khi có sự thay đổi trên mạng (ví dụ thay đổi về cấu trúc của mạng do sự cố tại một vài nút, phục hồi của một nút mạng, nối thêm một nút mới... hoặc thay đổi về mức độ lưu thông) các thông tin trên cần được cập nhật vào các cơ sở dữ liệu về trạng thái của mạng.

Hiện nay khi nhu cầu truyền thông đa phương tiện (tích hợp dữ liệu văn bản, đồ hoạ, hình ảnh, âm thanh) ngày càng phát triển đòi hỏi các công nghệ truyền dẫn tốc độ cao nên việc phát triển các hệ thống chọn đường tốc độ cao đang rất được quan tâm.

III.3.4 Tầng 4: Vận chuyển (Transport)

Tầng vận chuyển cung cấp các chức năng cần thiết giữa tầng mạng và các tầng trên. nó là tầng cao nhất có liên quan đến các giao thức trao đổi dữ liệu giữa các hệ thống mở. Nó cùng các tầng dưới cung cấp cho người sử dụng các phục vụ vận chuyển.

Tầng vận chuyển (transport layer) là tầng cơ sở mà ở đó một máy tính của mạng chia sẻ thông tin với một máy khác. Tầng vận chuyển đồng nhất mỗi trạm bằng một địa chỉ duy nhất và quản lý sự kết nối giữa các trạm. Tầng vận chuyển cũng chia các gói tin lớn thành các gói tin nhỏ hơn trước khi gửi đi. Thông thường tầng vận chuyển đánh số các gói tin và đảm bảo chúng chuyển theo đúng thứ tự.

Tầng vận chuyển là tầng cuối cùng chịu trách nhiệm về mức độ an toàn trong truyền dữ liệu nên giao thức tầng vận chuyển phụ thuộc rất nhiều vào bản chất của tầng mạng. Người ta chia giao thức tầng mạng thành các loại sau:

- Mạng loại A: Có tỷ suất lỗi và sự cố có báo hiệu chấp nhận được (tức là chất lượng chấp nhận được). Các gói tin được giả thiết là không bị mất. Tầng vận chuyển không cần cung cấp các dịch vụ phục hồi hoặc sắp xếp thứ tự lại.
- Mạng loại B: Có tỷ suất lỗi chấp nhận được nhưng tỷ suất sự cố có báo hiệu lại không chấp nhận được. Tầng giao vận phải có khả năng phục hồi lại khi xảy ra sự cố.
- Mạng loại C: Có tỷ suất lỗi không chấp nhận được (không tin cậy) hay là giao thức không liên kết. Tầng giao vận phải có khả năng phục hồi lại khi xảy ra lỗi và sắp xếp lại thứ tự các gói tin.

Trên cơ sở loại giao thức tầng mạng chúng ta có 5 lớp giao thức tầng vận chuyển đó là:

- *Giao thức lớp 0 (Simple Class - lớp đơn giản)*: cung cấp các khả năng rất đơn giản để thiết lập liên kết, truyền dữ liệu và hủy bỏ liên kết trên mạng "có liên kết" loại A. Nó có khả năng phát hiện và báo hiệu các lỗi nhưng không có khả năng phục hồi.
- *Giao thức lớp 1 (Basic Error Recovery Class - Lớp phục hồi lỗi cơ bản)* dùng với các loại mạng B, ở đây các gói tin (TPDU) được đánh số. Ngoài ra giao thức còn có khả năng báo nhận cho nơi gửi và truyền dữ liệu khẩn. So với giao thức lớp 0 giao thức lớp 1 có thêm khả năng phục hồi lỗi.
- *Giao thức lớp 2 (Multiplexing Class - lớp dồn kênh)* là một cải tiến của lớp 0 cho phép dồn một số liên kết chuyển vận vào một liên kết mạng duy nhất, đồng thời có thể kiểm soát luồng dữ liệu để tránh tắc nghẽn. Giao thức lớp 2 không có khả năng phát hiện và phục hồi lỗi. Do vậy nó cần đặt trên một tầng mạng loại A.

- *Giao thức lớp 3 (Error Recovery and Multiplexing Class - lớp phục hồi lỗi cơ bản và dồn kênh)* là sự mở rộng giao thức lớp 2 với khả năng phát hiện và phục hồi lỗi, nó cần đặt trên một tầng mạng loại B.
- *Giao thức lớp 4 (Error Detection and Recovery Class - Lớp phát hiện và phục hồi lỗi)* là lớp có hầu hết các chức năng của các lớp trước và còn bổ sung thêm một số khả năng khác để kiểm soát việc truyền dữ liệu.

III.3.5 Tầng 5: Giao dịch (Session)

Tầng giao dịch (session layer) thiết lập "các giao dịch" giữa các trạm trên mạng, nó đặt tên nhất quán cho mọi thành phần muốn đối thoại với nhau và lập ánh xa giữa các tên với địa chỉ của chúng. Một giao dịch phải được thiết lập trước khi dữ liệu được truyền trên mạng, tầng giao dịch đảm bảo cho các giao dịch được thiết lập và duy trì theo đúng qui định.

Tầng giao dịch còn cung cấp cho người sử dụng các chức năng cần thiết để quản trị các giao dịnh ứng dụng của họ, cụ thể là:

- Điều phối việc trao đổi dữ liệu giữa các ứng dụng bằng cách thiết lập và giải phóng (một cách lôgic) các phiên (hay còn gọi là các hội thoại - dialogues)
- Cung cấp các điểm đồng bộ để kiểm soát việc trao đổi dữ liệu.
- Áp đặt các qui tắc cho các tương tác giữa các ứng dụng của người sử dụng.
- Cung cấp cơ chế "lấy lượt" (nắm quyền) trong quá trình trao đổi dữ liệu.

Trong trường hợp mạng là hai chiều luân phiên thì nẩy sinh vấn đề: hai người sử dụng luân phiên phải "lấy lượt" để truyền dữ liệu. Tầng giao dịch duy trì tương tác luân phiên bằng cách báo cho mỗi người sử dụng khi đến lượt họ được truyền dữ liệu. Vấn đề đồng bộ hóa trong tầng giao dịch cũng được thực hiện như cơ chế kiểm tra/phục hồi, dịch vụ này cho phép người sử dụng xác định các điểm đồng bộ hóa trong dòng dữ liệu đang chuyển vận và khi cần thiết có thể khôi phục việc hội thoại bắt đầu từ một trong các điểm đó

Ở một thời điểm chỉ có một người sử dụng đó quyền đặc biệt được gọi các dịch vụ nhất định của tầng giao dịch, việc phân bổ các quyền này thông qua trao đổi thẻ bài (token). Ví dụ: Ai có được token sẽ có quyền truyền dữ liệu, và khi người giữ token trao token cho người khác thi cũng có nghĩa trao quyền truyền dữ liệu cho người đó.

Tầng giao dịch có các hàm cơ bản sau:

- *Give Token* cho phép người sử dụng chuyển một token cho một người sử dụng khác của một liên kết giao dịch.
- *Please Token* cho phép một người sử dụng chưa có token có thể yêu cầu token đó.
- *Give Control* dùng để chuyển tất cả các token từ một người sử dụng sang một người sử dụng khác.

III.3.6 Tầng 6: Trình bày (Presentation)

Trong giao tiếp giữa các ứng dụng thông qua mạng với cùng một dữ liệu có thể có nhiều cách biểu diễn khác nhau. Thông thường dạng biểu diễn dùng bởi ứng dụng nguồn và dạng biểu diễn dùng bởi ứng dụng đích có thể khác nhau do các ứng dụng được chạy trên các hệ thống hoàn toàn khác nhau (như hệ máy Intel và hệ máy Motorola). Tầng trình bày (Presentation layer) phải chịu trách nhiệm chuyển đổi dữ liệu gửi đi trên mạng từ một loại biểu diễn này sang một loại

khác. Để đạt được điều đó nó cung cấp một dạng biểu diễn chung dùng để truyền thông và cho phép chuyển đổi từ dạng biểu diễn cục bộ sang biểu diễn chung và ngược lại.

Tầng trình bày cũng có thể được dùng kĩ thuật mã hóa để xáo trộn các dữ liệu trước khi được truyền đi và giải mã ở đầu đến để bảo mật. Ngoài ra tầng biểu diễn cũng có thể dùng các kĩ thuật nén sao cho chỉ cần một ít byte dữ liệu để thể hiện thông tin khi nó được truyền ở trên mạng, ở đầu nhận, tầng trình bày bung trở lại để được dữ liệu ban đầu.

III.3.7 Tầng 7: Ứng dụng (Application)

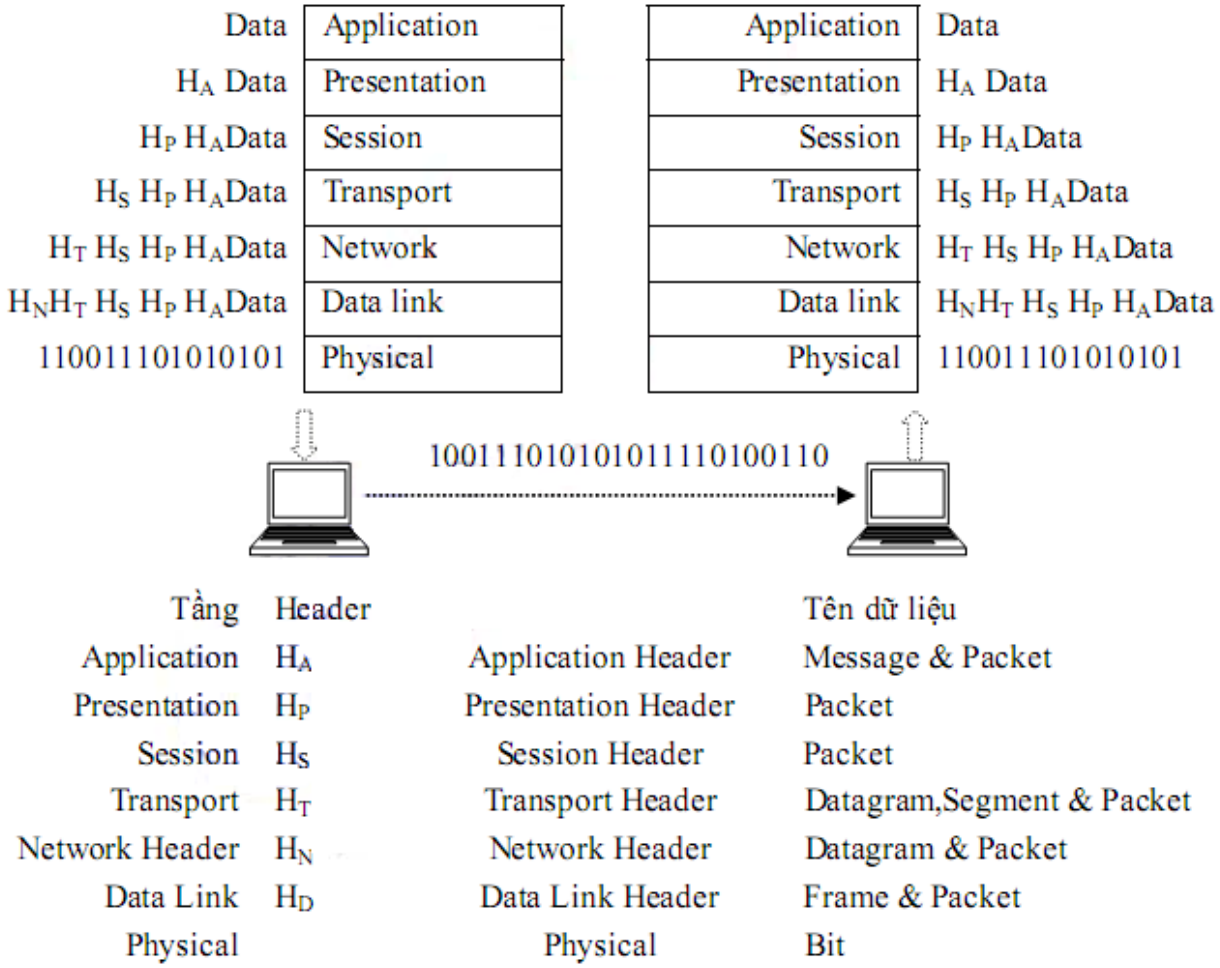
Tầng ứng dụng (Application layer) là tầng cao nhất của mô hình OSI, nó xác định giao diện giữa người sử dụng và môi trường OSI và giải quyết các kỹ thuật mà các chương trình ứng dụng dùng để giao tiếp với mạng.

Để cung cấp phương tiện truy nhập môi trường OSI cho các tiến trình ứng dụng, Người ta thiết lập các thực thể ứng dụng (AE), các thực thể ứng dụng sẽ gọi đến các phần tử dịch vụ ứng dụng (Application Service Element - viết tắt là ASE) của chúng. Mỗi thực thể ứng dụng có thể gồm một hoặc nhiều các phần tử dịch vụ ứng dụng. Các phần tử dịch vụ ứng dụng được phối hợp trong môi trường của thực thể ứng dụng thông qua các liên kết (association) gọi là đối tượng liên kết đơn (Single Association Object - viết tắt là SAO). SAO điều khiển việc truyền thông trong suốt vòng đời của liên kết đó cho phép tuần tự hóa các sự kiện đến từ các ASE thành tố của nó.

IV. Quá trình chuyển vận gói tin

IV.1. Quá trình đóng gói dữ liệu (tại máy gửi)

Đóng gói dữ liệu là quá trình đặt dữ liệu nhận được vào sau header (và trước trailer) trên mỗi lớp. Lớp Physical không đóng gói dữ liệu vì nó không dùng header và trailer. Việc đóng gói dữ liệu không nhất thiết phải xảy ra trong mỗi lần truyền dữ liệu của trình ứng dụng. Các lớp 5, 6, 7 sử dụng header trong quá trình khởi động, nhưng trong phần lớn các lần truyền thì không có header của lớp 5, 6, 7 lý do là không có thông tin mới để trao đổi.



Tầng	Header		Tên dữ liệu
Application	H_A	Application Header	Message & Packet
Presentation	H_P	Presentation Header	Packet
Session	H_S	Session Header	Packet
Transport	H_T	Transport Header	Datagram,Segment & Packet
Network Header	H_N	Network Header	Datagram & Packet
Data Link	H_D	Data Link Header	Frame & Packet
Physical		Physical	Bit

Hình II-5 Bổ sung phần đầu thông điệp & tên dữ liệu sử dụng

Các dữ liệu tại máy gửi được xử lý theo trình tự như sau:

- Người dùng thông qua lớp Application để đưa các thông tin vào máy tính. Các thông tin này có nhiều dạng khác nhau như: hình ảnh, âm thanh, văn bản…

- Tiếp theo các thông tin đó được chuyển xuống lớp Presentation để chuyển thành dạng chung, rồi mã hoá và nén dữ liệu.

- Tiếp đó dữ liệu được chuyển xuống lớp Session để bổ sung các thông tin về phiên giao dịch này.

- Dữ liệu tiếp tục được chuyển xuống lớp Transport, tại lớp này dữ liệu được cắt ra thành nhiều Segment và bổ sung thêm các thông tin về phương thức vận chuyển dữ liệu để đảm bảo độ tin cậy khi truyền.

- Dữ liệu tiếp tục được chuyển xuống lớp Network, tại lớp này mỗi Segment được cắt ra thành nhiều Packet và bổ sung thêm các thông tin định tuyến.

- Tiếp đó dữ liệu được chuyển xuống lớp Data Link, tại lớp này mỗi Packet sẽ được cắt ra thành nhiều Frame và bổ sung thêm các thông tin kiểm tra gói tin (để kiểm tra ở nơi nhận).

- Cuối cùng, mỗi Frame sẽ được tầng Vật Lý chuyển thành một chuỗi các bit, và được đẩy lên các phương tiện truyền dẫn để truyền đến các thiết bị khác.

IV.2. Quá trình truyền dữ liệu từ máy gửi đến máy nhận.

Bước 1: Trình ứng dụng (trên máy gửi) tạo ra dữ liệu và các chương trình phần cứng, phần mềm cài đặt mỗi lớp sẽ bổ sung vào header và trailer (quá trình đóng gói dữ liệu tại máy gửi).

Bước 2: Lớp Physical (trên máy gửi) phát sinh tín hiệu lên môi trường truyền tải để truyền dữ liệu.

Bước 3: Lớp Physical (trên máy nhận) nhận dữ liệu.

Bước 4: Các chương trình phần cứng, phần mềm (trên máy nhận) gỡ bỏ header và trailer và xử lý phần dữ liệu (quá trình xử lý dữ liệu tại máy nhận).

Giữa bước 1 và bước 2 là quá trình tìm đường đi của gói tin. Thông thường, máy gửi đã biết địa chỉ IP của máy nhận. Vì thế, sau khi xác định được địa chỉ IP của máy nhận thì lớp Network của máy gửi sẽ so sánh địa chỉ IP của máy nhận và địa chỉ IP của chính nó:

- Nếu cùng địa chỉ mạng thì máy gửi sẽ tìm trong bảng MAC Table của mình để có được địa chỉ MAC của máy nhận. Trong trường hợp không có được địa chỉ MAC tương ứng, nó sẽ thực hiện giao thức ARP để truy tìm địa chỉ MAC. Sau khi tìm được địa chỉ MAC, nó sẽ lưu địa chỉ MAC này vào trong bảng MAC Table để lớp Datalink sử dụng ở các lần gửi sau. Sau khi có địa chỉ MAC thì máy gửi sẽ gởi gói tin đi.

- Nếu khác địa chỉ mạng thì máy gửi sẽ kiểm tra xem máy có được khai báo Default Gateway hay không.

+ Nếu có khai báo Default Gateway thì máy gửi sẽ gởi gói tin thông qua Default Gateway.

+ Nếu không có khai báo Default Gateway thì máy gởi sẽ loại bỏ gói tin và thông báo "Destination host Unreachable"

IV.3. Chi tiết quá trình xử lý tại máy nhận

Bước 1: Lớp Physical kiểm tra quá trình đồng bộ bit và đặt chuỗi bit nhận được vào vùng đệm. Sau đó thông báo cho lớp Data Link dữ liệu đã được nhận.

Bước 2: Lớp Data Link kiểm lỗi frame bằng cách kiểm tra FCS trong trailer. Nếu có lỗi thì frame bị bỏ.

Sau đó kiểm tra địa chỉ lớp Data Link (địa chỉ MAC) xem có trùng với địa chỉ máy nhận hay không. Nếu đúng thì phần dữ liệu sau khi loại header và trailer sẽ được chuyển lên cho lớp Network.

Bước 3: Địa chỉ lớp Network được kiểm tra xem có phải là địa chỉ máy nhận hay không (địa chỉ IP) ? Nếu đúng thì dữ liệu được chuyển lên cho lớp Transport xử lý.

Bước 4: Nếu giao thức lớp Transport có hỗ trợ việc phục hồi lỗi thì số định danh phân đoạn được xử lý. Các thông tin ACK, NAK (gói tin ACK, NAK dùng để phản hồi về việc các gói tin đã được gởi đến máy nhận chưa) cũng được xử lý ở lớp này. Sau quá trình phục hồi lỗi và sắp thứ tự các phân đoạn, dữ liệu được đưa lên lớp Session.

Bước 5: Lớp Session đảm bảo một chuỗi các thông điệp đã trọn vẹn. Sau khi các luồng đã hoàn tất, lớp Session chuyển dữ liệu sau header lớp 5 lên cho lớp Presentation xử lý.

Bước 6: Dữ liệu sẽ được lớp Presentation xử lý bằng cách chuyển đổi dạng thức dữ liệu. Sau đó kết quả chuyển lên cho lớp Application.

Bước 7: Lớp Application xử lý header cuối cùng. Header này chứa các tham số thoả thuận giữa hai trình ứng dụng. Do vậy tham số này thường chỉ được trao đổi lúc khởi động quá trình truyền thông giữa hai trình ứng dụng.

V. Phương thức truyền tín hiệu

Thông thường có hai phương thức truyền tín hiệu trong mạng cục bộ là dùng băng tần cơ sở (baseband) và băng tần rộng (broadband). Sự khác nhau chủ yếu giữa hai phương thức truyền tín hiệu này là băng tầng cơ sở chỉ chấp nhận một kênh dữ liệu duy nhất trong khi băng rộng có thể chấp nhận đồng thời hai hoặc nhiều kênh truyền thông cùng phân chia giải thông của đường truyền.

Hầu hết các mạng cục bộ sử dụng phương thức băng tần cơ sở. Với phương thức truyền tín hiệu này này tín hiệu có thể được truyền đi dưới cả hai dạng: tương tự (analog) hoặc số (digital). Phương thức truyền băng tần rộng chia giải thông (tần số) của đường truyền thành nhiều giải tần con trong đó mỗi dải tần con đó cung cấp một kênh truyền dữ liệu tách biệt nhờ sử dụng một cặp modem đặc biệt gọi là bộ giải / Điều biến RF cai quản việc biến đổi các tín hiệu số thành tín hiệu tương tự có tần số vô tuyến (RF) bằng kỹ thuật ghép kênh.

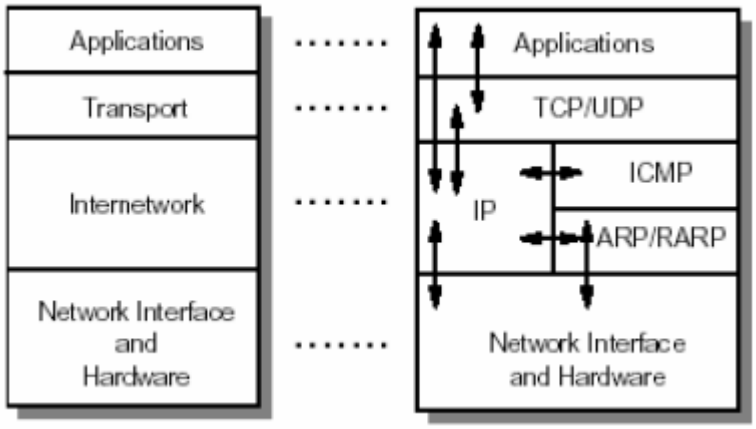
VI. Mô hình TCP/IP

VI.1. Tổng quan về bộ giao thức TCP/IP

TCP/IP là bộ giao thức cho phép kết nối các hệ thống mạng không đồng nhất với nhau. Ngày nay, TCP/IP được sử dụng rộng rãi trong các mạng cục bộ cũng như trên mạng Internet toàn cầu.

TCP/IP được xem là giản lược của mô hình tham chiếu OSI với bốn tầng như sau:

- Tầng liên kết mạng (Network Access Layer)
- Tầng Internet (Internet Layer)
- Tầng giao vận (Host-to-Host Transport Layer)
- Tầng ứng dụng (Application Layer)



Hình II-6 : Kiến trúc TCP/IP

Tầng liên kết:

Tầng liên kết (còn được gọi là tầng liên kết dữ liệu hay là tầng giao tiếp mạng) là tầng thấp nhất trong mô hình TCP/IP, bao gồm các thiết bị giao tiếp mạng và chương trình cung cấp các thông tin cần thiết để có thể hoạt động, truy nhập đường truyền vật lý qua thiết bị giao tiếp mạng đó.

Tầng Internet:

Tầng Internet (còn gọi là tầng mạng) xử lý qua trình truyền gói tin trên mạng. Các giao thức của tầng này bao gồm: IP (Internet Protocol), ICMP (Internet Control Message Protocol), IGMP (Internet Group Messages Protocol).

Tầng giao vận:

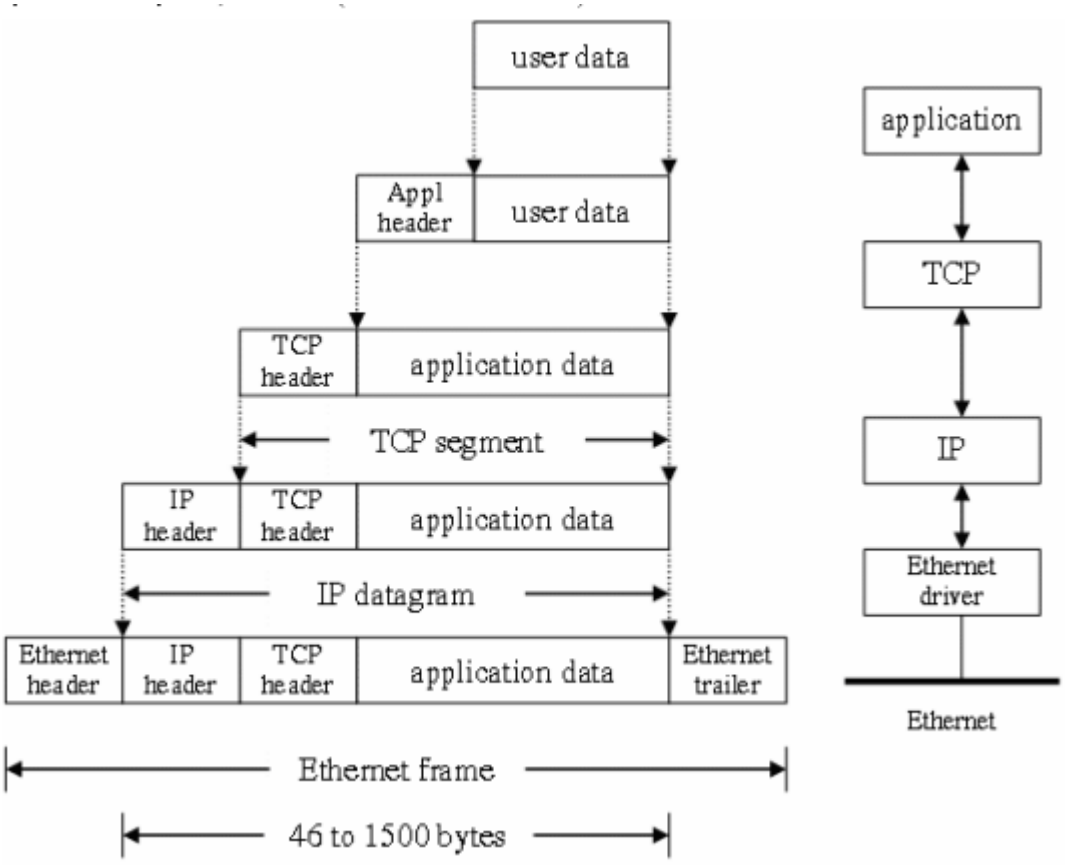
Tầng giao vận phụ trách luồng dữ liệu giữa hai trạm thực hiện các ứng dụng của tầng trên. Tầng này có hai giao thức chính: TCP (Transmission Control Protocol) và UDP (User Datagram Protocol)

TCP cung cấp một luồng dữ liệu tin cậy giữa hai trạm, nó sử dụng các cơ chế như chia nhỏ các gói tin của tầng trên thành các gói tin có kích thước thích hợp cho tầng mạng bên dưới, báo nhận gói tin,đặt hạn chế thời gian time-out để đảm bảo bên nhận biết được các gói tin đã gửi đi. Do tầng này đảm bảo tính tin cậy, tầng trên sẽ không cần quan tâm đến nữa.

UDP cung cấp một dịch vụ đơn giản hơn cho tầng ứng dụng. Nó chỉ gửi các gói dữ liệu từ trạm này tới trạm kia mà không đảm bảo các gói tin đến được tới đích. Các cơ chế đảm bảo độ tin cậy cần được thực hiện bởi tầng trên.

Tầng ứng dụng:

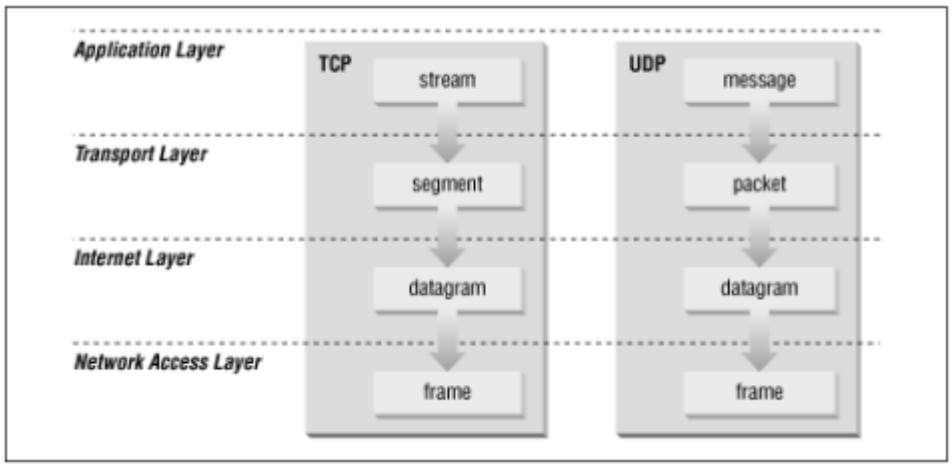
Tầng ứng dụng là tầng trên cùng của mô hình TCP/IP bao gồm các tiến trình và các ứng dụng cung cấp cho người sử dụng để truy cập mạng. Có rất nhiều ứng dụng được cung cấp trong tầng này, mà phổ biến là: Telnet: sử dụng trong việc truy cập mạng từ xa, FTP (File Transfer Protocol): dịch vụ truyền tệp, Email: dịch vụ thư tín điện tử, WWW (World Wide Web).



Hình II-7: Quá trình đóng/mở gói dữ liệu trong TCP/IP

Cũng tương tự như trong mô hình OSI, khi truyền dữ liệu, quá trình tiến hành từ tầng trên xuống tầng dưới, qua mỗi tầng dữ liệu được thêm vào một thông tin điều khiển được gọi là phần header. Khi nhận dữ liệu thì quá trình xảy ra ngược lại, dữ liệu được truyền từ tầng dưới lên và qua mỗi tầng thì phần header tương ứng được lấy đi và khi đến tầng trên cùng thì dữ liệu không còn phần header nữa. Hình vẽ sau cho ta thấy lược đồ dữ liệu qua các tầng. Trong hình vẽ này ta thấy tại các tầng khác nhau dữ liệu được mang những thuật ngữ khác nhau:

- Trong tầng ứng dụng dữ liệu là các luồng được gọi là stream.
- Trong tầng giao vận, đơn vị dữ liệu mà TCP gửi xuống tầng dưới gọi là TCP segment.
- Trong tầng mạng, dữ liệu mà IP gửi tới tầng dưới được gọi là IP datagram.
- Trong tầng liên kết, dữ liệu được truyền đi gọi là frame.



Hình II-8: Cấu trúc dữ liệu trong TCP/IP

VI.2. So sánh TCP/IP với OSI

Mỗi tầng trong TCP/IP có thể là một hay nhiều tầng của OSI.

Bảng sau chỉ rõ mối tương quan giữa các tầng trong mô hình TCP/IP với OSI

OSI	TCP/IP
Physical Layer và Data link Layer	Data link Layer
Network Layer	Internet Layer
Transport Layer	Transport Layer
Session Layer, Presentation Layer, Application Layer	Application Layer

Sự khác nhau giữa TCP/IP và OSI chỉ là:

- Tầng ứng dụng trong mô hình TCP/IP bao gồm luôn cả 3 tầng trên của mô hình OSI
- Tầng giao vận trong mô hình TCP/IP không phải luôn đảm bảo độ tin cậy của việc truyền tin như ở trong tầng giao vận của mô hình OSI mà cho phép thêm một lựa chọn khác là UDP

VII. Các giao thức truy cập đường truyền trên mạng LAN

Để truyền được dữ liệu trên mạng người ta phải có các thủ tục nhằm hướng dẫn các máy tính của mạng làm thế nào và lúc nào có thể thâm nhập vào đường dây cáp để gửi các gói dữ kiện. Ví dụ như đối với các dạng bus và ring thì chỉ có một đường truyền duy nhất nối các trạm với nhau, cho nên cần phải có các quy tắc chung cho tất cả các trạm nối vào mạng để đảm bảo rằng đường truyền được truy nhập và sử dụng một cách hợp lý.

Có nhiều giao thức khác nhau để truy nhập đường truyền vật lý nhưng phân thành hai loại: các giao thức truy nhập ngẫu nhiên và các giao thức truy nhập có điều khiển.

VII.1. Giao thức chuyển mạch (yêu cầu và chấp nhận)

Giao thức chuyển mạch là loại giao thức hoạt động theo cách thức sau: một máy tính của mạng khi cần có thể phát tín hiệu thâm nhập vào mạng, nếu vào lúc này đường cáp không bận thì mạch điều khiển sẽ cho trạm này thâm nhập vào đường cáp còn nếu đường cáp đang bận, nghĩa là đang có giao lưu giữa các trạm khác, thì việc thâm nhập sẽ bị từ chối.

VII.2. Giao thức đường dây đa truy cập với cảm nhận va chạm

Giao thức đường dây đa truy cập (Carrier Sense Multiple Access with Collision Detection hay CSMA/CD) cho phép nhiều trạm thâm nhập cùng một lúc vào mạng, giao thức này thường dùng trong sơ đồ mạng dạng đường thẳng. Mọi trạm đều có thể được truy nhập vào đường dây chung một cách ngẫu nhiên và do vậy có thể dẫn đến xung đột (hai hoặc nhiều trạm đồng thời cùng truyền dữ liệu). Các trạm phải kiểm tra đường truyền gói dữ liệu đi qua có phải của nó hay không. Khi một trạm muốn truyền dữ liệu nó phải kiểm tra đường truyền xem có rảnh hay không để gửi gói dữ liệu của, nếu đường truyền đang bận trạm phải chờ đợi chỉ được truyền khi thấy đường truyền rảnh. Nếu cùng một lúc có hai trạm cùng sử dụng đường truyền thì giao thức phải phát hiện điều này và các trạm phải ngưng thâm nhập, chờ đợi lần sau các thời gian ngẫu nhiên khác nhau.

Khi đường cáp đang bận trạm phải chờ đợi theo một trong ba phương thức sau:

- Trạm tạm chờ đợi một thời gian ngẫu nhiên nào đó rồi lại bắt đầu kiểm tra đường truyền.
- Trạm tiếp tục kiểm tra đường truyền đến khi đường truyền rảnh thì truyền dữ liệu đi.
- Trạm tiếp tục kiểm tra đường truyềnđến khi đường truyền rảnh thì truyền dữ liệu đi với xác suất p xác định trước ($0 < p < 1$).

Tại đây phương thức 1 có hiệu quả trong việc tránh xung đột vì hai trạm cần truyền khi thấy đường truyền bận sẽ cùng rút lui và chờ đợi trong các thời gian ngẫu nhiên khác nhau. Ngược lại phương thức 2 cố gắng giảm thời gian trống của đường truyền bằng các cho phép trạm có thể truyền ngay sau khi một cuộc truyền kết thúc song nếu lúc đó có thêm một trạm khác đang đợi thì khả năng xảy ra xung đột là rất cao. Phương thức 3 với giá trị p phải lựa chọn hợp lý có thể tối thiểu hóa được khả năng xung đột lẫn thời gian trống của đường truyền.

Khi lưu lượng các gói dữ liệu cần di chuyển trên mạng quá cao, thì việc đụng độ có thể xảy ra với số lượng lớn có gây tắc nghẽn đường truyền dẫn đến làm chậm tốc độ truyền tin của hệ thống.

VII.3. Giao thức dùng thẻ bài vòng (Token ring)

Đây là giao thức truy nhập có điều khiển chủ yếu dùng kỹ thuật chuyển thẻ bài (token) để cấp phát quyền truy nhập đường truyền tức là quyền được truyền dữ liệu đi. Thẻ bài ở đay là một đơn vị dữ liệu đặc biệt, có kích thước và nội dung (gồm các thông tin điều khiển) được quy định riêng cho mỗi giao thức. Theo giao thức dùng thẻ bài vòng trong đường cáp liên tục có một thẻ bài chạy quanh trong mạng Thẻ bài là một đơn vị dữ liệu đặc biệt trong đó có một bit biểu diễn trạng thái sử dụng của nó (bận hoặc rỗi). Một trạm muốn truyền dữ liệu thì phải đợi đến khi nhận được một thẻ bài rảnh. Khi đó trạm sẽ đổi bit trạng thái của thẻ bài thành bận, nép gói dữ liệu có kèm theo địa chỉ nơi nhận vào thẻ bài và truyền đi theo chiều của vòng.

Vì thẻ bài chạy vòng quang trong mạng kín và chỉ có một thẻ nên việc đụng độ dữ liệu không thể xẩy ra, do vậy hiệu suất truyền dữ liệu của mạng không thay đổi.

Trong các giao thức này cần giải quyết hai vấn đề có thể dẫn đến phá vỡ hệ thống. Một là việc mất thẻ bài làm cho trên vòng không còn thẻ bài lưu chuyển nữa. Hai là một thẻ bài bận lưu chuyển không dừng trên vòng.

VII.4. Giao thức dùng thẻ bài cho dạng đường thẳng (Token bus)

Đây là giao thức truy nhập có điều khiển trong để cấp phát quyền truy nhập đường truyền cho các trạm đang có nhu cầu truyền dữ liệu, một thẻ bài được lưu chuyển trên một vòng logic thiết lập bởi các trạm đó. Khi một trạm có thẻ bài thì nó có quyền sử dụng đường truyền trong một thời gian xác định trước. Khi đã hết dữ liệu hoặc hết thời đoạn cho phép, trạm chuyển thẻ bài đến trạm tiếp theo trong vòng logic.

Như vậy trong mạng phải thiết lập được vòng logic (hay còn gọi là vòng ảo) bao gồm các trạm đang hoạt động nối trong mạng được xác định vị trí theo một chuỗi thứ tự mà trạm cuối cùng của chuỗi sẽ tiếp liền sau bởi trạm đầu tiên. Mỗi trạm được biết địa chỉ của các trạm kề trước và sau nó trong đó thứ tự của các trạm trên vòng logic có thể độc lập với thứ tự vật lý. Cùng với việc thiết lập vòng thì giao thức phải luôn luôn theo dõi sự thay đổi theo trạng thái thực tế của mạng.

VIII. Các phương tiện kết nối mạng liên khu vực (WAN)

Bên cạnh phương pháp sử dụng đường điện thoại thuê bao để kết nối các mạng cục bộ hoặc mạng khu vực với nhau hoặc kết nối vào Internet, có một số phương pháp khác:

- **Đường thuê bao (leased line)**. Đây là phương pháp cũ nhất, là phương pháp truyền thống nhất cho sự nối kết vĩnh cửu. Bạn thuê đường dây từ công ty điện thoại (trực tiếp hoặc qua nhà cung cấp dịch vụ). Bạn cần phải cài đặt một "*Chanel Service Unit*" (CSU) để nối đến mạng T, và một "*Digital Service Unit*" (DSU) để nối đến mạng chủ (primary) hoặc giao diện mạng.
- **ISDN (Integrated Service Digital Nework).** Sử dụng đường điện thoại số thay vì đường tương tự. Do ISDN là mạng dùng tín hiệu số, bạn không phải dùng một modem để nối với đường dây mà thay vào đó bạn phải dùng một thiết bị gọi là "*codec*" với modem có khả năng chạy ở 14.4 kbit/s. ISDN thích hợp cho cả hai trường hợp cá nhân và tổ chức. Các tổ chức có thể quan tâm hơn đến ISDN có khả năng cao hơn ("*primary*" ISDN) với tốc độ tổng cộng bằng tốc độ 1.544 Mbit/s của đường T1. Cước phí khi sử dụng ISDN được tính theo thời gian, một số trường hợp tính theo lượng dữ liệu được truyền đi và một số thì tính theo cả hai.

- **CATV link**. Công ty dẫn cáp trong khu vực của bạn có thể cho bạn thuê một "chỗ" trên đường cáp của họ với giá hấp dẫn hơn với đường điện thoại. Cần phải biết những thiết bị gì cần cho hệ thống của mình và độ rộng của dải mà bạn sẽ được cung cấp là bao nhiêu. Cũng như việc đóng góp chi phí với những khách hàng khác cho kênh liên lạc đó là như thế nào. Một dạng kỳ lạ hơn được đưa ra với tên gọi là mạng "lai" ("*hybrid" Network*), với một kênh CATV được sử dụng để lưu thông theo một hướng và một đường ISDN hoặc gọi số sử dụng cho đường trở lại. Nếu muốn cung cấp thông tin trên Internet, bạn phải xác định chắc chắn rằng "kênh ngược" của bạn đủ khả năng phục vụ cho nhu cầu thông tin của khách hàng của bạn.
- **Frame relay.** Frame relay "uyển chuyển" hơn đường thuê bao. Khách hàng thuê đường Frame relay có thể mua một dịch vụ có mức độ xác định - một "tốc độ thông tin uỷ thác" ("*Committed Information Rale" - CIR*). Nếu như nhu cầu của bạn trên mạng là rất "bột phát" (*burty*), hay người sử dụng của bạn có nhu cầu cao trên đường liên lạc trong suốt một khoảng thời gian xác định trong ngày, và có ít hoặc không có nhu cầu vào ban đêm - Frame relay có thể sẽ kinh tế hơn là thuê hoàn toàn một đường T1 (hoặc T3). Nhà cung cấp dịch vụ của bạn có thể đưa ra một phương pháp tương tự như là phương pháp thay thế đó là *Switched Multimegabit Data Service.*
- **Chế độ truyền không đồng bộ (Asynchoronous Trangfer Mode - ATM)**. ATM là một phương pháp tương đối mới đầu tiên báo hiệu cùng một kỹ thuật cho mạng cục bộ và liên khu vực. ATM thích hợp cho *real-time multimedia* song song với truyền dữ liệu truyền thống. ATM hứa hẹn sẽ trở thành một phần lớn của mạng tương lai.
- **Đường vi sóng (Microware links)**. Nếu cần kết nối vĩnh viễn đến nhà cung cấp dịch vụ nhưng lại thấy rằng đường thuê bao hay những lựa chọn khác là quá đắt, bạn sẽ thấy *microware* như là một lựa chọn thích hợp. Bạn không cần trả quá đắt cho cách này của *microware*, tuy nhiên bạn cần phải đầu tư nhiều tiền hơn vào lúc đầu, và bạn sẽ gặp một số rủi ro như tốc độ truyền đến mạng của bạn quá nhanh.
- **Đường vệ tinh (satellite links).** Nếu bạn muốn được chuyển một lượng lớn dữ liệu đặc biệt là từ những địa điểm từ xa thì đường vệ tinh là câu trả lời. Tầm hoạt động của những vệ tinh cùng vị trí địa lý với trái đất cũng tạo ra một sự chậm trễ (hoặc "bị che dấu") mà những người sử dụng Telnet có thể cảm nhận được.

Chương III Địa chỉ IP

I. Giao thức TCP/IP

Giao thức TCP/IP được phát triển từ mạng ARPANET và Internet và được dùng như giao thức mạng và vận chuyển trên mạng Internet. TCP (Transmission Control Protocol) là giao thức thuộc tầng vận chuyển và IP (Internet Protocol) là giao thức thuộc tầng mạng của mô hình OSI. Họ giao thức TCP/IP hiện nay là giao thức được sử dụng rộng rãi nhất để liên kết các máy tính và các mạng.

Hiện nay các máy tính của hầu hết các mạng có thể sử dụng giao thức TCP/IP để liên kết với nhau thông qua nhiều hệ thống mạng với kỹ thuật khác nhau. Giao thức TCP/IP thực chất là một họ giao thức cho phép các hệ thống mạng cùng làm việc với nhau thông qua việc cung cấp phương tiện truyền thông liên mạng.

II. Địa chỉ IP

II.1. Tổng quát

Nhiệm vụ chính của giao thức IP là cung cấp khả năng kết nối các mạng con thành liên kết mạng để truyền dữ liệu, vai trò của IP là vai trò của giao thức tầng mạng trong mô hình OSI. Giao thức IP là một giao thức kiểu không liên kết (connectionlees) có nghĩa là không cần có giai đoạn thiết lập liên kết trước khi truyền dữ liệu.

Sơ đồ địa chỉ hóa để định danh các trạm (host) trong liên mạng được gọi là địa chỉ IP 32 bits (32 bit IP address). Mỗi giao diện trong 1 máy có hỗ trợ giao thức IP đều phải được gán 1 địa chỉ IP (một máy tính có thể gắn với nhiều mạng do vậy có thể có nhiều địa chỉ IP). Địa chỉ IP gồm 2 phần: địa chỉ mạng (netid) và địa chỉ máy (hostid). Mỗi địa chỉ IP có độ dài 32 bits được tách thành 4 vùng (mỗi vùng 1 byte), có thể biểu thị dưới dạng thập phân, bát phân, thập lục phân hay nhị phân. Cách viết phổ biến nhất là dùng ký pháp thập phân có dấu chấm (dotted decimal notation) để tách các vùng. Mục đích của địa chỉ IP là để định danh duy nhất cho một máy tính bất kỳ trên liên mạng.

Do tổ chức và độ lớn của các mạng con (subnet) của liên mạng có thể khác nhau, người ta chia các địa chỉ IP thành 5 lớp, ký hiệu là A, B, C, D và E. Trong lớp A, B, C chứa địa chỉ có thể gán được. Lớp D dành riêng cho lớp kỹ thuật multicasting. Lớp E được dành những ứng dụng trong tương lai.

Netid trong địa chỉ mạng dùng để nhận dạng từng mạng riêng biệt. Các mạng liên kết phải có địa chỉ mạng (netid) riêng cho mỗi mạng. Ở đây các bit đầu tiên của byte đầu tiên được dùng để định danh lớp địa chỉ (0 - lớp A, 10 - lớp B, 110 - lớp C, 1110 - lớp D và 11110 - lớp E).

Ở đây ta xét cấu trúc của các lớp địa chỉ có thể gán được là lớp A, lớp B, lớp C.

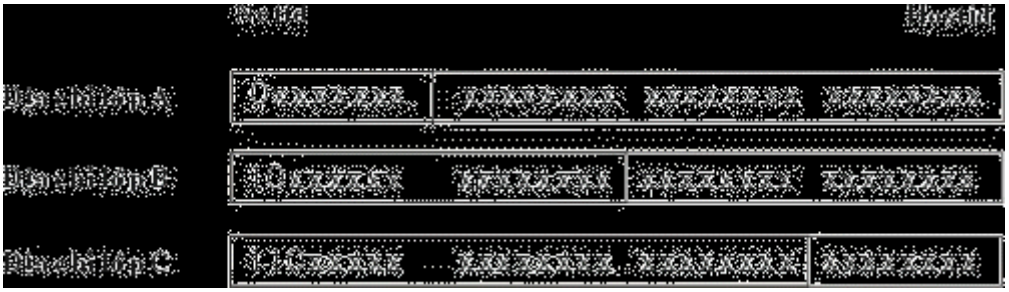
II.2. Cấu trúc của các địa chỉ IP

Mạng lớp A: địa chỉ mạng (netid) là 1 Byte và địa chỉ host (hostid) là 3 byte.
Mạng lớp B: địa chỉ mạng (netid) là 2 Byte và địa chỉ host (hostid) là 2 byte.
Mạng lớp C: địa chỉ mạng (netid) là 3 Byte và địa chỉ host (hostid) là 1 byte.

Lớp A cho phép định danh tới 126 mạng, với tối đa 16 triệu host trên mỗi mạng. Lớp này được dùng cho các mạng có số trạm cực lớn.

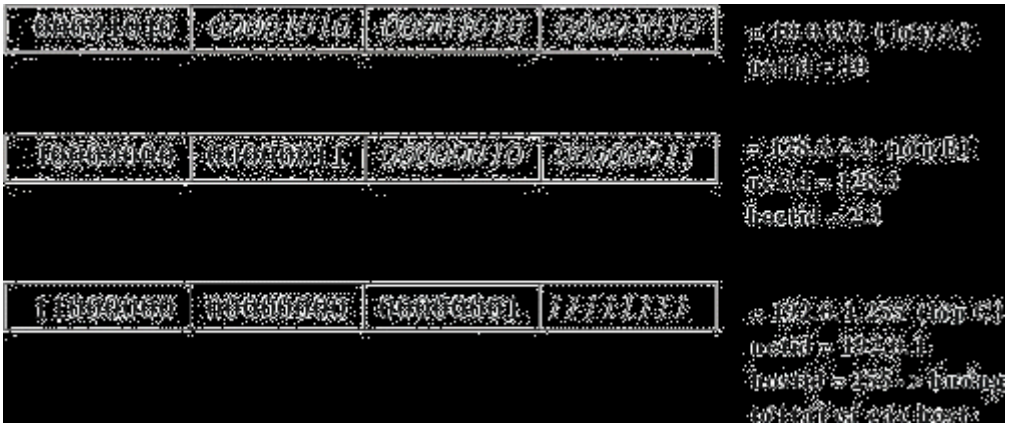
Lớp B cho phép định danh tới 16384 mạng, với tối đa 65534 host trên mỗi mạng.

Lớp C cho phép định danh tới 2 triệu mạng, với tối đa 254 host trên mỗi mạng. Lớp này được dùng cho các mạng có ít trạm.



Hình III-1: Cấu trúc các lớp địa chỉ IP

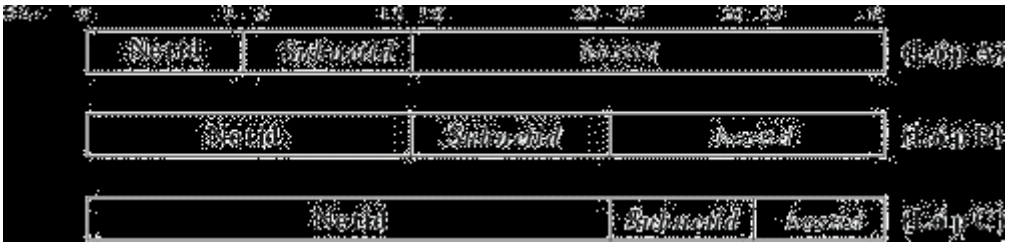
Một số địa chỉ có tính chất đặc biệt: Một địa chỉ có hostid = 0 được dùng để hướng tới mạng định danh bởi vùng netid. Ngược lại, một địa chỉ có vùng hostid gồm toàn số 1 được dùng để hướng tới tất cả các host nối vào mạng netid, và nếu vùng netid cũng gồm toàn số 1 thì nó hướng tới tất cả các host trong liên mạng



Hình III-2: Ví dụ cấu trúc các lớp địa chỉ IP

Cần lưu ý rằng các địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, và chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm trên đó một mạng cục bộ (Ethernet, Token Ring.).

Trong nhiều trường hợp, một mạng có thể được chia thành nhiều mạng con (subnet), lúc đó có thể đưa thêm các vùng subnetid để định danh các mạng con. Vùng subnetid được lấy từ vùng hostid, cụ thể đối với lớp A, B, C như ví dụ sau:



Hình III-3: Ví dụ địa chỉ khi bổ sung vùng subnetid

Đơn vị dữ liệu dùng trong IP được gọi là gói tin (datagram), có khuôn dạng

Hình III-4: Dạng thức của gói tin IP

Ý nghĩa của thông số như sau:

- *VER* (4 bits)**:** chỉ version hiện hành của giao thức IP hiện được cài đặt, Việc có chỉ số version cho phép có các trao đổi giữa các hệ thống sử dụng version cũ và hệ thống sử dụng version mới.

- *IHL* (4 bits): chỉ độ dài phần đầu (Internet header Length) của gói tin datagram, tính theo đơn vị từ (32 bits). Trường này bắt buột phải có vì phần đầu IP có thể có độ dài thay đổi tùy ý. Độ dài tối thiểu là 5 từ (20 bytes), độ dài tối đa là 15 từ hay là 60 bytes.

- *Type of service* (8 bits): đặc tả các tham số về dịch vụ nhằm thông báo cho mạng biết dịch vụ nào mà gói tin muốn được sử dụng, chẳng hạn ưu tiên, thời hạn chậm trễ, năng suất truyền và độ tin cậy. Hình sau cho biết ý nghĩ của trường 8 bits này.

- Precedence (3 bit): chỉ thị về quyền ưu tiên gửi datagram, nó có giá trị từ 0 (gói tin bình thường) đến 7 (gói tin kiểm soát mạng).

- D (Delay) (1 bit): chỉ độ trễ yêu cầu trong đó

  - D = 0 gói tin có độ trễ bình thường

  - D = 1 gói tin độ trễ thấp

- T (Throughput) (1 bit): chỉ độ thông lượng yêu cầu sử dụng để truyền gói tin với lựa chọn truyền trên đường thông suất thấp hay đường thông suất cao.

  - T = 0 thông lượng bình thường và

  - T = 1 thông lượng cao

- R (Reliability) (1 bit): chỉ độ tin cậy yêu cầu

  - R = 0 độ tin cậy bình thường

  - R = 1 độ tin cậy cao

- *Total Length* (16 bits): chỉ độ dài toàn bộ gói tin, kể cả phần đầu tính theo đơn vị byte với chiều dài tối đa là 65535 bytes. Hiện nay giới hạn trên là rất lớn nhưng trong tương lai với những mạng Gigabit thì các gói tin có kích thước lớn là cần thiết.

- *Identification* (16 bits): cùng với các tham số khác (như Source Address và Destination Address) tham số này dùng để định danh duy nhất cho một datagram trong khoảng thời gian nó vẫn còn trên liên mạng.

- *Flags (3 bits):* liên quan đến sự phân đoạn (fragment) các datagram, Các gói tin khi đi trên đường đi có thể bị phân thành nhiều gói tin nhỏ, trong trường hợp bị phân đoạn thì trường Flags được dùng điều khiển phân đoạn và tái lắp ghép bó dữ liệu. Tùy theo giá trị của Flags sẽ có ý nghĩa là gói tin sẽ không phân đoạn, có thể phân đoạn hay là gói tin phân đoạn cuối cùng. Trường ***Fragment Offset*** cho biết vị trí dữ liệu thuộc phân đoạn tương ứng với đoạn bắt đầu của gói dữ liệu gốc. Ý nghĩa cụ thể của trường Flags là:

  - bit 0: reserved - chưa sử dụng, luôn lấy giá trị 0.

  - bit 1: (DF) = 0 (May Fragment) = 1 (Don't Fragment)

  - bit 2: (MF) = 0 (Last Fragment) = 1 (More Fragments)

- *Fragment Offset (13 bits)*: chỉ vị trí của đoạn (fragment) ở trong datagram tính theo đơn vị 8 bytes, có nghĩa là phần dữ liệu mỗi gói tin (trừ gói tin cuối cùng) phải chứa một vùng dữ liệu có độ dài là bội số của 8 bytes. Điều này có ý nghĩa là phải nhân giá trị của Fragment offset với 8 để tính ra độ lệch byte.

- *Time to Live (8 bits)*: qui định thời gian tồn tại (tính bằng giây) của gói tin trong mạng để tránh tình trạng một gói tin bị quẩn trên mạng. Thời gian này được cho bởi trạm gửi và được giảm đi (thường qui ước là 1 đơn vị) khi datagram đi qua mỗi router của liên mạng. Thời lượng này giảm xuống tại mỗi router với mục đích giới hạn thời gian tồn tại của các gói tin và kết thúc những lần lặp lại vô hạn trên mạng. Sau đây là 1 số điều cần lưu ý về trường ***Time To Live*****:**

  - Nút trung gian của mạng không được gởi 1 gói tin mà trường này có giá trị= 0.

  - Một giao thức có thể ấn định ***Time To Live*** để thực hiện cuộc ra tìm tài nguyên trên mạng trong phạm vi mở rộng.

  - Một giá trị cố định tối thiểu phải đủ lớn cho mạng hoạt động tốt.

- *Protocol (8 bits):* chỉ giao thức tầng trên kế tiếp sẽ nhận vùng dữ liệu ở trạm đích (hiện tại thường là TCP hoặc UDP được cài đặt trên IP). Ví dụ: **TCP** có giá trị trường **Protocol** là 6, **UDP** có giá trị trường **Protocol** là 17

- *Header Checksum (16 bits)*: Mã kiểm soát lỗi của header gói tin IP.

- *Source Address (32 bits)*: Địa chỉ của máy nguồn.

- Destination Address (32 bits): địa chỉ của máy đích

- *Options (độ dài thay đổi)*: khai báo các lựa chọn do người gửi yêu cầu (tuỳ theo từng chương trình).

- *Padding (độ dài thay đổi)*: Vùng đệm, được dùng để đảm bảo cho phần header luôn kết thúc ở một mốc 32 bits.

- *Data (độ dài thay đổi)*: Trên một mạng cục bộ như vậy, hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. Như vậy vấn đề đặt ra là phải thực hiện ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý (48 bits) của một trạm.

III. Một số khái niệm và thuật ngữ liên quan

Địa chỉ mạng (Network Address): là địa chỉ IP mà giá trị của tất cả các bits ở phần nhận dạng máy tính đều là 0, được sử dụng để xác định một mạng.

Ví dụ : 10.0.0.0; 172.18.0.0 ; 192.1.1.0

Địa chỉ quảng bá (Broadcast Address) : Là địa chỉ IP mà giá trị của tất cả các bits ở phần nhận dạng máy tính đều là 1, được sử dụng để chỉ tất cả các máy tính trong mạng.

Ví dụ : 10.255.255.255, 172.18.255.255, 192.1.1.255

Mặt nạ mạng chuẩn (Netmask) : Là địa chỉ IP mà giá trị của các bits ở phần nhận dạng mạng đều là 1, các bits ở phần nhận dạng máy tính đều là 0. Như vậy ta có 3 mặt nạ mạng tương ứng cho 3 lớp mạng A, B và C là :

- Mặt nạ mạng lớp A : 255.0.0.0
- Mặt nạ mạng lớp B : 255.255.0.0
- Mặt nạ mạng lớp C : 255.255.255.0

Ta gọi chúng là các mặt nạ mạng mặc định (Default Netmask)

Lưu ý : Địa chỉ mạng, địa chỉ quảng bá, mặt nạ mạng không được dùng để đặt địa chỉ cho các máy tính

Địa chỉ mạng 127.0.0.0 là địa chỉ được dành riêng để đặt trong phạm vi một máy tính. Nó chỉ có giá trị cục bộ (trong phạm vi một máy tính). Thông thường khi cài đặt giao thức IP thì máy tính sẽ được gián địa chỉ 127.0.0.1. Địa chỉ này thông thường để kiểm tra xem giao thức IP trên máy hiện tại có hoạt động không.

Địa chỉ dành riêng cho mạng cục bộ không nối kết trực tiếp Internet: Các mạng cục bộ không nối kết trực tiếp vào mạng Internet có thể sử dụng các địa chỉ mạng sau để đánh địa chỉ cho các máy tính trong mạng của mình :
- Lớp A : 10.0.0.0
- Lớp B : 172.16.0.0 đến 172.32.0.0
- Lớp C : 192.168.0.0

Ý nghĩa của Netmask

Với một địa chỉ IP và một Netmask cho trước, ta có thể dùng phép toán AND BIT để tính ra được địa chỉ mạng mà địa chỉ IP này thuộc về. Công thức như sau :

Network Address = IP Address & Netmask

Ví dụ : Cho địa chỉ IP = 198.53.147.45 và Netmask = 255.255.255.0. Ta thực hiện phép toán AND BIT (&) hai địa chỉ trên:

	Biểu diễn thập phân	Biểu diễn nhị phân
IP Address	198.53.147.45	11000110 00110101 10010011 00101101
Netmask	255.255.255.0	11111111 11111111 11111111 00000000
Network Address	198.53.147.0	11000110 00110101 10010011 00000000

III.1. Các giao thức trong mạng IP

Để mạng với giao thức IP hoạt động được tốt người ta cần một số giao thức bổ sung, các giao thức này đều không phải là bộ phận của giao thức IP và giao thức IP sẽ dùng đến chúng khi cần.

- ***Giao thức ARP (Address Resolution Protocol):*** Ở đây cần lưu ý rằng các địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, và chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm trên đó một mạng cục bộ (Ethernet, Token Ring.). Trên một mạng cục bộ hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. Như vậy vấn đề đặt ra là phải tìm được ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý của một trạm. *Giao thức ARP* đã được xây dựng để tìm địa chỉ vật lý từ địa chỉ IP khi cần thiết.

- ***Giao thức RARP (Reverse Address Resolution Protocol)***: Là giao thức ngược với *giao thức ARP*. Giao thức RARP được dùng để tìm địa chỉ IP từ địa chỉ vật lý.

- ***Giao thức ICMP (Internet Control Message Protocol):*** Giao thức này thực hiện truyền các thông báo điều khiển (báo cáo về các tình trạng các lỗi trên mạng.) giữa các gateway hoặc một nút của liên mạng. Tình trạng lỗi có thể là: một gói tin IP không thể tới đích của nó, hoặc một router không đủ bộ nhớ đệm để lưu và chuyển một gói tin IP, Một thông báo ICMP được tạo và chuyển cho IP. IP sẽ "bọc" (encapsulate) thông báo đó với một IP header và truyền đến cho router hoặc trạm đích.

III.2. Các bước hoạt động của giao thức IP

Khi giao thức IP được khởi động nó trở thành một thực thể tồn tại trong máy tính và bắt đầu thực hiện những chức năng của mình, lúc đó thực thể IP là cấu thành của tầng mạng, nhận yêu cầu từ các tầng trên nó và gửi yêu cầu xuống các tầng dưới nó.

Đối với thực thể IP ở máy nguồn, khi nhận được một yêu cầu gửi từ tầng trên, nó thực hiện các bước sau đây:

- Tạo một IP datagram dựa trên tham số nhận được.
- Tính checksum và ghép vào header của gói tin.
- Ra quyết định chọn đường: hoặc là trạm đích nằm trên cùng mạng hoặc một gateway sẽ được chọn cho chặng tiếp theo.
- Chuyển gói tin xuống tầng dưới để truyền qua mạng.

Đối với router, khi nhận được một gói tin đi qua, nó thực hiện các động tác sau:

1) Tính chesksum, nếu sai thì loại bỏ gói tin.

2) Giảm giá trị tham số Time - to Live. nếu thời gian đã hết thì loại bỏ gói tin.

3) Ra quyết định chọn đường.

4) Phân đoạn gói tin, nếu cần.

5) Kiến tạo lại IP header, bao gồm giá trị mới của các vùng Time - to -Live, Fragmentation và Checksum.

6) Chuyển datagram xuống tầng dưới để chuyển qua mạng.

Cuối cùng khi một datagram nhận bởi một thực thể IP ở trạm đích, nó sẽ thực hiện bởi các công việc sau:

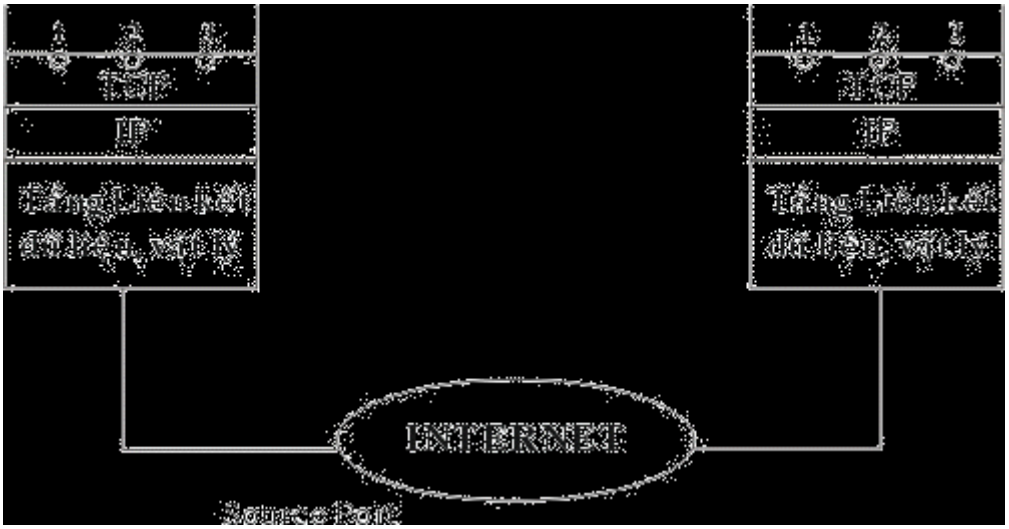
1) Tính checksum. Nếu sai thì loại bỏ gói tin.

2) Tập hợp các đoạn của gói tin (nếu có phân đoạn)

3) Chuyển dữ liệu và các tham số điều khiển lên tầng trên.

IV. Giao thức điều khiển truyền dữ liệu TCP

TCP là một giao thức "có liên kết" (connection - oriented), nghĩa là cần phải thiết lập liên kết giữa hai thực thể TCP trước khi chúng trao đổi dữ liệu với nhau. Một tiến trình ứng dụng trong một máy tính truy nhập vào các dịch vụ của giao thức TCP thông qua một cổng (port) của TCP. Số hiệu cổng TCP được thể hiện bởi 2 bytes.



Hình III-5: Cổng truy nhập dịch vụ TCP

Một cổng TCP kết hợp với địa chỉ IP tạo thành một đầu nối TCP/IP (socket) duy nhất trong liên mạng. Dịch vụ TCP được cung cấp nhờ một liên kết logic giữa một cặp đầu nối TCP/IP. Một đầu nối TCP/IP có thể tham gia nhiều liên kết với các đầu nối TCP/IP ở xa khác nhau. Trước khi truyền dữ liệu giữa 2 trạm cần phải thiết lập một liên kết TCP giữa chúng và khi không còn nhu cầu truyền dữ liệu thì liên kết đó sẽ được giải phóng.

Các thực thể của tầng trên sử dụng giao thức TCP thông qua các hàm gọi (function calls) trong đó có các hàm yêu cầu để yêu cầu, để trả lời. Trong mỗi hàm còn có các tham số dành cho việc trao đổi dữ liệu.

IV.1. Các bước thực hiện để thiết lập một liên kết TCP/IP:

Thiết lập một liên kết mới có thể được mở theo một trong 2 phương thức: chủ động (active) hoặc bị động (passive).

- Phương thức bị động, người sử dụng yêu cầu TCP chờ đợi một yêu cầu liên kết gửi đến từ xa thông qua một đầu nối TCP/IP (tại chỗ). Người sử dụng dùng hàm passive Open có khai báo cổng TCP và các thông số khác (mức ưu tiên, mức an toàn)

- Với phương thức chủ động, người sử dụng yêu cầu TCP mở một liên kết với một một đầu nối TCP/IP ở xa. Liên kết sẽ được xác lập nếu có một hàm Passive Open tương ứng đã được thực hiện tại đầu nối TCP/IP ở xa đó.

Bảng III-1 Lệt kê một vài cổng TCP phổ biến

Số hiệu cổng	Mô tả
0	Reserved
5	Remote job entry
7	Echo

9	Discard
11	Systat
13	Daytime
15	Nestat
17	Quotd (quote odd day)
20	ftp-data
21	ftp (control)
23	Telnet
25	SMTP
37	Time
53	Name Server
102	ISO - TSAP
103	X.400
104	X.400 Sending
111	Sun RPC
139	Net BIOS Session source
160 - 223	Reserved

Khi người sử dụng gửi đi một yêu cầu mở liên kết sẽ được nhận hai thông số trả lời từ TCP.

- Thông số Open ID được TCP trả lời ngay lập tức để gán cho một liên kết cục bộ (local connection name) cho liên kết được yêu cầu. Thông số này về sau được dùng để tham chiếu tới liên kết đó. (Trong trường hợp nếu TCP không thể thiết lập được liên kết yêu cầu thì nó phải gửi tham số Open Failure để thông báo.)

- Khi TCP thiết lập được liên kết yêu cầu nó gửi tham số Open Sucsess được dùng để thông báo liên kết đã được thiết lập thành công. Thông báo này dược chuyển đến trong cả hai trường hợp bị động và chủ động. Sau khi một liên kết được mở, việc truyền dữ liệu trên liên kết có thể được thực hiện.

IV.2. Các bước thực hiện khi truyền và nhận dữ liệu

Sau khi xác lập được liên kết người sữ dụng gửi và nhận dữ liệu. Việc gửi và nhận dữ liệu thông qua các hàm Send và receive.

- *Hàm Send:* Dữ liệu được gửi xuống TCP theo các khối (block). Khi nhận được một khối dữ liệu, TCP sẽ lưu trữ trong bộ đệm (buffer). Nếu cờ PUSH được dựng thì toàn bộ dữ liệu trong bộ đệm được gửi, kể cả khối dữ liệu mới đến sẽ được gửi đi. Ngược lại cờ PUSH không được dựng thì dữ liệu được giữ lại trong bộ đệm và sẽ gửi đi khi có cơ hội thích hợp (chẳng hạn chờ thêm dữ liệu nữa để gữi đi với hiệu quả hơn).

- *Hàm reveive*: Ở trạm đích dữ liệu sẽ được TCP lưu trong bộ đệm gắn với mỗi liên kết. Nếu dữ liệu được đánh dấu với một cờ PUSH thì toàn bộ dữ liệu trong bộ đệm (kể cả các dữ liệu được lưu từ trước) sẽ được chuyển lên cho người sữ dụng. Còn nếu dữ liệu đến không được đánh dấu với cờ PUSH thì TCP chờ tới khi thích hợp mới chuyển dữ liệu với mục tiêu tăng hiệu quả hệ thống.

Nói chung việc nhận và giao dữ liệu cho người sử dụng đích của TCP phụ thuộc vào việc cài đặt cụ thể. Trường hợp cần chuyển gấp dữ liệu cho người sử dụng thì có thể dùng cờ URGENT và đánh dấu dữ liệu bằng bit URG để báo cho người sử dụng cần phải sử lý khẩn cấp dữ liệu đó.

IV.3. Các bước thực hiện khi đóng một liên kết

Việc đóng một liên kết khi không cần thiết được thực hiên theo một trong hai cách: *dùng hàm Close* hoặc dùng *hàm Abort*.

- *Hàm Close*: Yêu cầu đóng liên kết một cách bình thường. Có nghĩa là việc truyền dữ liệu trên liên kết đó đã hoàn tất. Khi nhận được một *hàm Close* TCP sẽ truyền đi tất cả dữ liệu còn trong bộ đệm thông báo rằng nó đóng liên kết. Lưu ý rằng khi một người sử dụng đã gửi đi một *hàm Close* thì nó vẫn phải tiếp tục nhận dữ liệu đến trên liên kết đó cho đến khi TCP đã báo cho phía bên kia biết về việc đóng liên kết và chuyển giao hết tất cả dữ liệu cho người sử dụng của mình.

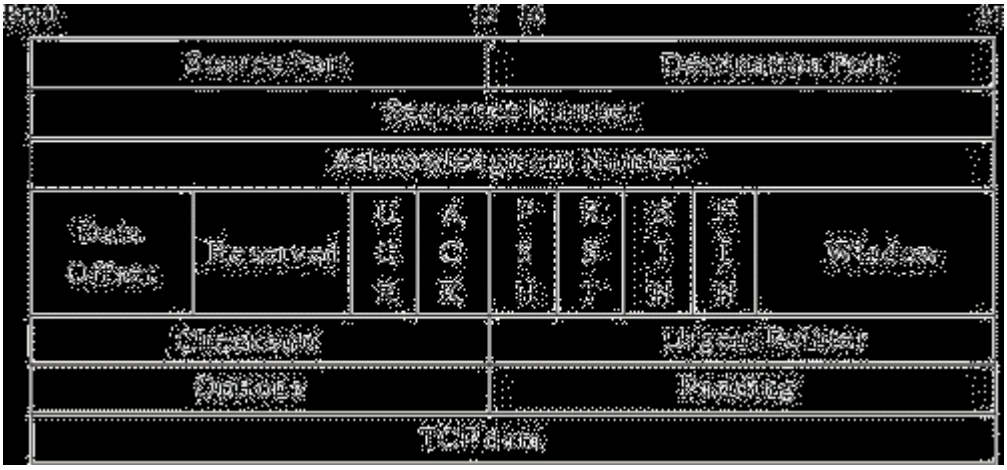
- *Hàm Abort*: Người sử dụng có thể đóng một liên kết bất và sẽ không chấp nhận dữ liệu qua liên kết đó nữa. Do vậy dữ liệu có thể bị mất đi khi đang được truyền đi. TCP báo cho TCP ở xa biết rằng liên kết đã được hủy bỏ và TCP ở xa sẽ thông báo cho người sử dụng cũa mình.

IV.4. Một số hàm khác của TCP

- *Hàm Status*: cho phép người sử dụng yêu cầu cho biết trạng thái của một liên kết cụ thể, khi đó TCP cung cấp thông tin cho người sử dụng.

- *Hàm Error*: thông báo cho người sử dụng TCP về các yêu cầu dịch vụ bất hợp lệ liên quan đến một liên kết có tên cho trước hoặc về các lỗi liên quan đến môi trường.

Đơn vị dữ liệu sử dụng trong TCP được gọi là segment (đoạn dữ liệu), có các tham số với ý nghĩa như sau:



Hình III-6: Dạng thức của segment TCP

- Source Por (16 bits): Số hiệu cổng TCP của trạm nguồn.
- Destination Port (16 bit): Số hiệu cổng TCP của trạm đích.
- Sequence Number (32 bit): số hiệu của byte đầu tiên của segment trừ khi bit SYN được thiết lập. Nếy bit SYN được thiết lập thì Sequence Number là số hiệu tuần tự khởi đầu (ISN) và byte dữ liệu đầu tiên là ISN+1.
- Acknowledgment Number (32 bit): số hiệu của segment tiếp theo mà trạm nguồn đang chờ để nhận. Ngầm ý báo nhận tốt (các) segment mà trạm đích đã gửi cho trạm nguồn.
- Data offset (4 bit): số lượng bội của 32 bit (32 bit words) trong TCP header (tham số này chỉ ra vị trí bắt đầu của nguồn dữ liệu).
- Reserved (6 bit): dành để dùng trong tương lai
- Control bit (các bit điều khiển):
  - URG: Vùng con trỏ khẩn (Ucgent Poiter) có hiệu lực.
  - ACK: Vùng báo nhận (ACK number) có hiệu lực.
  - PSH: Chức năng PUSH.
  - RST: Khởi động lại (reset) liên kết.
  - SYN: Đồng bộ hóa số hiệu tuần tự (sequence number).
  - FIN: Không còn dữ liệu từ trạm nguồn.
- Window (16 bit): cấp phát credit để kiểm soát nguồn dữ liệu (cơ chế cửa sổ). Đây chính là số lượng các byte dữ liệu, bắt đầu từ byte được chỉ ra trong vùng ACK number, mà trạm nguồn đã sẵn sàng để nhận.
- Checksum (16 bit): mã kiểm soát lỗi cho toàn bộ segment (header + data)

- Urgemt Poiter (16 bit): con trỏ này trỏ tới số hiệu tuần tự của byte đi theo sau dữ liệu khẩn. Vùng này chỉ có hiệu lực khi bit URG được thiết lập.

- Options (độ dài thay đổi): khai báo các option của TCP, trong đó có độ dài tối đa của vùng TCP data trong một segment.

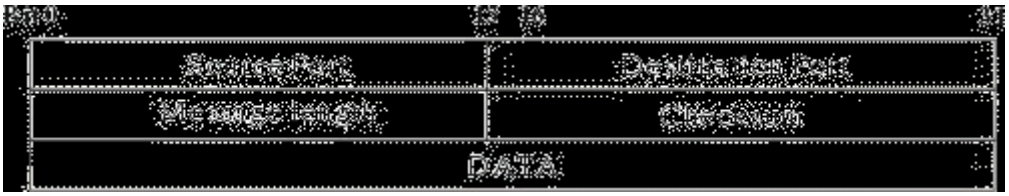
- Paddinh (độ dài thay đổi): phần chèn thêm vào header để đảm bảo phần header luôn kết thúc ở một mốc 32 bit. Phần thêm này gồm toàn số 0.

- TCP data (độ dài thay đổi): chứa dữ liệu của tầng trên, có độ dài tối đa ngầm định là 536 byte. Giá trị này có thể điều chỉnh bằng cách khai báo trong vùng options.

V. Giao thức UDP (User Datagram Protocol)

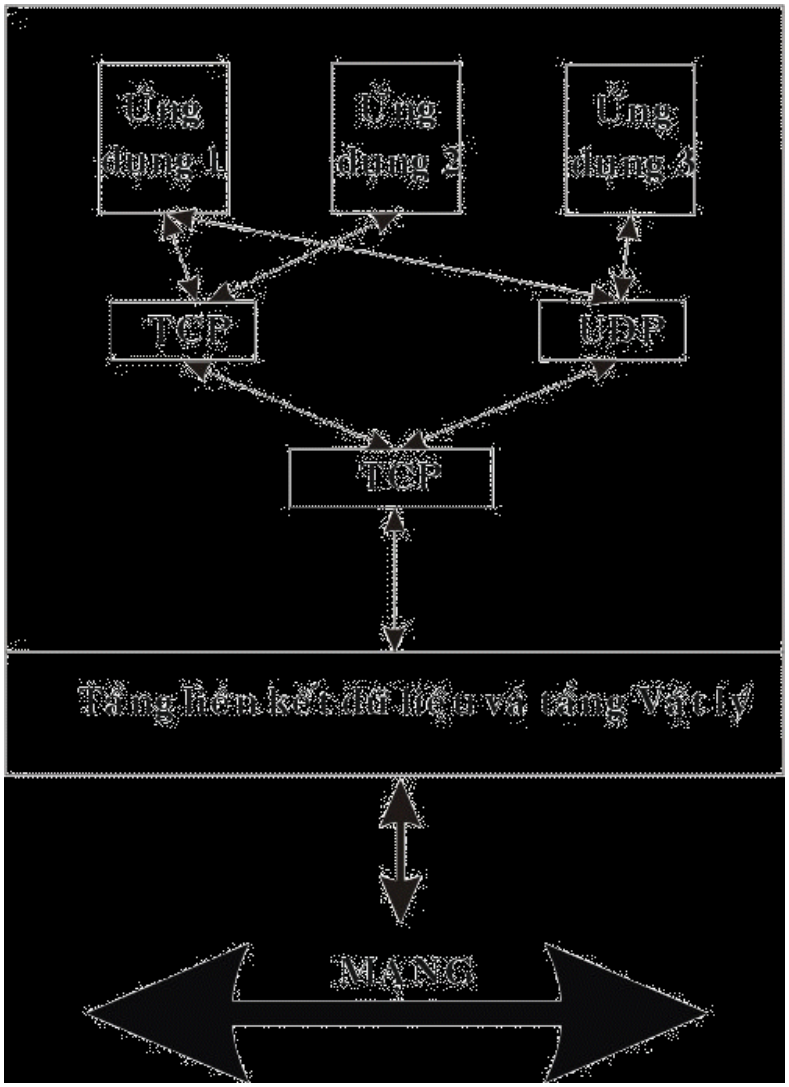
UDP (User Datagram Protocol) là giao thức theo phương thức không liên kết được sử dụng thay thế cho TCP ở trên IP theo yêu cầu của từng ứng dụng. Khác với TCP, UDP không có các chức năng thiết lập và kết thúc liên kết. Tương tự như IP, nó cũng không cung cấp cơ chế báo nhận (acknowledgment), không sắp xếp tuần tự các gói tin (datagram) đến và có thể dẫn đến tình trạng mất hoặc trùng dữ liệu mà không có cơ chế thông báo lỗi cho người gửi. Qua đó ta thấy UDP cung cấp các dịch vụ vận chuyển không tin cậy như trong TCP.

Khuôn dạng UDP datagram được mô tả với các vùng tham số đơn giản hơn nhiều so với TCP segment.



Hình III-7: Dạng thức của gói tin UDP

UDP cũng cung cấp cơ chế gán và quản lý các số hiệu cổng (port number) để định danh duy nhất cho các ứng dụng chạy trên một trạm của mạng. Do ít chức năng phức tạp nên UDP thường có xu thế hoạt động nhanh hơn so với TCP. Nó thường được dùng cho các ứng không đòi hỏi độ tin cậy cao trong giao vận.



Hình III-8: Mô hình quan hệ họ giao thức TCP/IP

VI. Địa chỉ IPv4

VI.1. Thành phần và hình dạng của địa chỉ IP

Địa chỉ IP đang được sử dụng hiện tại (IPv4) có 32 bit chia thành 4 Octet (mỗi Octet có 8 bit, tương đương 1 byte) cách đếm đều từ trái qua phải bit 1 cho đến bit 32, các Octet tách biệt nhau bằng dấu chấm (.), bao gồm có 3 thành phần chính.

Bit 1.....Bit 32

* Bit nhận dạng lớp (Class bit)

* Địa chỉ của đường mạng (Net ID)

* Địa chỉ của máy tính (Host ID).

Bit nhận dạng lớp (Class bit) để phân biệt địa chỉ ở lớp nào.

1. Địa chỉ Internet biểu hiện ở dạng bit nhị phân:

x y x y x y x y. x y x y x y x y. x y x y x y x y. x y x y x y x y

x, y = 0 hoặc 1.

Ví dụ:

0	**0 1 0 1 10.**	**0 1 1 1 1 0 1 1.**	**0 1 1 0 1 1 1 0.**	**1 1 1 0 0 0 0 0**
Bit nhận dạng	Octet 1	Octet 2	Octet 3	Octet 4

2. Địa chỉ Internet biểu hiện ở dạng thập phân:

xxx.xxx.xxx.xxx

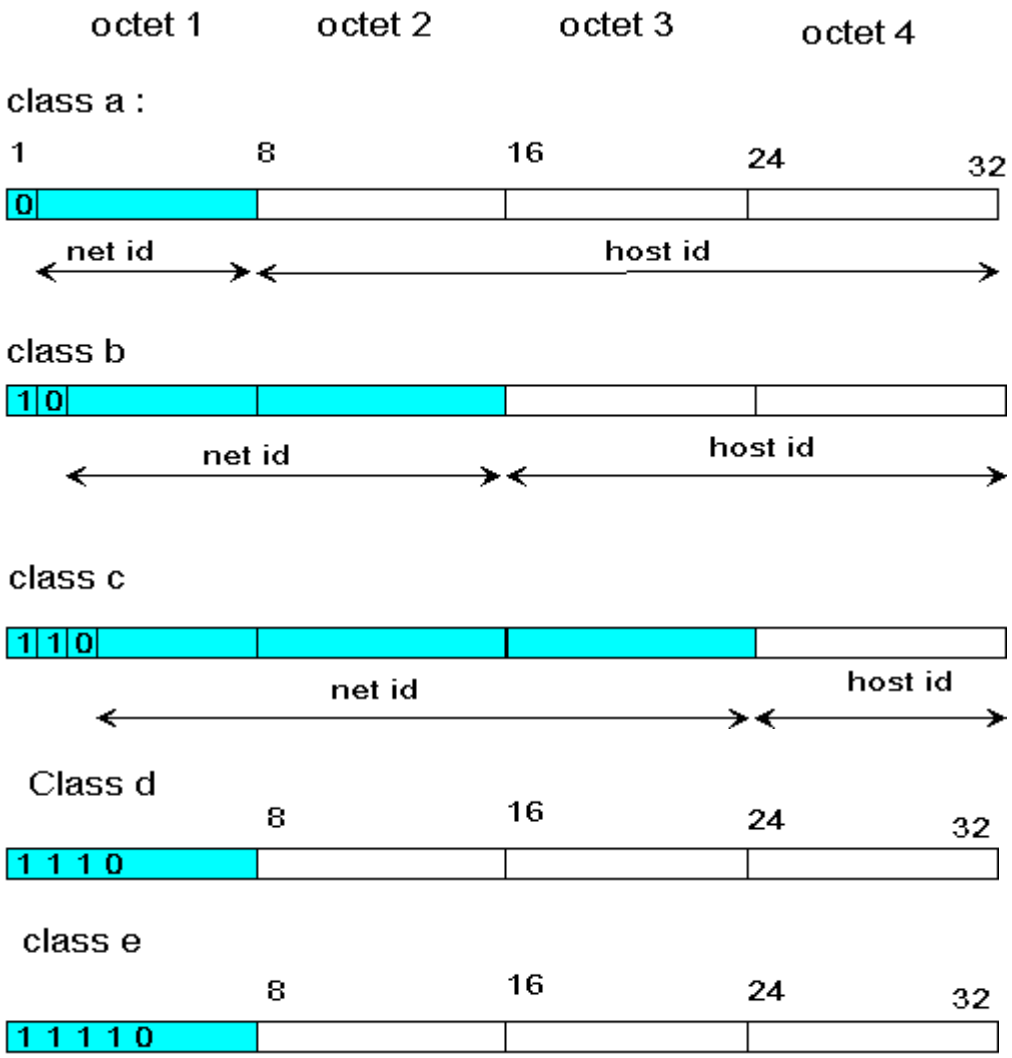
x là số thập phân từ 0 đến 9

Ví dụ: 146. 123. 110. 224

Dạng viết đầy đủ của địa chỉ IP là 3 con số trong từng Octet. Ví dụ: địa chỉ IP thường thấy trên thực tế có thể là 53.143.10.2 nhưng dạng đầy đủ là 053.143.010.002.

VI.2. Các lớp địa chỉ IP

Địa chỉ IP chia ra 5 lớp A,B,C, D, E. Hiện tại đã dùng hết lớp A,B và gần hết lớp C, còn lớp D và E Tổ chức internet đang để dành cho mục đích khác không phân, nên chúng ta chỉ nghiên cứu 3 lớp đầu.



Qua cấu trúc các lớp địa chỉ IP chúng ta có nhận xét sau:

* Bit nhận dạng là những bit đầu tiên - của lớp A là 0, của lớp B là 10, của lớp C là 110.

* Lớp D có 4 bit đầu tiên để nhận dạng là 1110, còn lớp E có 5 bít đầu tiên để nhận dạng là 11110.

* Địa chỉ lớp A: Địa chỉ mạng ít và địa chỉ máy chủ trên từng mạng nhiều.

* Địa chỉ lớp B: Địa chỉ mạng vừa phải và địa chỉ máy chủ trên từng mạng vừa phải.

* Địa chỉ lớp C: Địa chỉ mạng nhiều, địa chỉ máy chủ trên từng mạng ít.

Địa chỉ lớp	Vùng địa chỉ lý thuyết	Số mạng tối đa sử dụng	Số máy chủ tối đa trên từng mạng
A	Từ 0.0.0.0 đến 127.0.0.0	126	16777214
B	Từ 128.0.0.0 đến 191.255.0.0	16382	65534
C	Từ 192.0.0.0 đến 223.255.255.0	2097150	254

D	Từ 224.0.0.0 đến 240.0.0.0	Không phân	
E	Từ 241.0.0.0 đến 255.0.0.0	Không phân	

Bảng III-2 Các lớp địa chỉ IP

Địa chỉ lớp	Vùng địa chỉ sử dụng	Bit nhận dạng	Số bit dùng để phân cho mạng
A	Từ 1 đến 127	0	7
B	Từ 128.1 đến 191.254	10	14
C	Từ 192.0.1 đến 223.255.254	110	21
D		1110	---
E		11110	---

Bảng III-3 Bit nhận dạng các lớp

Như vậy nếu chúng ta thấy 1 địa chỉ IP có 4 nhóm số cách nhau bằng dấu chấm, nếu thấy nhóm số thứ nhất nhỏ hơn 126 biết địa chỉ này ở lớp A, nằm trong khoảng 128 đến 191 biết địa chỉ này ở lớp B và từ 192 đến 223 biết địa chỉ này ở lớp C.

VII. IPv6

VII.1. Giao thức liên mạng thế hệ mới (IPv6)

Giao thức IPv4 đã được coi là nền tảng cho mạng Internet với những tính chất ưu việt của nó, tuy nhiên với sự bùng nổ về Internet giao thức IPv4 đã bộc lộ một số yếu điểm về tính năng, trong đó nổi bật là:

- Thiếu hụt về tính năng xác thực, an ninh của gói tin trên mạng. Khả năng mở rộng hạn chế.
- Thiếu hụt không gian địa chỉ. Với sự phát triển của mạng Internet, không gian địa chỉ IP có thể sử dụng thực sự là rất nhỏ do các địa chỉ lớp A được dành chủ yếu cho các công ty cung cấp dịch vụ lớn tại Mỹ và rất hạn chế trong việc cấp phát. Các địa chỉ lớp B nhanh chóng bị sử dụng hết do nó cung cấp số địa chỉ vừa phải. Hiện nay nhiều yêu cầu chỉ được đáp ứng bằng các địa chỉ lớp C với số địa chỉ rất hạn chế.
- Sự gia tăng số lượng các chỉ mục trong bảng định tuyến do cơ chế định tuyến không phân cấp dẫn đến yêu cầu nâng cấp các router và và định tuyến không hiệu quả.
- Ngày nay, với các nhu cầu kết nối vào mạng Internet của các dịch vụ khác như điện thoại di động, truyền hình số,… đòi hỏi giao thức IPv4 cần có các sửa đổi để đáp ứng các nhu cầu mới.

Trước những nhu cầu này, giao thức liên mạng thế hệ mới IPv6 đã ra đời nhằm thay thế cho IPv4, nhưng cho đến nay IPv6 vẫn chỉ mới chủ yếu là đang trong quá trình thử nghiệm và hoàn thiện. Trong khuôn khổ giáo trình cũng đề cập một cách tổng quát về giao thức liên mạng thế hệ mới IPv6.

VII.2. Một số đặc điểm mới của IPv6:

- Khuôn dạng header mới: Header của IPv6 được thiết kế để giảm chi phí đến mức tối thiểu. Điều này đạt được bằng cách chuyển các trường lựa chọn sang các header

mở rộng được đặt phía sau của IPv6 header. Khuôn dạng mới của IPv6 tạo ra sự xử lý hiệu quả hơn tại các router.

− Header của IPv4 và IPv6 không thể xử lý chung. Một trạm hay một router phải cài đặt cả IPv4 và IPv6 để có thể xử lý được cả hai khuôn dạng header này. Header của IPv6 chỉ có kích thước gấp 2 lần header của IPv4 mặc dù không gian địa chỉ của IPv6 lớn gấp 4 lần không gian địa chỉ IPv4.

− Không gian địa chỉ lớn: IPv6 có địa chỉ nguồn và đích dài 128 bit. Mặc dù 128 bit có thể tạo ra hơn 3.4x1038 tổ hợp, không gian địa chỉ của IPv6 được thiết kế cho phép phân bổ địa chỉ và mạng con từ trục xương sống Internet đến từng mạng con trong một tổ chức.

− Hiện tại chỉ một lượng nhỏ các địa chỉ hiện đang được phân bổ để sử dụng bởi các trạm, vẫn còn dư thừa rất nhiều địa chỉ sẵn sàng cho việc sử dụng trong tương lai.

− Hiệu quả, phân cấp địa chỉ hóa và hạ tầng định tuyến: Các địa chỉ toàn cục của IPv6 được thiết kế để tạo ra mọt hạ tầng định tuyến hiệu quả, phân cấp và có thể tổng quát hóa dựa trên sự phân cấp thường thấy của các nhà cung cấp dịch vụ (ISP) trên thực tế.

− Hỗ trợ chất lượng dịch vụ (QoS) tốt hơn: Các trường mới trong header của IPv6 định ra cách thức xử lý và định danh trên mạng. Giao thông trên mạng được định danh nhờ trường gán nhãn luồng (Flow Label) cho phép router có thể nhận ra và cung cấp các xử lý đặc biệt đối với các gói tin thuộc về một luồng nhất định, một chuẩn các gói tin giữa nguồn và đích.

Do giao thông mạng được xác định trong header, các dịch vụ QoS có thể được thực hiện ngay cả khi phần dữ liệu được mã hóa theo IPSec.

− Khả năng mở rộng: IPv6 có thể dễ dàng mở rộng thêm các tính năng mới bằng việc thêm các header mới sau header IPv6.

VII.3. Kiến trúc địa chỉ trong IPv6:

VII.3.1 Không gian địa chỉ:

− IPv6 sử dụng địa chỉ có độ dài lớn hơn IPv4 (128 bit so với 32 bit) do đó cung cấp không gian địa chỉ lớn hơn rất nhiều. Trong khi không gian địa chỉ 32 bit của IPv4 cho phép khoảng 4 tỷ địa chỉ, không gian địa chỉ của

IPv6 có thể có khoảng 3.4x1038 địa chỉ. Số lượng điạc hỉ này rất lớn, hỗ trợ khoảng 6.5x1023 địa chỉ trên mỗi mét vuông bề mặt trái đất. Địa chỉ IPv6 128 bit được chia thành các miền phân cấp theo trật tự trên Internet.

Nó tạo ra nhiều mức phân cấp và linh hoạt trong địa chỉ hóa và định tuyến còn đang thiếu trong IPv4.

− Không gian địa chỉ IPv6 được chia trên cơ sở các bit đầu trong địa chỉ. Trường có độ dài thay đổi bao gồm các bit đầu tiên trong địa chỉ gọi là tiền tố định dạng (Format Prefix) FP.

− Ban đầu chỉ mới có 15% lượng địa chỉ được sử dụng, 85% còn lại để dùng trong tương lai.

− Các tiền tố định dạng từ 001 đến 111, ngoại trừ kiểu địa chỉ multicast (1111 1111) đều bắt buộc có định danh giao diện theo khuôn dạng EUI-64.

− Các địa chỉ dự trữ không lẫn với các địa chỉ chưa cấp phát. Chúng chiếm 1/256 không gian địa chỉ (FP = 0000 0000) và dùng cho các địa chỉ chưa chỉ định, địa chỉ quay vòng và các địa chỉ IPv6 có nhúng IPv4

VII.3.2 Cú pháp địa chỉ:

Các địa chỉ IPv6 dài 128 bit, khi viết mỗi nhóm 16 bit được biểu diễn thành một số nguyên không dấu dưới dạng hệ 16 và được phân tách bởi dấu hai chấm (:),

Ví dụ: FEDC:BA98:7654:3210:FEDC:BA98:7654:3210

Trên thực tế địa chỉ IPv6 thường có nhiều số 0, ví dụ địa chỉ:

1080:0000:0000:0000:0008:0800:200C:417A. Do đó cơ chế nén địa chỉ được dùng để biểu diễn dễ dàng hơn các loại địa chỉ dạng này. Ta không cần viết các số 0 ở đầu mỗi nhóm, ví dụ 0 thay cho 0000, 20 thay cho 0020.

Địa chỉ trong ví dụ trên sẽ trở thành 1080:0:0:0:8:800:200C:417A.

Hơn nữa ta có thể sử dụng ký hiệu :: để chỉ một chuỗi số 0. Địa chỉ trong ví dụ trên sẽ trở thành: 1080::8:800:200C:417A. Do địa chỉ IPv6 có độ dài cố định, ta có thể tính được số các bit 0 mà ký hiệu đó biểu diễn.

Tiền tố địa chỉ IPv6 được biểu diễn theo ký pháp CIDR như IPv4 như sau:

IPv6-address/prefix length trong đó IPv6-address là bất kỳ kiểu biểu diễn nào, còn prefix length là độ dài tiền tố theo bit.

Ví dụ: biểu diễn mạng con có tiền tố 80 bit: 1080:0:0:0:8::/80.

Với node address: 12AB:0:0:CD30:123:4567:89AB:CDEF,

prefix: 12AB:0:0:CD30::/60 có thể viết tắt thành

12AB:0:0:CD30:123:4567:89AB:CDEF/60

Chương IV Thiết bị mạng

I. Môi trường truyền dẫn

I.1. Khái niệm

Trên một mạng máy tính, các dữ liệu được truyền trên một môi trường truyền dẫn (transmission media), nó là phương tiện vật lý cho phép truyền tải tín hiệu giữa các thiết bị. Có hai loại phương tiện truyền dẫn chủ yếu:

- Hữu tuyến (bounded media)

- Vô tuyến (boundless media)

Thông thường hệ thống mạng sử dụng hai loại tín hiệu là: digital và analog.

I.2. Tần số truyền thông

Phương tiện truyền dẫn giúp truyền các tín hiệu điện tử từ máy tính này sang máy tính khác. Các tín hiệu điện tử này biểu diễn các giá trị dữ liệu theo dạng các xung nhị phân (bật/tắt). Các tín hiệu truyền thông giữa các máy tính và các thiết bị là các dạng sóng điện từ trải dài từ tần số radio đến tần số hồng ngoại.

Các sóng tần số radio thường được dùng để phát tín hiệu LAN. Các tần số này có thể được dùng với cáp xoắn đôi, cáp đồng trục hoặc thông qua việc truyền phủ sóng radio.

Sóng viba (microware) thường dùng truyền thông tập trung giữa hai điểm hoặc giữa các trạm mặt đất và các vệ tinh, ví dụ như mạng điện thoại cellular.

Tia hồng ngoại thường dùng cho các kiểu truyền thông qua mạng trên các khoảng cách tương đối ngắn và có thể phát được sóng giữa hai điểm hoặc từ một điểm phủ sóng cho nhiều trạm thu. Chúng ta có thể truyền tia hồng ngoại và các tần số ánh sáng cao hơn thông qua cáp quang.

I.3. Các đặc tính của phương tiện truyền dẫn

Mỗi phương tiện truyền dẫn đều có những tính năng đặc biệt thích hợp với mỗi kiểu dịch vụ cụ thể, nhưng thông thường chúng ta quan tâm đến những yếu tố sau:

- Chi phí

- Yêu cầu cài đặt

- Độ bảo mật

- Băng thông (bandwidth): được xác định bằng tổng lượng thông tin có thể truyền dẫn trên đường truyền tại một thời điểm. Băng thông là một số xác định, bị giới hạn bởi phương tiện truyền dẫn, kỹ thuật truyền dẫn và thiết bị mạng được sử dụng. Băng thông là một trong những thông số dùng để phân tích độ hiệu quả của đường mạng. Đơn vị của băng thông:

+ Bps (Bits per second-số bit trong một giây): đây là đơn vị cơ bản của băng thông.

+ KBps (Kilobits per second): 1 KBps=103 bps=1000 Bps

+ MBps (Megabits per second): 1 MBps = 103 KBps

+ GBps (Gigabits per second): 1 GBps = 103 MBps

+ TBps (Terabits per second): 1 TBps = 103 GBPS.

- Thông lượng (Throughput): lượng thông tin thực sự được truyền dẫn trên thiết bị tại một thời điểm.

- Băng tầng cơ sở (baseband): dành toàn bộ băng thông cho một kênh truyền, băng tầng mở rộng (broadband):cho phép nhiều kênh truyền chia sẻ một phương tiện truyền dẫn (chia sẻ băng thông).

- Độ suy giảm (attenuation): độ đo sự suy yếu đi của tín hiệu khi di chuyển trên một phương tiện truyền dẫn. Các nhà thiết kế cáp phải chỉ định các giới hạn về chiều dài dây cáp vì khi cáp dài sẽ dẫn đến tình trạng tín hiệu yếu đi mà không thể phục hồi được.

- Nhiễu điện từ (Electromagnetic interference - EMI): bao gồm các nhiễu điện từ bên ngoài làm biến dạng tín hiệu trong một phương tiện truyền dẫn.

- Nhiễu xuyên kênh (crosstalk): hai dây dẫn đặt kề nhau làm nhiễu lẫn nhau.

I.4. Các kiểu truyền dẫn.

Có các kiểu truyền dẫn như sau:

- **Đơn công (Simplex)**: Trong kiểu truyền dẫn này, thiết bị phát tín hiệu và thiết bị nhận tín hiệu được phân biệt rõ ràng, thiết bị phát chỉ đảm nhiệm vai trò phát tín hiệu, còn thiết bị thu chỉ đảm nhiệm vai trò nhận tín hiệu. Truyền hình là một ví dụ của kiểu truyền dẫn này.
- **Bán song công (Half-Duplex)**: trong kiểu truyền dẫn này, thiết bị có thể là thiết bị phát, vừa là thiết bị thu. Nhưng tại một thời điểm thì chỉ có thể ở một trạng thái (phát hoặc thu). Bộ đàm là thiết bị hoạt động ở kiểu truyền dẫn này.
- **Song công (Full-Duplex)**: trong kiểu truyền dẫn này, tại một thời điểm, thiết bị có thể vừa phát vừa thu. Điện thoại là một minh họa cho kiểu truyền dẫn này.

II. Đường cáp truyền mạng

Đường cáp truyền mạng là cơ sở hạ tầng của một hệ thống mạng, nên nó rất quan trọng và ảnh hưởng rất nhiều đến khả năng hoạt động của mạng. Hiện nay người ta thường dùng 3 loại dây cáp là cáp xoắn cặp, cáp đồng trục và cáp quang.

II.1. Cáp xoắn cặp

Đây là loại cáp gồm hai đường dây dẫn đồng được xoắn vào nhau nhằm làm giảm nhiễu điện từ gây ra bởi môi trường xung quanh và giữa chúng với nhau.

Hiện nay có hai loại cáp xoắn là cáp có bọc kim loại (STP - Shield Twisted Pair) và cáp không bọc kim loại (UTP -Unshield Twisted Pair).

- Cáp có bọc kim loại (STP): Lớp bọc bên ngoài có tác dụng chống nhiễu điện từ, có loại có một đôi giây xoắn vào nhau và có loại có nhiều đôi giây xoắn với nhau.
- Cáp không bọc kim loại (UTP): Tính tương tự như STP nhưng kém hơn về khả năng chống nhiễu và suy hao vì không có vỏ bọc.

STP và UTP có các loại (Category - Cat) thường dùng:

- Loại 1 & 2 (Cat 1 & Cat 2): Thường dùng cho truyền thoại và những đường truyền tốc độ thấp (nhỏ hơn 4Mb/s).
- Loại 3 (Cat 3): tốc độ truyền dữ liệu khoảng 16 Mb/s , nó là chuẩn cho hầu hết các mạng điện thoại.
- Loại 4 (Cat 4): Thích hợp cho đường truyền 20Mb/s.
- Loại 5 (Cat 5): Thích hợp cho đường truyền 100Mb/s.
- Loại 6 (Cat 6): Thích hợp cho đường truyền 300Mb/s.

Đây là loại cáp rẻ, dễ cài đặt tuy nhiên nó dễ bị ảnh hưởng của môi trường.

II.2. Cáp đồng trục

Cáp đồng trục có hai đường dây dẫn và chúng có cùng một trục chung, một dây dẫn trung tâm (thường là dây đồng cứng) đường dây còn lại tạo thành đường ống bao xung quanh dây dẫn trung tâm (dây dẫn này có thể là dây bện kim loại và vì nó có chức năng chống nhiễu nên còn gọi là lớp bọc kim). Giữa hai dây dẫn trên có một lớp cách ly, và bên ngoài cùng là lớp vỏ plastic để bảo vệ cáp.

Các loại cáp	***Dây xoắn cặp***	***Cáp đồng trục mỏng***	***Cáp đồng trục dày***	***Cáp quang***
Chi tiết	Bằng đồng, có 4 và 25 cặp dây (loại 3, 4, 5)	Bằng đồng, 2 dây, đường kính 5mm	Bằng đồng, 2 dây, đường kính 10mm	Thủy tinh, 2 sợi
Loại kết nối	RJ-25 hoặc 50-pin telco	BNC	N-series	ST
Chiều dài đoạn tối đa	100m	185m	500m	1000m
Số đầu nối tối đa trên 1 đoạn	2	30	100	2
Chạy 10 Mbit/s	Được	Được	Được	Được
Chạy 100 Mbit/s	Được	Không	Không	Được
Chống nhiễu	Tốt	Tốt	Rất tốt	Hoàn toàn
Bảo mật	Trung bình	Trung bình	Trung bình	Hoàn toàn
Độ tin cậy	Tốt	Trung bình	Tốt	Tốt
Lắp đặt	Dễ dàng	Trung bình	Khó	Khó
Khắc phục lỗi	Tốt	Dở	Dở	Tốt
Quản lý	Dễ dàng	Khó	Khó	Trung bình
Chi phí cho 1 trạm	Rất thấp	Thấp	Trung bình	Cao
Ứng dụng tốt nhất	Hệ thống Workgroup	Đường backbone	Đường backbone trong tủ mạng	Đường backbone dài trong tủ mạng hoặc các tòa nhà

Bảng IV-1 ***Tính năng kỹ thuật của một số loại cáp mạng***

Cáp đồng trục có độ suy hao ít hơn so với các loại cáp đồng khác (ví dụ như cáp xoắn đôi) do ít bị ảnh hưởng của môi trường. Các mạng cục bộ sử dụng cáp đồng trục có thể có kích thước trong phạm vi vài ngàn mét, cáp đồng trục được sử dụng nhiều trong các mạng dạng đường thẳng. Hai loại cáp thường được sử dụng là cáp đồng trục mỏng và cáp đồng trục dày trong đường kính cáp đồng trục mỏng là 0,25 inch, cáp đồng trục dày là

0,5 inch. Cả hai loại cáp đều làm việc ở cùng tốc độ nhưng cáp đồng trục mỏng có độ hao suy tín hiệu lớn hơn

Hiện nay có cáp đồng trục sau:

- RG -58,50 ohm: dùng cho mạng Thin Ethernet
- RG -59,75 ohm: dùng cho truyền hình cáp
- RG -62,93 ohm: dùng cho mạng ARCnet

Các mạng cục bộ thường sử dụng cáp đồng trục có dải thông từ 2,5 - 10 Mb/s, cáp đồng trục có độ suy hao ít hơn so với các loại cáp đồng khác vì nó có lớp vỏ bọc bên ngoài, độ dài thông thường của một đoạn cáp nối trong mạng là 200m, thường sử dụng cho dạng Bus.

II.3. Cáp sợi quang (Fiber - Optic Cable)

Cáp sợi quang bao gồm một dây dẫn trung tâm (là một hoặc một bó sợi thủy tinh có thể truyền dẫn tín hiệu quang) được bọc một lớp vỏ bọc có tác dụng phản xạ các tín hiệu trở lại để giảm sự mất mát tín hiệu. Bên ngoài cùng là lớp vỏ plastic để bảo vệ cáp. Như vậy cáp sợi quang không truyền dẫn các tín hiệu điện mà chỉ truyền các tín hiệu quang (các tín hiệu dữ liệu phải được chuyển đổi thành các tín hiệu quang và khi nhận chúng sẽ lại được chuyển đổi trở lại thành tín hiệu điện).

Cáp quang có đường kính từ 8.3 - 100 micron, Do đường kính lõi sợi thuỷ tinh có kích thước rất nhỏ nên rất khó khăn cho việc đấu nối, nó cần công nghệ đặc biệt với kỹ thuật cao đòi hỏi chi phí cao.

Dải thông của cáp quang có thể lên tới hàng Gbps và cho phép khoảng cách đi cáp khá xa do độ suy hao tín hiệu trên cáp rất thấp. Ngoài ra, vì cáp sợi quang không dùng tín hiệu điện từ để truyền dữ liệu nên nó hoàn toàn không bị ảnh hưởng của nhiễu điện từ và tín hiệu truyền không thể bị phát hiện và thu trộm bởi các thiết bị điện tử của người khác.

Chỉ trừ nhược điểm khó lắp đặt và giá thành còn cao , nhìn chung cáp quang thích hợp cho mọi mạng hiện nay và sau này.

II.4. Các yêu cầu cho một hệ thống cáp

- **An toàn, thẩm mỹ**: Tất cả các dây mạng phải được bao bọc cẩn thận, cách xa các nguồn điện, các máy có khả năng phát sóng để tránh trường hợp bị nhiễu. Các đầu nối phải đảm bảo chất lượng, tránh tình trạng hệ thống mạng bị chập chờn.
- **Đúng chuẩn**: Hệ thống cáp phải thực hiện đúng chuẩn, đảm bảo cho khả năng nâng cấp sau này cũng như dễ dàng cho việc kết nối các thiết bị khác nhau của các nhà sản xuất khác nhau. Tiêu chuẩn quốc tế dùng cho các hệ thống mạng hiện nay là EIA/TIA 568B.
- **Tiết kiệm và "linh hoạt" (flexible)**: Hệ thống cáp phải được thiết kế sao cho kinh tế nhất, dễ dàng trong việc di chuyển các trạm làm việc và có khả năng mở rộng sau này.

III. Đường truyền vô tuyến

Khi dùng các loại cáp ta gặp một số khó khăn như cơ sở cài đặt cố định, khoảng cách không xa, vì vậy để khắc phục những khuyết điểm trên người ta dùng đường truyền vô tuyến. Đường truyền vô tuyến mang lại những lợi ích sau:

- Cung cấp nối kết tạm thời với mạng cáp có sẵn.

- Những người liên tục di chuyển vẫn nối kết vào mạng dùng cáp.

- Lắp đặt đường truyền vô tuyến ở những nơi địa hình phức tạp không thể đi dây được.

- Phù hợp cho những nơi phục vụ nhiều kết nối cùng một lúc cho nhiều khách hàng. Ví dụ như: Dùng đường vô tuyến cho phép khách hàng ở sân bay kết vào mạng để duyệt Internet.

- Dùng cho những mạng có giới hạn rộng lớn vượt quá khả năng cho phép của cáp đồng và cáp quang.

- Dùng làm kết nối dự phòng cho các kết nối hệ thống cáp.

Tuy nhiên, đường truyền vô tuyến cũng có một số hạn chế:

- Tín hiệu không an toàn.

- Dễ bị nghe lén.

- Khi có vật cản thì tín hiệu suy yếu rất nhanh.

- Băng thông không cao.

III.1. Sóng vô tuyến (radio)

Sóng radio nằm trong phạm vi từ 10 KHz đến 1 GHz, trong miền này ta có rất nhiều dải tần ví dụ như: sóng ngắn, VHF (dùng cho tivi và radio FM), UHF (dùng cho tivi). Tại mỗi quốc gia, nhà nước sẽ quản lý cấp phép sử dụng các băng tần để tránh tình trạng các sóng bị nhiễu. Nhưng có một số băng tần được chỉ định là vùng tự do có nghĩa là chúng ta dùng nhưng không cần đăng ký (vùng này thường có dải tần 2,4 Ghz). Tận dụng lợi điểm này các thiết bị Wireless của các hãng như Cisco, Compex đều dùng ở dải tần này. Tuy nhiên, chúng ta sử dụng tần số không cấp phép sẽ có nguy cơ nhiễu nhiều hơn.

III.2. Sóng viba

Truyền thông viba thường có hai dạng: truyền thông trên mặt đất và các nối kết với vệ tinh. Miền tần số của viba mặt đất khoảng 21-23 GHz, các kết nối vệ tinh khoảng 11-14 Mhz. Băng thông từ 1-10 MBps.

Sự suy yếu tín hiệu tùy thuộc vào điều kiện thời tiết, công suất và tần số phát. Chúng dễ bị nghe trộm nên thường được mã hóa.

III.3. Hồng ngoại

Tất cả mạng vô tuyến hồng ngoại đều hoạt động bằng cách dùng tia hồng ngoại để truyền tải dữ liệu giữa các thiết bị. Phương pháp này có thể truyền tín hiệu ở tốc độ cao do dải thông cao của tia hồng ngoại. Thông thường mạng hồng ngoại có thể truyền với tốc độ từ 1-10 Mbps. Miền tần số từ 100 Ghz đến 1000 GHz. Có bốn loại mạng hồng ngoại:

- **Mạng đường ngắm**: mạng này chỉ truyền khi máy phát và máy thu có một đường ngắm rõ rệt giữa chúng.

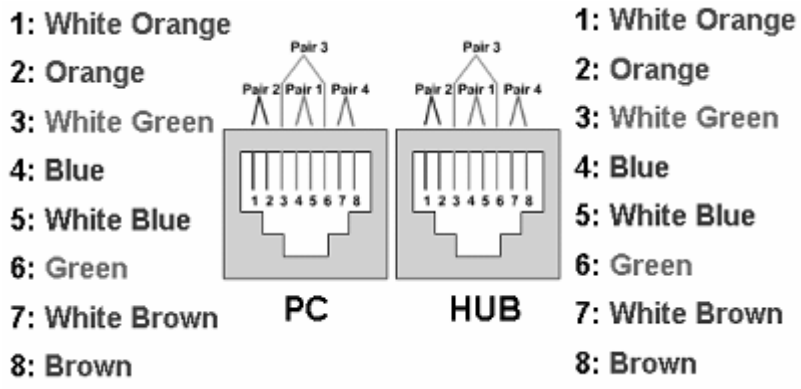
- **Mạng hồng ngoại tán xạ**: kỹ thuật này phát tia truyền dội tường và sàn nhà rồi mới đến máy thu. Diện tích hiệu dụng bị giới hạn ở khoảng 100 feet (35m) và có tín hiệu chậm do hiện tượng dội tín hiệu.

- **Mạng phản xạ**: ở loại mạng hồng ngoại này, máy thu-phát quang đặt gần máy tính sẽ truyền tới một vị trí chung, tại đây tia truyền được đổi hướng đến máy tính thích hợp.

- **Broadband optical telepoint**: loại mạng cục bộ vô tuyến hồng ngoại cung cấp các dịch vụ dải rộng. Mạng vô tuyến này có khả năng xử lý các yêu cầu đa phương tiện chất lượng cao, vốn có thể trùng khớp với các yêu cầu đa phương tiện của mạng cáp.

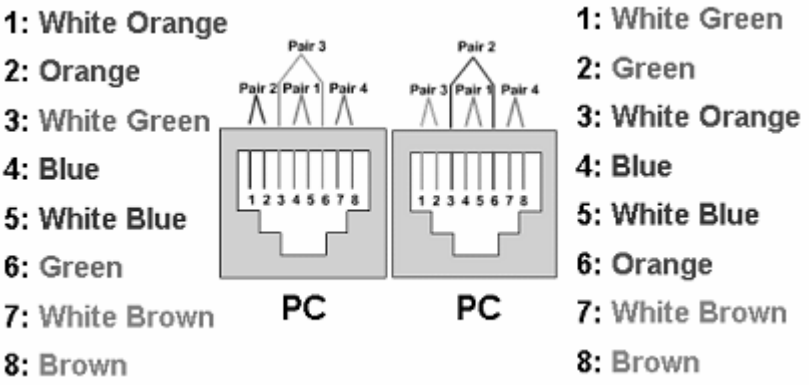
IV. Các kỹ thuật bấm cáp mạng

- **Cáp thẳng (Straight-through cable)**: là cáp dùng để nối PC và các thiết bị mạng như Hub, Switch, Router… Cáp thẳng theo chuẩn 10/100 Base-T dùng hai cặp dây xoắn nhau và dùng chân 1, 2, 3, 6 trên đầu RJ45. Cặp dây xoắn thứ nhất nối vào chân 1, 2, cặp xoắn thứ hai nối vào chân 3, 6. Đầu kia của cáp dựa vào màu nối vào chân của đầu RJ45 và nối tương tự.



Hình IV-1: Cách đấu dây thẳng.

- **Cáp chéo (Crossover cable)**: là cáp dùng nối trực tiếp giữa hai thiết bị giống nhau như PC – PC, Hub – Hub, Switch – Switch. Cáp chéo trật tự dây cũng giống như cáp thẳng nhưng đầu dây còn lại phải chéo cặp dây xoắn sử dụng (vị trí thứ nhất đổi với vị trí thứ 3, vị trí thứ hai đổi với vị trí thứ sáu) .

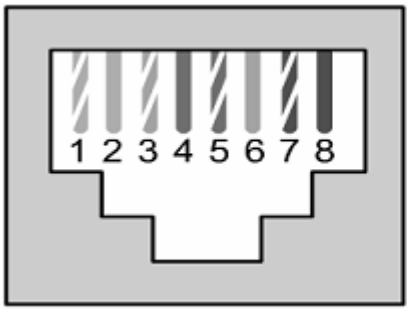


Hình IV-2: Cách đấu dây chéo.

- **Cáp Console**: Dùng để nối PC vào các thiết bị mạng chủ yếu dùng để cấu hình các thiết bị. Thông thường khoảng cách dây Console ngắn nên chúng ta không cần chọn cặp dây xoắn, mà chọn theo màu từ 1-8 sao cho dễ nhớ và đầu bên kia ngược lại từ 8-1.

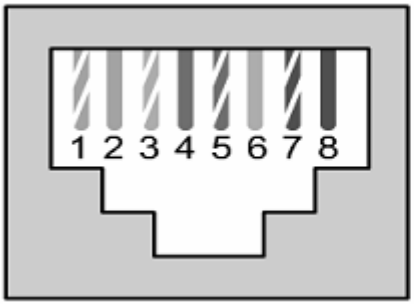
ANSI (Viện tiêu chuẩn quốc gia Hoa kỳ), TIA (hiệp hội công nghiệp viễn thông), EIA (hiệp hội công nghiệp điện tử) đã đưa ra 2 cách xếp đặt vị trí dây như sau:

- Chuẩn T568-A (còn gọi là Chuẩn A):



1. Trắng Xanh lá cây (White Green)
2. Xanh lá cây (Green)
3. Trắng Cam (White Orange)
4. Xanh đậm (Blue)
5. Trắng Xanh đậm (White Blue)
6. Cam (Orange)
7. Trắng Nâu (White Brown)
8. Nâu (Brown)

- Chuẩn T568-B (còn gọi là Chuẩn B):



1. Trắng Cam (White Orange)
2. Cam (Orange)
3. Trắng Xanh lá cây (White Green)
4. Xanh đậm (Blue)
5. Trắng Xanh đậm (White Blue)
6. Xanh lá cây (Green)
7. Trắng Nâu (White Brown)
8. Nâu (Brown)

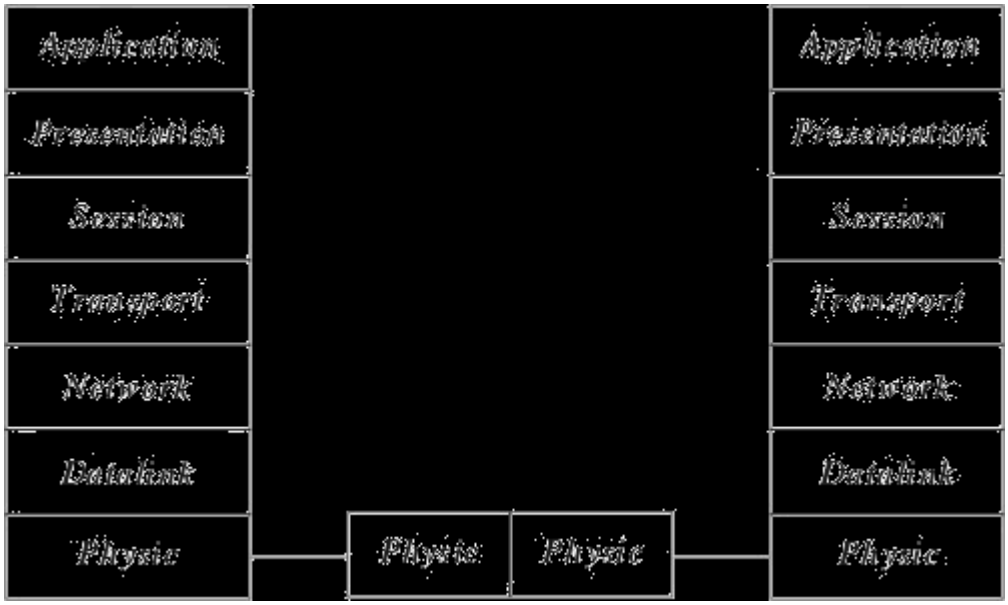
V. Các thiết bị liên kết mạng

V.1. Repeater (Bộ tiếp sức)

Repeater là loại thiết bị phần cứng đơn giản nhất trong các thiết bị liên kết mạng, nó được hoạt động trong tầng vật lý của mô hình hệ thống mở OSI. Repeater dùng để nối 2 mạng giống nhau hoặc các phần một mạng cùng có một nghi thức và một cấu hình. Khi Repeater nhận được một tín hiệu từ một phía của mạng thì nó sẽ phát tiếp vào phía kia của mạng.

Hình IV-3: Mô hình liên kết mạng của Repeater.

Repeater không có xử lý tín hiệu mà nó chỉ loại bỏ các tín hiệu méo, nhiễu, khuếch đại tín hiệu đã bị suy hao (vì đã được phát với khoảng cách xa) và khôi phục lại tín hiệu ban đầu. Việc sử dụng Repeater đã làm tăng thêm chiều dài của mạng.



Hình IV-4: Hoạt động của bộ tiếp sức trong mô hình OSI

Hiện nay có hai loại Repeater đang được sử dụng là Repeater điện và Repeater điện quang.

- Repeater điện nối với đường dây điện ở cả hai phía của nó, nó nhận tín hiệu điện từ một phía và phát lại về phía kia. Khi một mạng sử dụng Repeater điện để nối các phần của mạng lại thì có thể làm tăng khoảng cách của mạng, nhưng khoảng cách đó luôn bị hạn chế bởi một khoảng cách tối đa do độ trễ của tín hiệu. Ví dụ với mạng sử dụng cáp đồng trục 50 thì khoảng cách tối đa là 2.8 km, khoảng cách đó không thể kéo thêm cho dù sử dụng thêm Repeater.
- Repeater điện quang liên kết với một đầu cáp quang và một đầu là cáp điện, nó chuyển một tín hiệu điện từ cáp điện ra tín hiệu quang để phát trên cáp quang và ngược lại. Việc sử dụng Repeater điện quang cũng làm tăng thêm chiều dài của mạng.

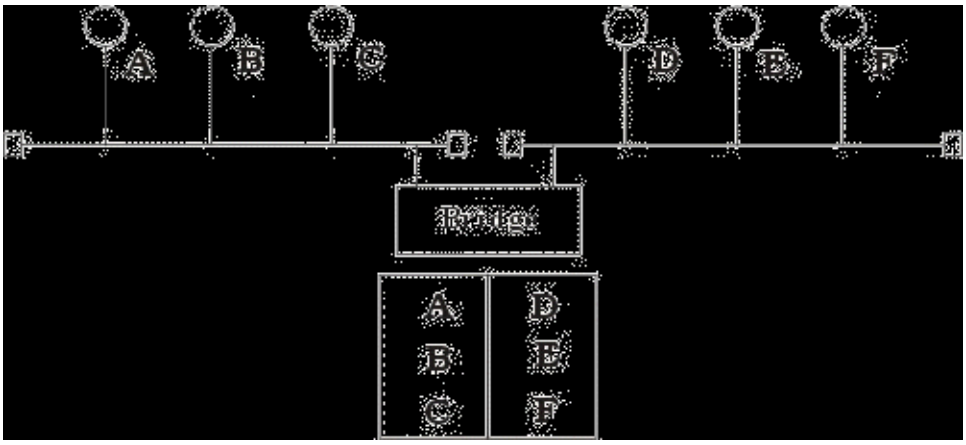
Việc sử dụng Repeater không thay đổi nội dung các tín hiện đi qua nên nó chỉ được dùng để nối hai mạng có cùng giao thức truyền thông (như hai mạng Ethernet hay hai mạng Token ring) nhưng không thể nối hai mạng có giao thức truyền thông khác nhau (như một mạng Ethernet và một mạng Token ring). Thêm nữa Repeater không làm thay đổi khối lượng chuyển vận trên mạng nên việc sử dụng không tính toán nó trên mạng lớn sẽ hạn chế hiệu năng của mạng. Khi lưa chọn sử dụng Repeater cần chú ý lựa chọn loại có tốc độ chuyển vận phù hợp với tốc độ của mạng.

V.2. Bridge (Cầu nối)

Bridge là một thiết bị có xử lý dùng để nối hai mạng giống nhau hoặc khác nhau, nó có thể được dùng với các mạng có các giao thức khác nhau. Cầu nối hoạt động trên tầng liên kết dữ liệu nên không như bộ tiếp sức phải phát lại tất cả những gì nó nhận được thì cầu nối đọc được các gói tin của tầng liên kết dữ liệu trong mô hình OSI và xử lý chúng trước khi quyết định có chuyển đi hay không.

Khi nhận được các gói tin Bridge chọn lọc và chỉ chuyển những gói tin mà nó thấy cần thiết. Điều này làm cho Bridge trở nên có ích khi nối một vài mạng với nhau và cho phép nó hoạt động một cách mềm dẻo.

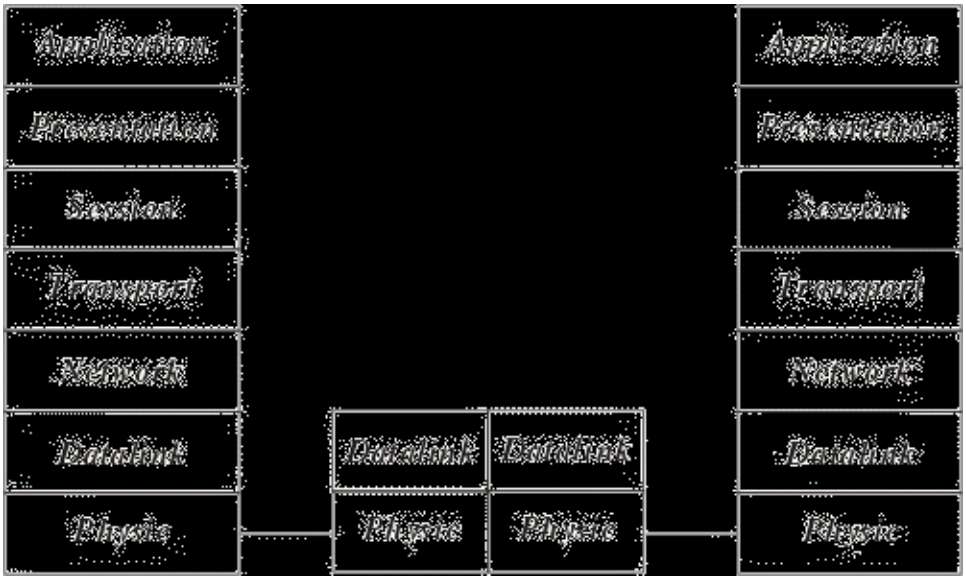
Để thực hiện được điều này trong Bridge ở mỗi đầu kết nối có một bảng các địa chỉ các trạm được kết nối vào phía đó, khi hoạt động cầu nối xem xét mỗi gói tin nó nhận được bằng cách đọc địa chỉ của nơi gửi và nhận và dựa trên bảng địa chỉ phía nhận được gói tin nó quyết định gửi gói tin hay không và bổ xung bảng địa chỉ.



Hình IV-5: Hoạt động của Bridge

Khi đọc địa chỉ nơi gửi Bridge kiểm tra xem trong bảng địa chỉ của phần mạng nhận được gói tin có địa chỉ đó hay không, nếu không có thì Bridge tự động bổ xung bảng địa chỉ (cơ chế đó được gọi là tự học của cầu nối).

Khi đọc địa chỉ nơi nhận Bridge kiểm tra xem trong bảng địa chỉ của phần mạng nhận được gói tin có địa chỉ đó hay không, nếu có thì Bridge sẽ cho rằng đó là gói tin nội bộ thuộc phần mạng mà gói tin đến nên không chuyển gói tin đó đi, nếu ngược lại thì Bridge mới chuyển sang phía bên kia. Ở đây chúng ta thấy một trạm không cần thiết chuyển thông tin trên toàn mạng mà chỉ trên phần mạng có trạm nhận mà thôi.



Hình IV-6: Hoạt động của Bridge trong mô hình OSI

Để đánh giá một Bridge người ta đưa ra hai khái niệm : Lọc và chuyển vận. Quá trình xử lý mỗi gói tin được gọi là quá trình lọc trong đó tốc độ lọc thể hiện trực tiếp khả năng hoạt động của

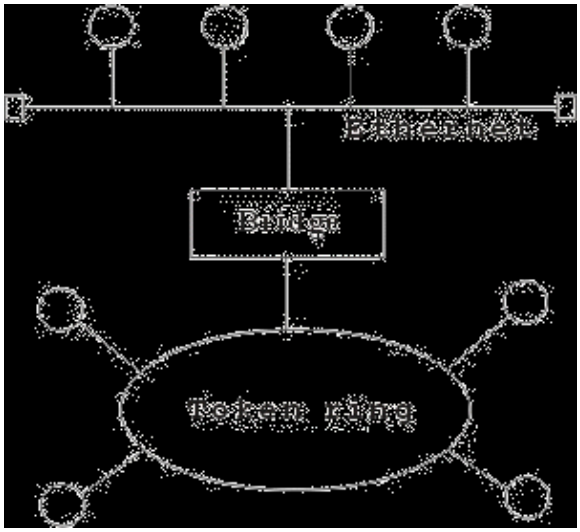
Bridge. Tốc độ chuyển vận được thể hiện số gói tin/giây trong đó thể hiện khả năng của Bridge chuyển các gói tin từ mạng này sang mạng khác.

Hiện nay có hai loại Bridge đang được sử dụng là Bridge vận chuyển và Bridge biên dịch. Bridge *vận chuyển* dùng để nối hai mạng cục bộ cùng sử dụng một giao thức truyền thông của tầng liên kết dữ liệu, tuy nhiên mỗi mạng có thể sử dụng loại dây nối khác nhau. Bridge vận chuyển không có khả năng thay đổi cấu trúc các gói tin mà nó nhận được mà chỉ quan tâm tới việc xem xét và chuyển vận gói tin đó đi.

Bridge biên dịch dùng để nối hai mạng cục bộ có giao thức khác nhau nó có khả năng chuyển một gói tin thuộc mạng này sang gói tin thuộc mạng kia trước khi chuyển qua

Ví dụ : Bridge biên dịch nối một mạng Ethernet và một mạng Token ring. Khi đó Cầu nối thực hiện như một nút token ring trên mạng Token ring và một nút Enthernet trên mạng Ethernet. Cầu nối có thể chuyển một gói tin theo chuẩn đang sử dụng trên mạng Enthernet sang chuẩn đang sử dụng trên mạng Token ring.

Tuy nhiên chú ý ở đây cầu nối không thể chia một gói tin ra làm nhiều gói tin cho nên phải hạn chế kích thước tối đa các gói tin phù hợp với cả hai mạng. Ví dụ như kích thước tối đa của gói tin trên mạng Ethernet là 1500 bytes và trên mạng Token ring là 6000 bytes do vậy nếu một trạm trên mạng token ring gửi một gói tin cho trạm trên mạng Ethernet với kích thước lớn hơn 1500 bytes thì khi qua cầu nối số lượng byte dư sẽ bị chặt bỏ.

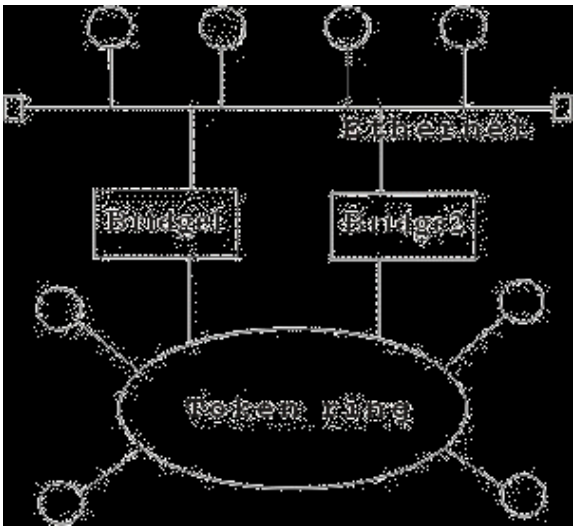


Hình IV-7: Ví dụ về Bridge biên dịch

Người ta sử dụng Bridge trong các trường hợp sau :

- Mở rộng mạng hiện tại khi đã đạt tới khoảng cách tối đa do Bridge sau khi sử lý gói tin đã phát lại gói tin trên phần mạng còn lại nên tín hiệu tốt hơn bộ tiếp sức.
- Giảm bớt tắc nghẽn mạng khi có quá nhiều trạm bằng cách sử dụng Bridge, khi đó chúng ta chia mạng ra thành nhiều phần bằng các Bridge, các gói tin trong nội bộ tùng phần mạng sẽ không được phép qua phần mạng khác.
- Để nối các mạng có giao thức khác nhau.

Một vài Bridge còn có khả năng lựa chọn đối tượng vận chuyển. Nó có thể chỉ chuyển vận những gói tin của nhưng địa chỉ xác định. Ví dụ : cho phép gói tin của máy A, B qua Bridge 1, gói tin của máy C, D qua Bridge 2.

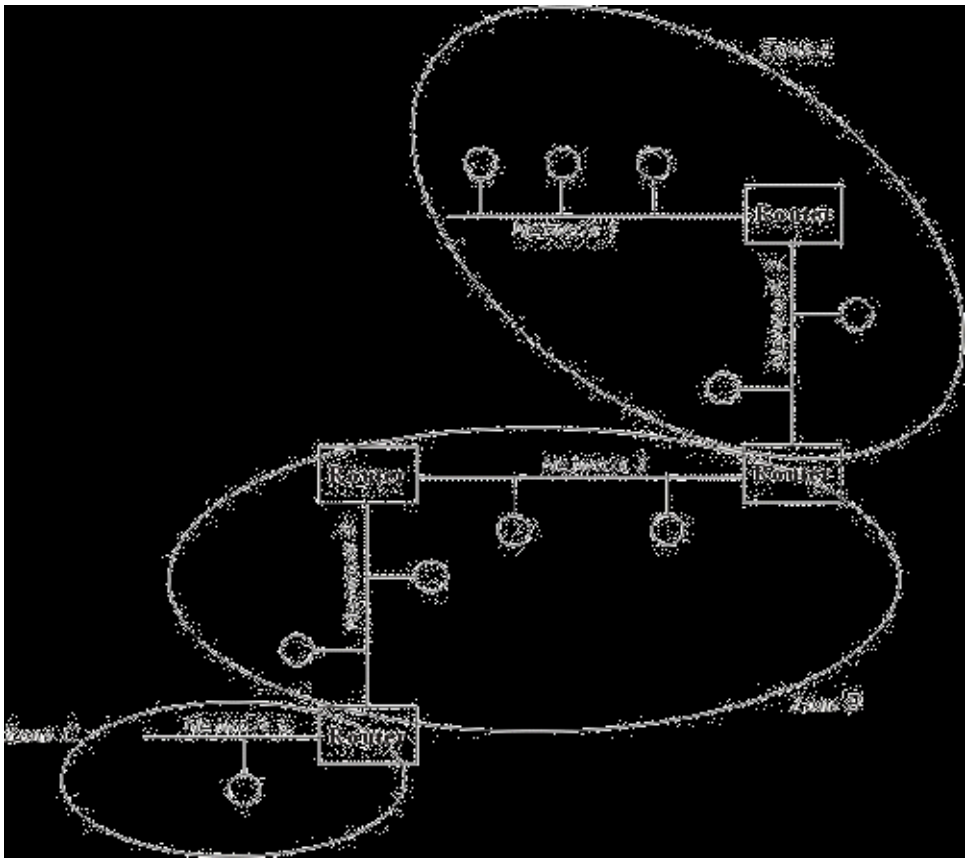


Hình IV-8: Liên kết mạng với 2 Bridge

Một số Bridge được chế tạo thành một bộ riêng biệt, chỉ cần nối dây và bật. Các Bridge khác chế tạo như card chuyên dùng cắïm vào máy tính, khi đó trên máy tính sẽ sử dụng phần mềm Bridge. Việc kết hợp phần mềm với phần cứng cho phép uyển chuyển hơn trong hoạt động của Bridge.

V.3. Router (Bộ tìm đường)

Router là một thiết bị hoạt động trên tầng mạng, nó có thể tìm được đường đi tốt nhất cho các gói tin qua nhiều kết nối để đi từ trạm gửi thuộc mạng đầu đến trạm nhận thuộc mạng cuối. Router có thể được sử dụng trong việc nối nhiều mạng với nhau và cho phép các gói tin có thể đi theo nhiều đường khác nhau để tới đích.



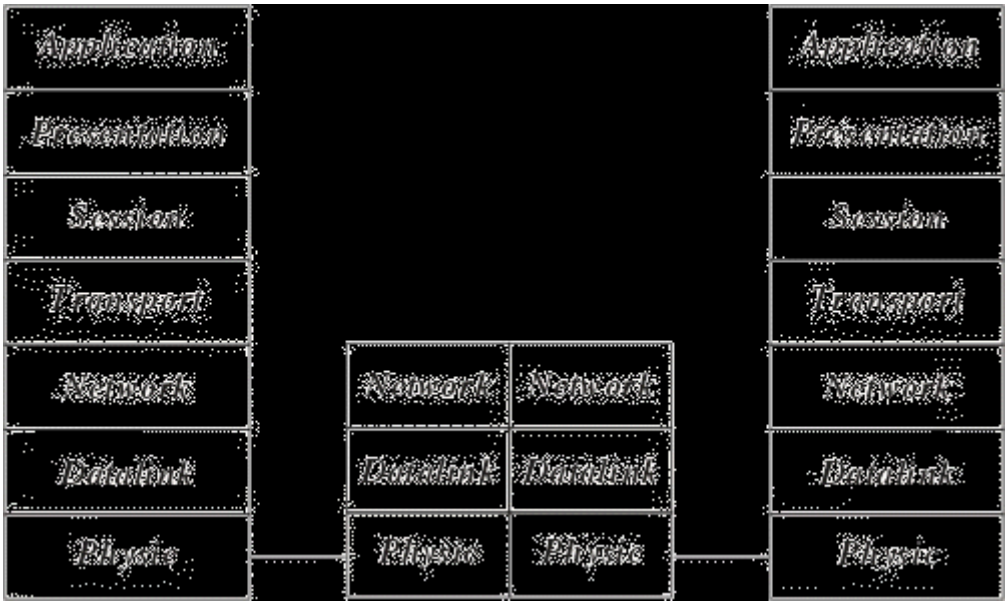
Hình IV-9: Hoạt động của Router.

Khác với Bridge hoạt động trên tầng liên kết dữ liệu nên Bridge phải xử lý mọi gói tin trên đường truyền thì Router có địa chỉ riêng biệt và nó chỉ tiếp nhận và xử lý các gói tin gửi đến nó mà thôi. Khi một trạm muốn gửi gói tin qua Router thì nó phải gửi gói tin với địa chỉ trực tiếp của Router (Trong gói tin đó phải chứa các thông tin khác về đích đến) và khi gói tin đến Router thì Router mới xử lý và gửi tiếp.

Khi xử lý một gói tin Router phải tìm được đường đi của gói tin qua mạng. Để làm được điều đó Router phải tìm được đường đi tốt nhất trong mạng dựa trên các thông tin nó có về mạng, thông thường trên mỗi Router có một bảng chỉ đường (Router table). Dựa trên dữ liệu về Router gần đó và các mạng trong liên mạng, Router tính được bảng chỉ đường (Router table) tối ưu dựa trên một thuật toán xác định trước.

Người ta phân chia Router thành hai loại là Router có phụ thuộc giao thức (The protocol dependent routers) và Router không phụ thuộc vào giao thức (The protocol independent router) dựa vào phương thức xử lý các gói tin khi qua Router.

- Router có phụ thuộc giao thức: Chỉ thực hiện việc tìm đường và truyền gói tin từ mạng này sang mạng khác chứ không chuyển đổi phương cách đóng gói của gói tin cho nên cả hai mạng phải dùng chung một giao thức truyền thông.
- Router không phụ thuộc vào giao thức: có thể liên kết các mạng dùng giao thức truyền thông khác nhau và có thể chuyển đôi gói tin của giao thức này sang gói tin của giao thức kia, Router cũng chấp nhận kích thước các gói tin khác nhau (Router có thể chia nhỏ một gói tin lớn thành nhiều gói tin nhỏ trước truyền trên mạng).

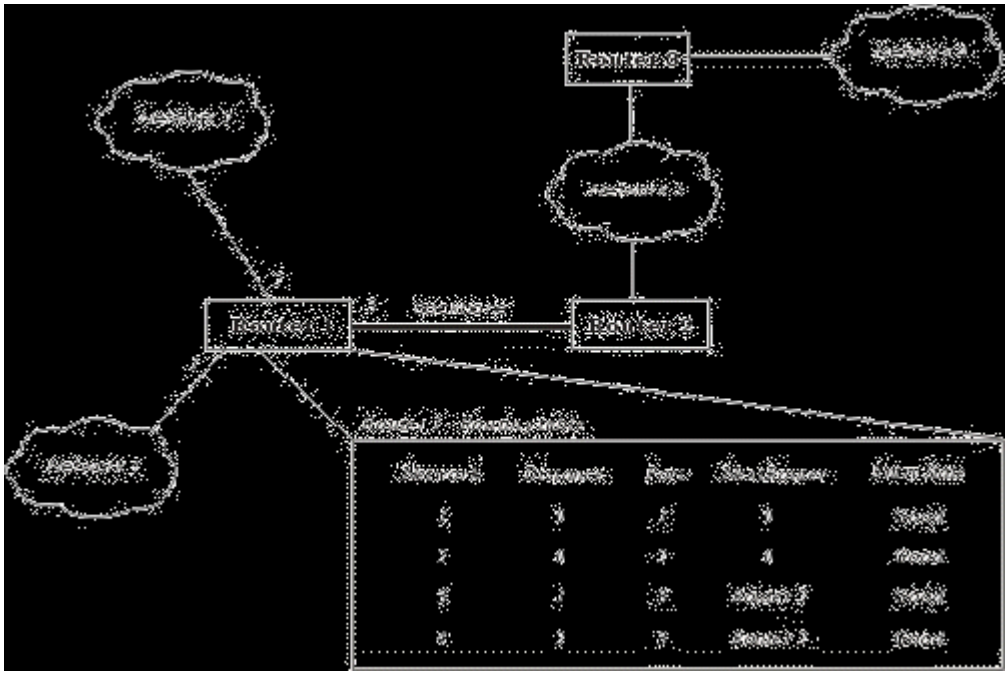


Hình IV-10: Hoạt động của Router trong mô hình OSI

Để ngăn chặn việc mất mát số liệu Router còn nhận biết được đường nào có thể chuyển vận và ngừng chuyển vận khi đường bị tắc.

Các lý do sử dụng Router :

- Router có các phần mềm lọc ưu việt hơn là Bridge do các gói tin muốn đi qua Router cần phải gửi trực tiếp đến nó nên giảm được số lượng gói tin qua nó. Router thường được sử dụng trong khi nối các mạng thông qua các đường dây thuê bao đắt tiền do nó không truyền dư lên đường truyền.
- Router có thể dùng trong một liên mạng có nhiều vùng, mỗi vùng có giao thức riêng biệt.
- Router có thể xác định được đường đi an toàn và tốt nhất trong mạng nên độ an toàn của thông tin được đảm bảo hơn.
- Trong một mạng phức hợp khi các gói tin luân chuyển các đường có thể gây nên tình trạng tắc nghẽn của mạng thì các Router có thể được cài đặt các phương thức nhằm tránh được tắc nghẽn.



Hình IV-11: Ví dụ về bảng chỉ đường (Routing table) của Router.

V.3.1 Các phương thức hoạt động của Router

Đó là phương thức mà một Router có thể nối với các Router khác để qua đó chia sẻ thông tin về mạng hiện co. Các chương trình chạy trên Router luôn xây dựng bảng chỉ đường qua việc trao đổi các thông tin với các Router khác.

- **Phương thức véc tơ khoảng cách** : mỗi Router luôn luôn truyền đi thông tin về bảng chỉ đường của mình trên mạng, thông qua đó các Router khác sẽ cập nhật lên bảng chỉ đường của mình.
- **Phương thức trạng thái tĩnh** : Router chỉ truyền các thông báo khi có phát hiện có sự thay đổi trong mạng và chỉ khi đó các Router khác cập nhật lại bảng chỉ đường, thông tin truyền đi khi đó thường là thông tin về đường truyền.

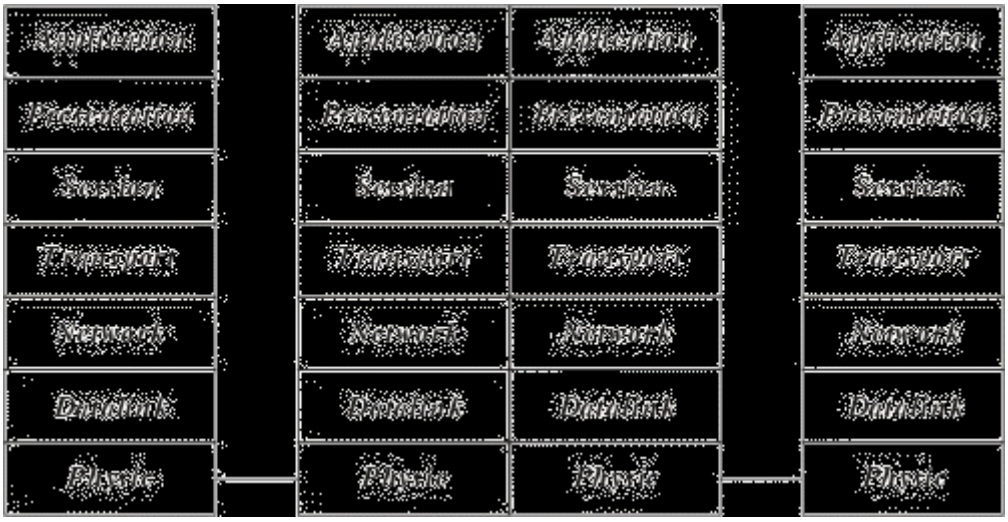
V.3.2 Một số giao thức hoạt động chính của Router

- **RIP (Routing Information Protocol)** được phát triển bởi Xerox Network system và sử dụng SPX/IPX và TCP/IP. RIP hoạt động theo phương thức véc tơ khoảng cách.
- **NLSP (Netware Link Service Protocol)** được phát triển bởi Novell dùng để thay thế RIP hoạt động theo phương thức véctơ khoảng cách, mỗi Router được biết cấu trúc của mạng và việc truyền các bảng chỉ đường giảm đi..
- **OSPF (Open Shortest Path First)** là một phần của TCP/IP với phương thức trạng thái tĩnh, trong đó có xét tới ưu tiên, giá đường truyền, mật độ truyền thông...
- **OSPF-IS (Open System Interconnection Intermediate System to Intermediate System)** là một phần của TCP/IP với phương thức trạng thái tĩnh, trong đó có xét tới ưu tiên, giá đường truyền, mật độ truyền thông...

V.4. Gateway (cổng nối)

Gateway dùng để kết nối các mạng không thuần nhất chẳng hạn như các mạng cục bộ và các mạng máy tính lớn (Mainframe), do các mạng hoàn toàn không thuần nhất nên việc chuyển đổi

thực hiện trên cả 7 tầng của hệ thống mở OSI. Thường được sử dụng nối các mạng LAN vào máy tính lớn. Gateway có các giao thức xác định trước thường là nhiều giao thức, một Gateway đa giao thức thường được chế tạo như các Card có chứa các bộ xử lý riêng và cài đặt trên các máy tính hoặc thiết bị chuyên biệt.



Hình IV-12: Hoạt động của Gateway trong mô hình OSI

Hoạt động của Gateway thông thường phức tạp hơn là Router nên thông suất của nó thường chậm hơn và thường không dùng nối mạng LAN -LAN.

V.5. Hub (Bộ tập trung)

Hub thường được dùng để nối mạng, thông qua những đầu cắm của nó người ta liên kết với các máy tính dưới dạng hình sao.

Người ta phân biệt các Hub thành 3 loại như sau sau :

- **Hub bị động (Passive Hub)** : Hub bị động không chứa các linh kiện điện tử và cũng không xử lý các tín hiệu dữ liệu, nó có chức năng duy nhất là tổ hợp các tín hiệu từ một số đoạn cáp mạng. Khoảng cách giữa một máy tính và Hub không thể lớn hơn một nửa khoảng cách tối đa cho phép giữa 2 máy tính trên mạng (ví dụ khoảng cách tối đa cho phép giữa 2 máy tính của mạng là 200m thì khoảng cách tối đa giữa một máy tính và hub là 100m). Các mạng ARCnet thường dùng Hub bị động.
- **Hub chủ động (Active Hub)** : Hub chủ động có các linh kiện điện tử có thể khuyếch đại và xử lý các tín hiệu điện tử truyền giữa các thiết bị của mạng. Qúa trình xử lý tín hiệu được gọi là tái sinh tín hiệu, nó làm cho tín hiệu trở nên tốt hơn, ít nhạy cảm với lỗi do vậy khoảng cách giữa các thiết bị có thể tăng lên. Tuy nhiên những ưu điểm đó cũng kéo theo giá thành của Hub chủ động cao hơn nhiều so với Hub bị động. Các mạng Token ring có xu hướng dùng Hub chủ động.
- **Hub thông minh (Intelligent Hub)**: cũng là Hub chủ động nhưng có thêm các chức năng mới so với loại trước, nó có thể có bộ vi xử lý của mình và bộ nhớ mà qua đó nó không chỉ cho phép điều khiển hoạt động thông qua các chương trình quản trị mạng mà nó có thể hoạt động như bộ tìm đường hay một cầu nối. Nó có thể cho phép tìm đường

cho gói tin rất nhanh trên các cổng của nó, thay vì phát lại gói tin trên mọi cổng thì nó có thể chuyển mạch để phát trên một cổng có thể nối tới trạm đích.

V.6. Bộ chuyển mạch (*switch)*

Chức năng chính của *switch* là cùng một lúc duy trì nhiều cầu nối giữa các thiết bị mạng bằng cách dựa vào một loại đường truyền xương sống (*backbone*) nội tại tốc độ cao. *Switch* có nhiều cổng, mỗi cổng có thể hỗ trợ toàn bộ *Ethernet* LAN hoặc Token Ring.

Bộ chuyển mạch kết nối một số LAN riêng biệt và cung cấp khả năng lọc gói dữ liệu giữa chúng.

Switch là thiết bị giống như bridge nhưng nhiều port hơn cho phép ghép nối nhiều đoạn mạng với nhau. Switch cũng dựa vào bảng địa chỉ MAC để quyết định gói tin nào đi ra port nào nhằm tránh tình trạng giảm băng thông khi số máy trạm trong mạng tăng lên. Switch cũng hoạt động tại lớp hai trong mô hình OSI. Việc xử lý gói tin dựa trên phần cứng (chip).

Khi một gói tin đi đến Switch (hoặc Bridge), Switch (hoặc Bridge) sẽ thực hiện như sau:

- Kiểm tra địa chỉ nguồn của gói tin đã có trong bảng MAC chưa, nếu chưa có thì nó sẽ thêm địa chỉ MAC này và port nguồn (nơi gói tin đi vào Switch (hoặc Bridge)) vào trong bảng MAC.

- Kiểm tra địa chỉ đích của gói tin đã có trong bảng MAC chưa:

+ Nếu chưa có thì nó sẽ gởi gói tin ra tất cả các port (ngoại trừ port gói tin đi vào).

+ Nếu địa chỉ đích đã có trong bảng MAC:

 Nếu port đích trùng với port nguồn thì Switch (hoặc Bridge) sẽ loại bỏ gói tin.

 Nếu port đích khác với port nguồn thì gói tin sẽ được gởi ra port đích tương ứng.

Chú ý:

- Địa chỉ nguồn và địa chỉ đích được nói ở trên đều là địa chỉ MAC.
- Port nguồn là Port mà gói tin đi vào.
- Port đích là Port mà gói tin đi ra.

Chương V Mô hình mạng

I. Kiến trúc mạng (Topology)

Hình trạng của mạng cục bộ thể hiện qua cấu trúc hay hình dáng hình học cuả các đường dây cáp mạng dùng để liên kết các máy tính thuộc mạng với nhau. Các mạng cục bộ thường hoạt động dựa trên cấu trúc đã định sẵn liên kết các máy tính và các thiết bị có liên quan.

Trước hết chúng ta xem xét hai phương thức nối mạng chủ yếu được sử dụng trong việc liên kết các máy tính là "một điểm - một điểm" và "một điểm - nhiều điểm ".

Với phương thức "một điểm - một điểm" các đường truyền riêng biệt được thiết lập để nối các cặp máy tính lại với nhau. Mỗi máy tính có thể truyền và nhận trực tiếp dữ liệu hoặc có thể làm trung gian như lưu trữ những dữ liệu mà nó nhận được rồi sau đó chuyển tiếp dữ liệu đi cho một máy khác để dữ liệu đó đạt tới đích.

Theo phương thức "một điểm - nhiều điểm " tất cả các trạm phân chia chung một đường truyền vật lý. Dữ liệu được gửi đi từ một máy tính sẽ có thể được tiếp nhận bởi tất cả các máy tính còn lại, bởi vậy cần chỉ ra địa chỉ đích của dữ liệu để mỗi máy tính căn cứ vào đó kiểm tra xem dữ liệu có phải dành cho mình không nếu đúng thì nhận còn nếu không thì bỏ qua.

Hình V-1 Các phương thức liên kết mạng

Tùy theo cấu trúc của mỗi mạng chúng sẽ thuộc vào một trong hai phương thức nối mạng và mỗi phương thức nối mạng sẽ có những yêu cầu khác nhau về phần cứng và phần mềm.

II. Những cấu trúc chính của mạng cục bộ

II.1. Dạng đường thẳng (Bus)

Trong dạng đường thẳng các máy tính đều được nối vào một đường dây truyền chính (bus). Đường truyền chính này được giới hạn hai đầu bởi một loại đầu nối đặc biệt gọi là terminator (dùng để nhận biết là đầu cuối để kết thúc đường truyền tại đây). Mỗi trạm được nối vào bus qua một đầu nối chữ T (T_connector) hoặc một bộ thu phát (transceiver). Khi một trạm truyền dữ liệu, tín hiệu được truyền trên cả hai chiều của đường truyền theo từng gói một, mỗi gói đều phải mang địa chỉ trạm đích. Các trạm khi thấy dữ liệu đi qua nhận lấy, kiểm tra, nếu đúng với địa chỉ của mình thì nó nhận lấy còn nếu không phải thì bỏ qua.

Sau đây là vài thông số kỹ thuật của topology bus. Theo chuẩn IEEE 802.3 (cho mạng cục bộ) với cách đặt tên qui ước theo thông số: tốc độ truyền tính hiệu (1,10 hoặc 100 Mb/s); BASE (nếu là Baseband) hoặc BROAD (nếu là Broadband).

- 10BASE5: Dùng cáp đồng trục đường kính lớn (10mm) với trở kháng 50 Ohm, tốc độ 10 Mb/s, phạm vi tín hiệu 500m/segment, có tối đa 100 trạm, khoảng cách giữa 2 tranceiver tối thiểu 2,5m (Phương án này còn gọi là Thick Ethernet hay Thicknet)

- 10BASE2: tương tự như Thicknet nhưng dùng cáp đồng trục nhỏ (RG 58A), có thể chạy với khoảng cách 185m, số trạm tối đa trong 1 segment là 30, khoảng cách giữa hai máy tối thiểu là 0,5m.

Dạng kết nối này có ưu điểm là ít tốn dây cáp, tốc độ truyền dữ liệu cao tuy nhiên nếu lưu lượng truyền tăng cao thì dễ gây ách tắc và nếu có trục trặc trên hành lang chính thì khó phát hiện ra.

Hiện nay các mạng sử dụng hình dạng đường thẳng là mạng Ethernet và G-net.

II.2. Dạng vòng tròn (Ring)

Các máy tính được liên kết với nhau thành một vòng tròn theo phương thức "một điểm - một điểm ", qua đó mỗi một trạm có thể nhận và truyền dữ liệu theo vòng một chiều và dữ liệu được truyền theo từng gói một. Mỗi gói dữ liệu đều có mang địa chỉ trạm đích, mỗi trạm khi nhận được một gói dữ liệu nó kiểm tra nếu đúng với địa chỉ của mình thì nó nhận lấy còn nếu không phải thì nó sẽ phát lại cho trạm kế tiếp, cứ như vậy gói dữ liệu đi được đến đích. Với dạng kết nối này có ưu điểm là không tốn nhiều dây cáp, tốc độ truyền dữ liệu cao, không gây ách tắc tuy nhiên các giao thức để truyền dữ liệu phức tạp và nếu có trục trặc trên một trạm thì cũng ảnh hưởng đến toàn mạng.

Hiện nay các mạng sử dụng hình dạng vòng tròn là mạng Tocken ring của IBM.

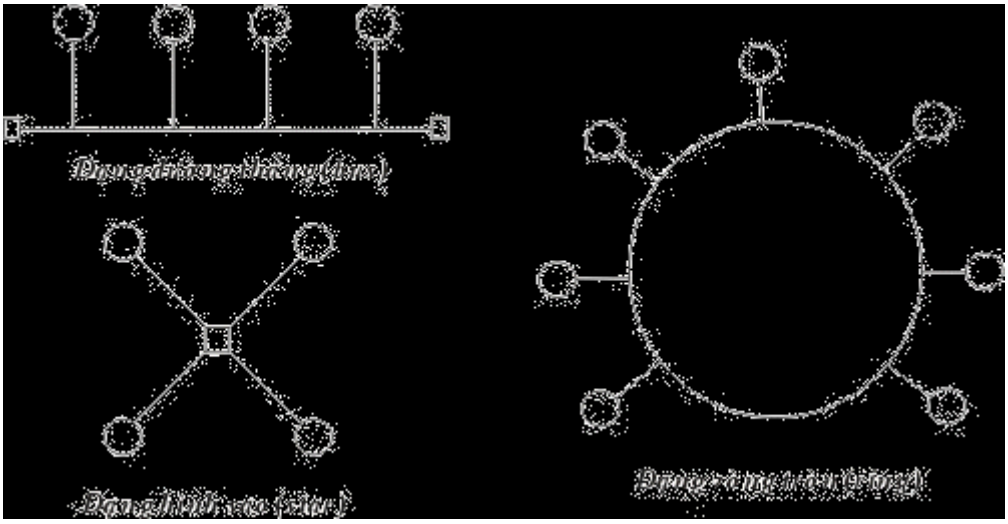
II.3. Dạng hình sao (Star)

Ở dạng hình sao, tất cả các trạm được nối vào một thiết bị trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển tín hiệu đến trạm đích với phương thức kết nối là phương thức "một điểm - một điểm ". Thiết bị trung tâm hoạt động giống như một tổng đài cho phép thực hiện việc nhận và truyền dữ liệu từ trạm này tới các trạm khác. Tùy theo yêu cầu truyền thông trong mạng , thiết bị trung tâm có thể là một bộ chuyển mạch (switch), một bộ chọn đường (router) hoặc đơn giản là một bộ phân kênh (Hub). Có nhiều cổng ra và mỗi cổng nối với một máy. Theo chuẩn IEEE 802.3 mô hình dạng Star thường dùng:

- 10BASE-T: dùng cáp UTP, tốc độ 10 Mb/s, khoảng cách từ thiết bị trung tâm tới trạm tối đa là 100m.
- 100BASE-T tương tự như 10BASE-T nhưng tốc độ cao hơn 100 Mb/s.

Ưu và khuyết điểm

- Ưu điểm: Với dạng kết nối này có ưu điểm là không đụng độ hay ách tắc trên đường truyền, lắp đặt đơn giản, dễ dàng cấu hình lại (thêm, bớt trạm). Nếu có trục trặc trên một trạm thì cũng không gây ảnh hưởng đến toàn mạng qua đó dễ dàng kiểm soát và khắc phục sự cố.
- Nhược điểm: Độ dài đường truyền nối một trạm với thiết bị trung tâm bị hạn chế (trong vòng 100 m với công nghệ hiện đại) tốn đường dây cáp nhiều, tốc độ truyền dữ liệu không cao.
- Hiện nay các mạng sử dụng hình dạng hình sao là mạng STARLAN của AT&T và S-NET của Novell.



Hình V-2 Các loại cấu trúc chính của mạng cục bộ.

	Đường thẳng	**Vòng Tròn**	**Hình sao**
Ứng dụng	Tốt cho trường hợp mạng nhỏ và mạng có giao thông thấp và lưu lượng dữ liệu thấp	Tốt cho trường hợp mạng có số trạm ít hoạt động với tốc độ cao,không cách nhau xa lắm hoặc mạng có lưu lượng dữ liệu phân bố không đều.	Hiện nay mạng sao là cách tốt nhất cho trường hợp phải tích hợp dữ liệu và tín hiệutiếng.Các mạng đện thoại công cộng có cấu trúc này
Độ phức tạp	Tương đối không phức tạp	Đòi hỏi thiết bị tương đối phức tạp .Mặt khác việc đưa thông điệp đi trên tuyến là đơn giản, vì chỉ có 1 con đường, trạm phát chỉ cần biết địa chỉ của trạm nhận , các thông tin để dẫn đường khác thì không cần thiết	Mạng sao được xem là khá phức tạp . Các trạm được nối với thiết bị trung tâm và lần lượt hoạt động như thiết bị trung tâm hoặc nối được tới các dây dẫn truyền từ xa
Hiệu suất	Rất tốt dưới tải thấp có thể giảm hiệu suất rất mau khi tải tăng	Có hiệu quả trong trường hợp lượng lưu thông cao và khá ổn định nhờ sự tăng chậm thời gian trễ và sự xuống cấp so với các mạng khác	Tốt cho trường hợp tải vừa tuy nhiên kích thước và khả năng , suy ra hiệu suất của mạng phụ thuộc trực tiếp vào sức mạnh của thiết bị trung tâm.
Tổng phí	Tương đối thấp đặc biệt do nhiều thiết bị đã phát triển hòa chỉnh và bán sảm phẩm ở thị trường	Phải dự trù gấp đôi nguồn lực hoặc phải có 1 phương thức thay thế khi 1 nút không hoạt động nếu vẫn	Tổng phí rất cao khi làm nhiêm vụ của thiết bị trung tâm, thiết bị trung tâm ï không được dùng

	.Sự dư thừa kênh truyền được khuyến để giảm bớt nguy cơ xuất hiện sự cố trên mạng	muốn mạng hoạt động bình thường	vào việc khác .Số lượng dây riêng cũng nhiều.
Nguy cơ	Một trạm bị hỏng không ảnh hưởng đến cả mạng. Tuy nhiên mạng sẽ có nguy cơ bị tổn hại khi sự cố trên đường dây dẫn chính hoặc có vấn đề với tuyến. Vấn đề trên rất khó xác định được lại rất dễ sửa chữa	Một trạm bị hỏng có thể ảnh hưởng đến cả hệ thống vì các trạm phục thuộc vào nhau. Tìm 1 repeater hỏng rất khó ,vả lại việc sửa chữa thẳng hay dùng mưu mẹo xác định điểm hỏng trên mạng có địa bàn rôäng rất khó	Độ tin cậy của hệ thống phụ thuộc vào thiết bị trung tâm, .nếu bị hỏng thì mạng ngưng hoạt động Sự ngưng hoạt động tại thiết bị trung tâm thường không ảnh hưởng đến toàn bộ hệ thống .
Khả năng mở rộng	Việc thêm và định hình lại mạng này rất dễ.Tuy nhiên việc kết nối giữa các máy tính và thiết bị của các hãng khác nhau khó có thể vì chúng phải có thể nhận cùng địa chỉ và dữ liệu	Tương đối dễ thêm và bớt các trạm làm việc mà không phải nối kết nhiều cho mỗi thay đổi Giá thành cho việc thay đổi tương đối thấp	Khả năng mở rộâng hạn chế, đa số các thiết bị trung tâm chỉ chịu đựng nổi 1 số nhất định liên kết. Sự hạn chế về tốc độ truyền dữ liệu và băng tần thường được đòi hỏi ở mỗi người sử dụng. Các hạn chế này giúp cho các chức năng xử lý trung tâm không bị quá tải bởi tốc độ thu nạp tại tại cổng truyền và giá thành mỗi cổng truyền của thiết bị trung tâm thấp .

Bảng V-1 ***Bảng so sánh tính năng giữa các cấu trúc của mạng LAN***

II.4. Mạng dạng kết hợp

- Kết hợp hình sao và tuyến (*star/Bus Topology*)

Cấu hình mạng dạng này có bộ phận tách tín hiệu (*spitter*) giữ vai trò thiết bị trung tâm, hệ thống dây cáp mạng có thể chọn hoặc *Ring Topology* hoặc *Linear Bus Topology.*

Lợi điểm của cấu hình này là mạng có thể gồm nhiều nhóm làm việc ở cách xa nhau, ARCNET là mạng dạng kết hợp *Star/Bus Topology*. Cấu hình dạng này đưa lại sự uyển chuyển trong việc bố trí đường dây tương thích dễ dàng đối với bất cứ toà nhà nào.

- Kết hợp hình sao và vòng *(Star/Ring Topology)*

Cấu hình dạng kết hợp *Star/Ring Topology*, có một "thẻ bài" liên lạc (*Token*) được chuyển vòng quanh một cái HUB trung tâm. Mỗi trạm làm việc (*workstation*) được nối với HUB - là cầu nối giữa các trạm làm việc và để tăng khoảng cách cần thiết.

Chương VI Các dịch vụ của mạng diện rộng (WAN)

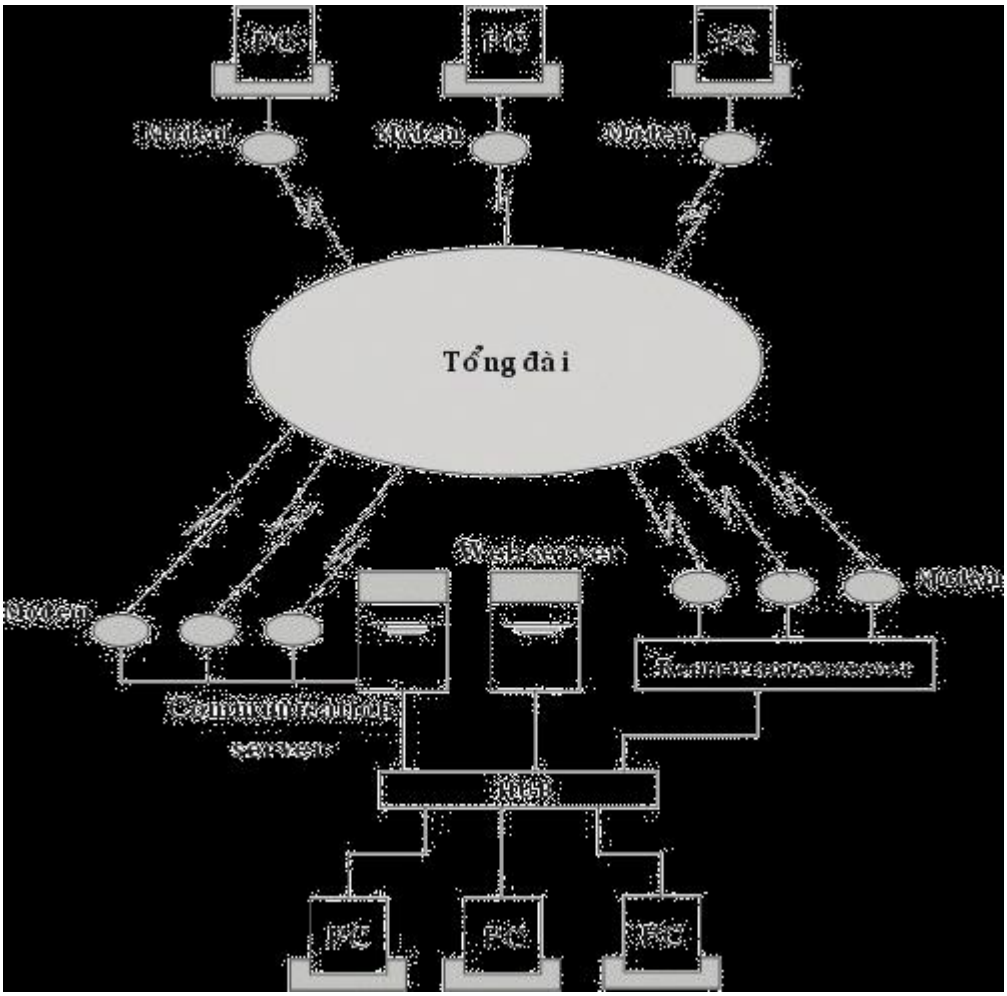
Hiện nay trên thế giới có nhiều dịch vụ dành cho việc chuyển thông tin từ khu vực này sang khu vực khác nhằm liên kết các mạng LAN của các khu vực khác nhau lại. Để có được những liên kết như vậy người ta thường sử dụng các dịch vụ của các mạng diện rộng. Hiện nay trong khi giao thức truyền thông cơ bản của LAN là Ethernet, Token Ring thì giao thức dùng để tương nối các LAN thông thường dựa trên chuẩn TCP/IP. Ngày nay khi các dạng kết nối có xu hướng ngày càng đa dạng và phân tán cho nên các mạng WAN đang thiên về truyền theo đơn vị tập tin thay vì truyền một lần xử lý.

Có nhiều cách phân loại mạng diện rộng, ở đây nếu phân loại theo phương pháp truyền thông tin thì có thể chia thành 3 loại mạng như sau:

- Mạng chuyển mạch (Circuit Swiching Network)
- Mạng thuê bao (Leased lines Network)
- Mạng chuyển gói tin (Packet Switching Network)

I. Mạng chuyển mạch (Circuit Swiching Network)

Để thực hiện được việc liên kết giữa hai điểm nút, một đường nối giữa điểm nút này và điêm nút kia được thiết lập trong mạng thể hiện dưới dạng cuộc gọi thông qua các thiết bị chuyển mạch.



Hình VI-1*: Mô hình mạng chuyển mạch*

Một ví dụ của mạng chuyển mạch là hoạt động của mạng điện thoại, các thuê bao khi biết số của nhau có thể gọi cho nhau và có một đường nối vật lý tạm thời được thiết lập giữa hai thuê bao.

Với mô hình này mọi đường đều có thể một đường bất kỳ khác, thông qua những đường nối và các thiết bị chuyên dùng người ta có thể liên kết một đường tạm thời từ nơi gửi tới nơi nhận một đường nối vật lý, đường nối trên duy trì trong suốt phiên làm việc và chỉ giải phóng sau khi phiên làm việc kết thúc. Để thực hiện một phiên làm việc cần có các thủ tục đầy đủ cho việc thiết lập liên kết trong đó có việc thông báo cho mạng biết địa chỉ của nút nhận.

Hiện nay có 2 loại mạng chuyển mạch là chuyển mạch tương tự (analog) và chuyển mạch số (digital)

- *Chuyển mạch tương tự (Analog):* Việc chuyển dữ liệu qua mạng chuyển mạch tương tự được thực hiện qua mạng điện thoại. Các trạm sử dụng một thiết bị có tên là modem, thiết bị này sẽ chuyển các tín hiệu số từ máy tính sao tín hiệu tuần tự có trễ truyền đi trên mạng điện thoại và ngược lại.

Hình VI-2*: Mô hình chuyển mạch tương tự*

Khi sử dụng đường truyền điện thoại để truyền số liệu thì các chuẩn của modem và các tính chất của nó sẽ quyết định tốc độ của đường truyền. Cùng với các kỹ thuật chuyển đổi tín hiệu các tính năng mới như nén tín hiệu cho phép nâng tốc độ truyền dữ liệu lên rất cao.

Loại	Tốc độ(bps)	Loại nén	Tốc độ thực tế (bps)
Bell 212A	1200		
CCITT V22	1200		
CCITT V22 bis	2400	MNP Class 5	2400 - 3600
CCITT V32	9600	MNP Class 5, V42 bis	9600 - 19200
CCITT V32 bis	14400	MNP Class 5, V42 bis	14400 - 33600

Bảng VI-1*: Bảng kỹ thuật modem*

Các kỹ thuật nén thường dùng là MNP Class 5 và V42 bis, MNP Class 5 cho phép nén với tỷ lệ 1.5:1 và V42 bis nén với tỷ lệ 2:1. Tuy nhiên trên thực tế tỷ lệ nén có thể thay đổi dựa vào dạng dữ liệu được truyền.

- *Chuyển mạch số (Digital):* Đường truyền chuyển mạch số lần đầu tiên được AT&T thiệu vào cuối 1980 khi AT&T giới thiệu mạng chuyển mạch số Acnet với đường truyền 56 kbs. Việc sử dụng đường chuyển mạch số cũng đòi hỏi sử dụng thiết bị phục vụ truyền dữ liệu số (Data Service Unit - DSU) vào vị trí modem trong chuyển mạch tương tự. Thiết bị phục vụ truyền dữ liệu số có nhiệm vụ chuyển các tín hiệu số đơn chiều (unipolar) từ máy tính ra thành tín hiệu số hai chiều (bipolar) để truyền trên đường truyền.

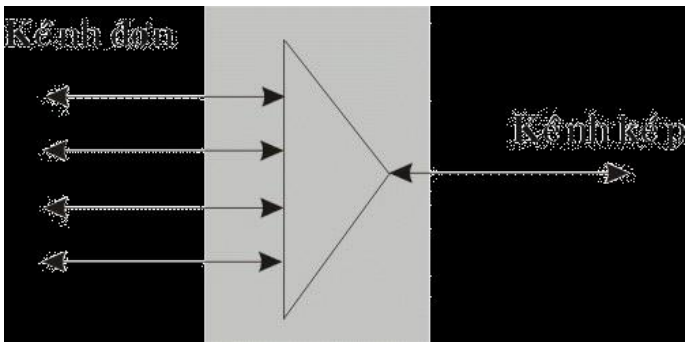
Hình VI-3*: Mô hình chuyển mạch số*

Mạng chuyển mạch số cho phép người sử dụng nâng cao tốc độ truyền (ở đây do khác biệt giữa kỹ thuật truyền số và kỹ thuật truyền tương tự nên hiệu năng của truyền mạch số cao hơn nhiều so với truyền tương tự cho dù cùng tốc độ), độ an toàn.

Vào năm 1991 AT&T giới thiệu mạng chuyển mạch số có tốc độ 384 Kbps. Người ta có thể dùng mạng chuyển mạch số để tạo các liên kết giữa các mạng LAN và làm các đường truyền dự phòng.

II. Mạng thuê bao (Leased line Network)

Với kỹ thuật chuyển mạch giữa các nút của mạng (tương tự hoặc số) có một số lượng lớn đường dây truyền dữ liệu, với mỗi đường dây trong một thời điểm chỉ có nhiều nhất một phiên giao dịch, khi số lượng các trạm sử dụng tăng cao người ta nhận thấy việc sử dụng mạng chuyển mạch trở nên không kinh tế. Để giảm bớt số lượng các đường dây kết nối giữa các nút mạng người ta đưa ra một kỹ thuật gọi là ghép kênh.



Hình VI-4*: Mô hình ghép kênh*

Mô hình đó được mô tả như sau: tại một nút người ta tập hợp các tín hiệu trên của nhiều người sử dụng ghép lại để truyền trên một kênh nối duy nhất đến các nút khác, tại nút cuối người ta phân kênh ghép ra thành các kênh riêng biệt và truyền tới các người nhận.

Có hai phương thức ghép kênh chính là ghép kênh theo tần số và ghép kênh theo thời gian, hai phương thức này tương ứng với mạng thuê bao tuần tự và mạng thuê bao kỹ thuật số. trong thời gian hiện nay mạng thuê bao kỹ thuật số sử dụng kỹ thuật ghép kênh

theo thời gian với đường truyền T đang được sử dụng ngày một rộng rãi và dần dần thay thế mạng thuê bao tuần tự.

1. **Phương thức ghép kênh theo tần số**

Để sử dụng phương thức ghép kênh theo tần số giữa các nút của mạng được liên kết bởi đường truyền băng tần rộng. Băng tần này được chia thành nhiều kênh con được phân biệt bởi tần số khác nhau. Khi truyền dử liệu, mỗi kênh truyền từ người sử dụng đến nút sẽ được chuyển thành một kênh con với tần số xác định và được truyền thông qua bộ ghép kênh đến nút cuối và tại đây nó được tách ra thành kênh riêng biệt để truyền tới người nhận. Theo các chuẩn của CCITT có các phương thức ghép kênh cho phép ghép 12, 60, 300 kênh đơn.

Người ta có thể dùng đường thuê bao tuần tự (Analog) nối giữa máy của người sử dụng tới nút mạng thuê bao gần nhất. Khi máy của người sử dụng gửi dữ liệu thì kênh dữ liệu được ghép với các kênh khác và truyền trên đường truyền tới nút đích và được phân ra thành kênh riêng biệt trước khi gửi tới máy của người sử dụng. Đường nối giữa máy trạm của người sử dụng tới nút mạng thuê bao cũng giống như mạng chuyển mạch tuần tự sử dụng đường dây điện thoại với các kỹ thuật chuyển đổi tín hiệu như V22, V22 bis, V32, V32 bis, các kỹ thuật nén V42 bis, MNP class 5.

2. **Phương thức ghép kênh theo thời gian:**

Khác với phương thức ghép kênh theo tần số, phương thức ghép kênh theo thời gian chia một chu kỳ thời gian hoạt động của đường truyền trục thành nhiều khoảng nhỏ và mỗi kênh tuyền dữ liệu được một khoảng. Sau khi ghép kênh lại thành một kênh chung dữ liệu được truyền đi tương tự như phương thức ghép kênh theo tần số. Người ta dùng đường thuê bao là đường truyền kỹ thuật số nối giữa máy của người sử dụng tới nút mạng thuê bao gần nhất.

Hiện nay người ta có các đường truyền thuê bao như sau :

Đường T1 với tốc độ 1.544 Mbps nó bao gồm 24 kênh vớp tốc độ 64 kbps và 8000 bits điều khiển trong 1 giây.

III. Mạng chuyển gói tin (Packet Switching NetWork)

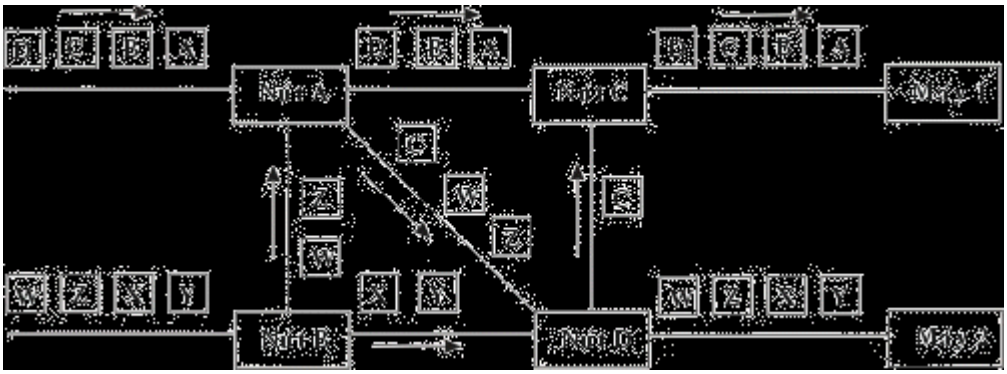
Mạng chuyển mạch gói hoạt động theo nguyên tắc sau : Khi một trạm trên mạng cần gửi dữ liệu nó cần phải đóng dữ liệu thành từng gói tin, các gói tin đó được đi trên mạng từ nút này tới nút khác tới khi đến được đích. Do việc sử dụng kỹ thuật trên nên khi một trạm không gửi tin thì mọi tài nguyên của mạng sẽ dành cho các trạm khác, do vậy mạng tiết kiệm được các tài nguyên và có thể sử dụng chúng một cách tốt nhất.

Người ta chia các phương thức chuyển mạch gói ra làm 2 phương thức:

Phương thức chuyển mạch gói theo sơ đồ rời rạc.
Phương thức chuyển mạch gói theo đường đi xác định.

1. **Phương thức chuyển mạch gói theo sơ đồ rời rạc:**

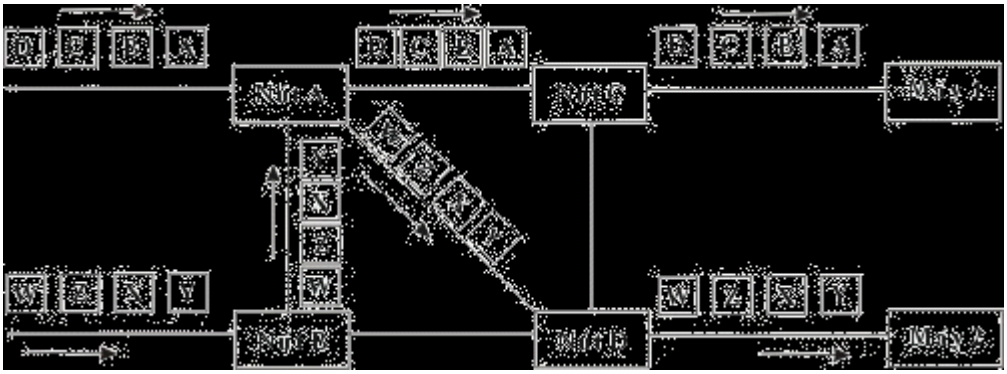
các gói tin được chuyển đi trên mạng một cách độc lập, mỗi gói tin đều có mang địa chỉ nơ i gửi và nơi nhận. Mỗi nút trong mạng khi tiếp nhận gói tin sẽ quyết định xenm đường đi của gói tin phụ thuộc vào thuật toán tìm đường tại nút và những thông tin về mạng mà nút đó có. Việc truyền theo phương thức này cho ta sự mềm dẻo nhất định do đường đi với mỗi gói tin trở nên mềm dẻo tuy nhiên điều này yêu cầu một số lượng tính toán rất lớn tại mỗi nút nên hiện nay phần lớn các mạng chuyển sang dùng phương chuyển mạch gói theo đường đi xác định.



Hình VI-5*: Ví dụ phương thức sơ đồ rời rạc.*

2. **Phương thức chuyển mạch gói theo đường đi xác định:**

Trước khi truyền dữ liệu một đường đi (hay còn gọi là đường đi ảo) được thiết lập giữa trạm gửi và trạm nhận thông qua các nút của mạng. Đường đi trên mang số hiệu phân biệt với các đường đi khác, sau đó các gói tin được gửi đi theo đường đã thiết lập để tới đích, các gói tin mang số hiệu củ đường ảo để có thể được nhận biết khi qua các nút. Điều này khiến cho việc tính toán đường đi cho phiên liên lạc chỉ cần thực hiện một lần.



Hình VI-6*: Ví dụ phương thức đường đi xác định*

IV. Mạng X25

Được CCITT công bố lần đầu tiên vào 1970 lúc lĩnh vực viễn thông lần đầu tiên tham gia vào thế giới truyền dữ liệu với các đặc tính:

- X25 cung cấp quy trình kiểm soát luồng giữa các đầu cuối đem lại chất lương đường truyền cao cho dù chất lương đương dây truyền không cao.

- X25 được thiết kế cho cả truyền thông chuyển mạch lẫn truyền thông kiểu điểm nối điểm.
- Được quan tâm và tham gia nhanh chóng trên toàn cầu.

Trong X25 có chức năng dồn kênh (multiplexing) đối với liên kết logic (virtual circuits) chỉ làm nhiệm vụ kiểm soát lỗi cho các frame đi qua. Điều này làm tăng độ phức tạp trong việc phối hợp các thủ tục giữa hai tầng kề nhau, dẫn đến thông lượng bị hạn chế do tổng phí xử lý mỗi gói tin tăng lên. X25 kiểm tra lỗi tại mỗi nút trước khi truyền tiếp, điều này làm cho đường truyền chó chất lượng rất cao gần như phi lỗi. Tuy nhiên do vậy khối lượng tích toán tại mỗi nút khá lớn, đối với những đường truyền của những năm 1970 thì điều đó là cần thiết nhưng hiện nay khi kỹ thuật truyền dẫn đã đạt được những tiến bộ rất cao thì việc đó trở nên lãng phí

V. Mạng Frame Relay

Mỗi gói tin trong mạng gọi là Frame, do vậy mạng gọi là Frame relay. Đặc điểm khác biệt giữa mạng Frame Relay và mạng X25 mạng Frame Relay là chỉ kiểm tra lỗi tại hai trạm gửi và trạm nhận còn trong quá trình chuyển vận qua các nút trung gian gói tin sẽ không được kiểm lỗi nữa. Do vậy thời gian xử lý trên mỗi nút nhanh hơn, tuy nhiên khi có lỗi thì gói tin phải được phát lại từ trạm đầu. Với độ an toàn cao của đường truyền hiện nay thì chi phí việc phát lại đó chỉ chiếm một tỷ lệ nhỏ nếu so với khối lượng tính toán được giảm đi tại các nút nên mạng Frame Relay tiết kiệm được tài nguyên của mạng hơn so với mạng X25.

Frame relay không chỉ là một kỹ thuật mà còn là thể hiện một phương pháp tổ chức mới. Với nguyên lý là truyền mạch gói nhưng các thao tác kiểm soát giữa các đầu cuối giảm đáng kể Kỹ thuật Frame Relay cho phép thông lượng tối đa đạt tới 2Mbps và hiện nay nó đang cung cấp các giải pháp để tương nối các mạng cục bộ LAN trong một kiến trúc xương sống tạo nên môi trường cho ứng dụng multimedia.

VI. Mạng ATM (Cell relay)

Hiện nay kỹ thuật Cell Relay dựa trên phương thức truyền thông không đồng bộ (ATM) có thể cho phép thông lương hàng trăm Mbps. Đơn vị dữ liệu dùng trong ATM được gọi là tế bào (cell). các tế bào trong ATM có độ dài cố định là 53 bytes, trong đó 5 bytes dành cho phần chứa thông tin điều khiển (cell header) và 48 bytes chứa dữ liệu của tầng trên.

Trong kỹ thuật ATM, các tế bào chứa các kiểu dữ liệu khác nhau được ghép kênh tới một đường dẫn chung được gọi là đường dẫn ảo (virtual path). Trong đường dẫn ảo đó có thể gồm nhiều kênh ảo (virtual chanell) khác nhau, mỗi kênh ảo được sử dụng bởi một ứng dung nào đó tại một thời điểm.

ATM đã kết hợp những đặc tính tốt nhất của dạng chuyển mạch liên tục và dạng chuyển mạch gói, nó có thể kết hợp dải thông linh hoạt và khả năng chuyển tiếp cao tốc và có khả năng quản lý đồng thời dữ liệu số, tiếng nói, hình ảnh và multimedia tương tác.

Mục tiêu của kỹ thuật ATM là nhằm cung cấp một mạng dồn kênh, và chuyển mạch tốc độ cao, độ trễ nhỏ dáp ứng cho các dạng truyền thông đa phương tiện (multimecdia)

Chuyển mạch cell cần thiết cho việc cung cấp các kết nối đòi hỏi băng thông cao, tình trạng tắt nghẽn thấp, hổ trợ cho lớp dịch vụ tích hợp lưu thông dữ liệu âm thanh hình ảnh. Đặc tính tốc độ cao là đặc tính nổi bật nhất của ATM.

ATM sử dụng cơ cấu chuyển mạch đặc biệt: ma trận nhị phân các thành tố chuyển mạch (a matrix of binary switching elements) để vận hành lưu thông. Khả năng vô hướng (scalability) là một đặc tính của cơ cấu chuyển mạch ATM. Đặc tính này tương phản trực tiếp với những gì diễn ra khi các trạm cuối được thêm vào một thiết bị liên mạng như router. Các router có năng suất tổng cố định được chia cho các trạm cuối có kết nối với chúng. Khi số lượng trạm cuối gia tăng, năng suất của router tương thích cho trạm cuối thu nhỏ lại. Khi cơ cấu ATM mở rộng, mỗi thiết bị thu trạm cuối, bằng con đường của chính nó đi qua bộ chuyển mạch bằng cách cho mỗi trạm cuối băng thông chỉ định. Băng thông rộng được chỉ định của ATM với đặc tính có thể xác nhận khiến nó trở thành một kỹ thuật tuyệt hảo dùng cho bất kỳ nơi nào trong mạng cục bộ của doanh nghiệp.

Như tên gọi của nó chỉ rõ, kỹ thuật ATM sử dụng phương pháp truyền không đồng bộ (asynchronouns) các tề bào từ nguồn tới đích của chúng. Trong khi đó, ở tầng vật lý người ta có thể sử dụng các kỹ thuật truyền thông đồng bộ như SDH (hoặc SONET).

Nhận thức được vị trí chưa thể thay thế được (ít nhất cho đến những năm đầu của thế kỷ 21) của kỹ thuật ATM, hầu hết các hãng khổng lồ về máy tính và truyền thông như IBM, ATT, Digital, Hewlett - Packard, Cisco Systems, Cabletron, Bay Network,... đều đang quan tâm đặc biệt đến dòng sản phẩm hướng đến ATM của mình để tung ra thị trường. Có thể kể ra đây một số sản phẩm đó như DEC 900 Multiwitch, IBM 8250 hub, Cisco 7000 rounter, Cablectron, ATM module for MMAC hub.

Nhìn chung thị trường ATM sôi động do nhu cầu thực sự của các ứng dụng đa phương tiện. Sự nhập cuộc ngày một đông của các hãng sản xuất đã làm giảm đáng kể giá bán của các sản phẩm loại này, từ đó càng mở rộng thêm thị trường. Ngay ở Việt Nam, các dự án lớn về mạng tin học đều đã được thiết kế với hạ tầng chấp nhận được với công nghệ ATM trong tương lai.

Chương VII CÁC DỊCH VỤ MẠNG THÔNG DỤNG

I. DỊCH VỤ WEB

I.1. Một số thuật ngữ cơ bản.

- **HTTP (Hypertext Transfer Protocol)**: là giao thức cho phép các máy tính giao tiếp qua Web và kết nối với nhau qua các siêu liên kết hyperlink.
- **HTML (Hypertext Markup Language)**: là ngôn ngữ định dạng dùng để tạo ra các trang Web giúp người dùng có thể đọc và truy cập từ bất kỳ máy nào trên mạng, dùng bất kỳ hệ điều hành nào.
- **WebPage**: là một trang tư liệu Web.
- **WebSite**: là tập hợp các trang Web của một tổ chức, một công ty, một web site có thể có nhiều Web Server.
- **Home page**: là trang Web đầu tin của một Web Site hoặc trang Web xuất hiện đầu tin khi khởi động Web Browser, đồng thời trang này chứa các liên kết tiêu biểu đến các trang Web còn lại.
- **HyperLink (link)**: là các mối liên kết giữa các tư liệu. Thông thường, trong một trang Web, các mối liên kết có màu xanh dương và được gạch dưới. Ngoài ra, bất kỳ một hình ảnh, văn bản nào khi di chuyển con trỏ chuột tới chuyển sang hình đều là các liên kết (link).
- **URL (Uniform Resource Locator)**: là đường dẫn chỉ tới một tập tin trong một máy chủ trên Internet. Chuỗi URL thường bao gồm: tên giao thức, tên máy chủ và đường dẫn đến tập tin trong máy chủ đó.

  Ví dụ: http://www.mait.vn/index.htm có nghĩa là: giao thức sử dụng http:// (Hypertext Transfer Protocol), tên máy chủ: www.mait.vn, đường dẫn và tên tập tin: index.htm. Lưu ý: đường dẫn sử dụng dấu "/" thay cho dấu "\".
- **IXP (Internet Exchange Provider)**: là nhà cung cấp đường truyền và cổng truy cập Internet.
- **ISP (Internet Service Provider)**: là nhà cung cấp dịch vụ Internet cho người dùng trực tiếp qua mạng điện thoại như là cấp quyền truy cập Internet, cung cấp các dịch vụ như Web, E-mail, Chat, Telnet…
- **ICP (Internet Content Provider)**: là nhà cung cấp thông tin lên Internet, thông tin được cập nhật định kỳ hay thường xuyên và thuộc nhiều lĩnh vực như thể thao, kinh tế giáo dục, chính trị, quân sự …

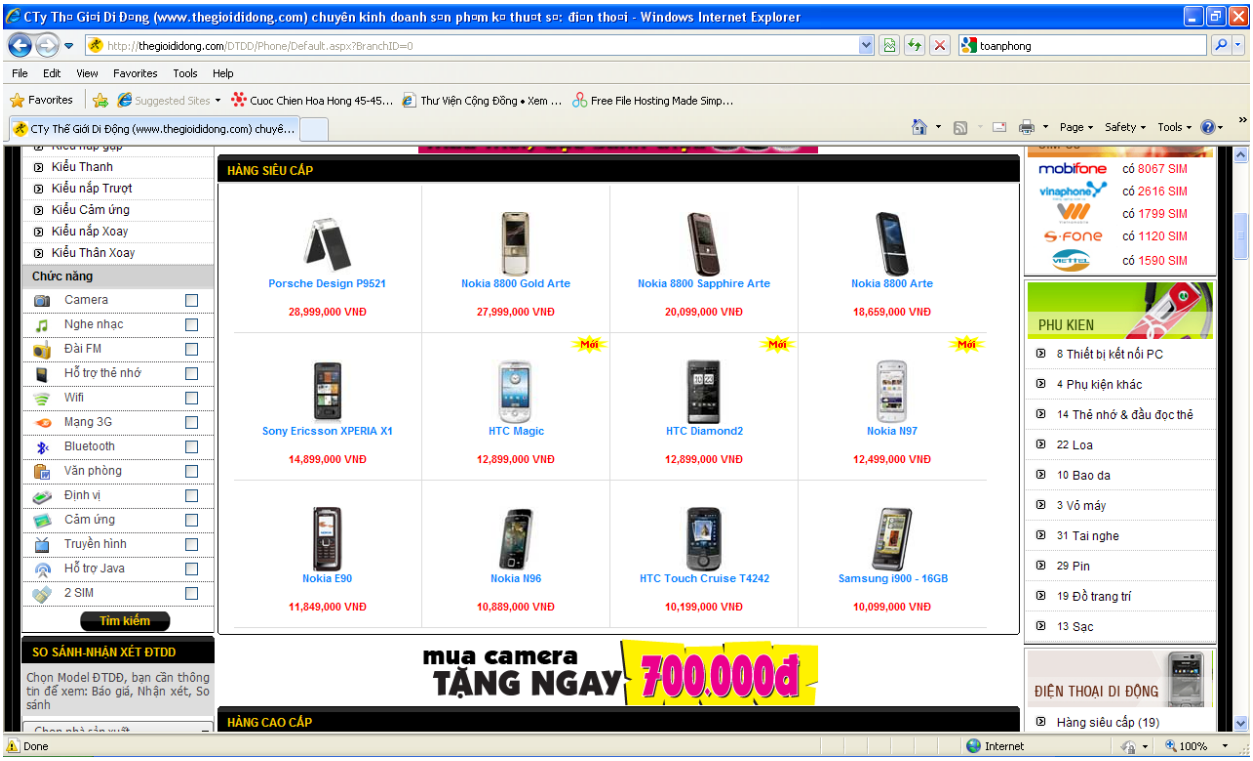
Các hoạt động chính trên Web.

- Duyệt Web tìm kiếm thông tin như số điện thoại, địa chỉ nhà, tin tức, tin dự báo thời tiết, bảng giá chứng khoán, các phần mềm miễn phí…
- Giải trí như nghe nhạc,xem phim, chơi game trên mạng.
- Trao đổi E-mail.
- Truy xuất và download các tập tin.
- Trao đổi thông tin (forum).
- Sắp xếp các chuyến đi du lịch như đặt vé máy bay, đăng ký phòng khách sạn...
- Giao dịch mua bán hàng qua mạng.

Hình VII-1 Minh họa truy cập trang Web để tìm kiếm thông tin.

Hình VII-2 : Minh họa một trang Web dùng để đào tạo trực tuyến.

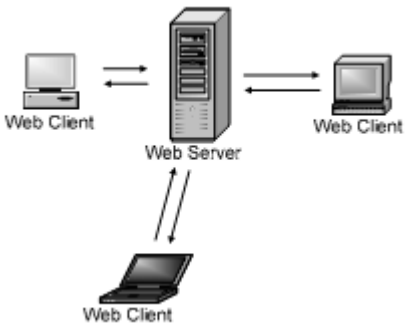
- Hội thảo từ xa.
- Quảng cáo sản phẩm.



Hình VII-3 Minh họa Website giới thiệu sản phẩm

I.2. Giới thiệu mô hình hoạt động của Web.

Dịch vụ World Wide Web (viết tắt là www hoặc Web) là một dịch vụ cung cấp thông tin trên hệ thống mạng. Các thông tin này được lưu trữ dưới dạng siêu văn bản (hypertext) và thường được thiết kế bằng ngôn ngữ HTML (Hypertext Markup Language). Siêu văn bản là các tư liệu có thể là văn bản (text), hình ảnh tĩnh (image), hình ảnh động (video), âm thanh (audio)...., được liên kết với nhau qua các mối liên kết (link) và được truyền trên mạng dựa trên giao thức HTTP (Hypertext Transfer Protocol), qua đó người dùng có thể xem các tư liệu có liên quan một cách dễ dàng. Mô hình hoạt động:



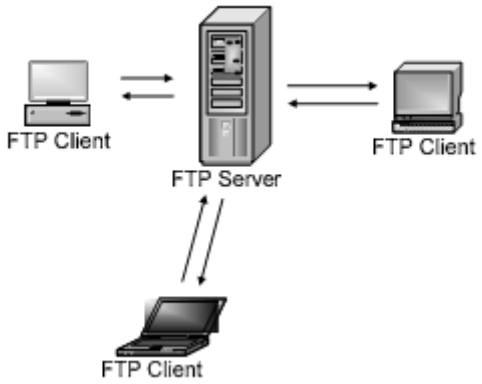
Web server: là một ứng dụng được cài đặt trên máy chủ trên mạng với chức năng là tiếp nhận các yêu cầu dạng HTTP từ máy trạm và tùy theo yêu cầu này máy chủ sẽ cung cấp cho máy trạm các thông tin web dạng HTML.

Web Client: là một ứng dụng cài trên máy trạm (máy của người dùng đầu cuối) gọi là Web Browser để gởi yêu cầu đến Web Server và nhận các thông tin phản hồi rồi hiện lên màn hình giúp người dùng có thể truy xuất được các thông tin trên máy Server. Một trong những trình duyệt Web (Web Browser) phổ biến nhất hiện nay là Internet Explorer.

II. DỊCH VỤ FTP

II.1. Mô hình hoạt động của FTP

FTP (File Transfer Protocol) là một dịch vụ cho phép ta truyền tải file giữa hai máy tính ở xa dùng giao thức TCP/IP. FTP cũng là một ứng dụng theo mô hình client-server, nghĩa là máy làm FTP Server sẽ quản lý các kết nối và cung cấp dịch vụ tập tin cho các máy trạm. Nói tóm lại FTP Server thường là một máy tính phục vụ cho việc quảng bá các tập tin cho người dùng hoặc là một nơi cho phép người dùng chia sẻ tập tin với những người dùng khác trên Internet. Máy trạm muốn kết nối vào FTP Server thì phải được Server cấp cho một account có đầy đủ các thông tin như: địa chỉ máy Server (tên hoặc địa chỉ IP), username và password. Phần lớn các FTP Server cho phép các máy trạm kết nối vào mình thông qua account anonymous (account anonymous thường được truy cập với password rỗng). Các máy trạm có thể sử dụng các lệnh ftp đã tích hợp sẵn trong hệ điều hành hoặc phần mềm chuyên dụng khác để tương tác với máy FTP Server.



Hình VII-4: Mô hình hoạt động của FTP Server.

II.2. Tập hợp các lệnh FTP

Lệnh	Chức năng
!	Chạy chương trình command dos trên máy tính cục bộ
?	Hiển thị giúp đỡ của các lệnh Ftp, lệnh này giống với lệnh Help.
Append	Chèn nội dung của một tập tin trên máy tính cục bộ vào cuối của một tập tin trên máy tính ở xa (máy FTP Server), dùng định dạng tập tin hiện tại.
Ascii	Đặt loại định dạng truyền file là ASCII, giá trị này là mặc định khi khởi tạo kết nối FTP.
Bell	Bật trạng thái chuông là on/off. Nếu là on thì sau mỗi lần lệnh truyền file hoàn thành thì máy phát ra tiếng chuông. Mặc định trạng thái này là off.
Binary	Đặt loại định dạng truyền file là binary.
Bye	Tắt kết nối với máy tính ở xa và thoát khỏi chương trình FTP.
Cd	Thay đổi thư mục hiện thành trên máy ở xa(Server).
Close	Ngừng phiên giao dịch với máy tính ở xa và trở về dòng lệnh

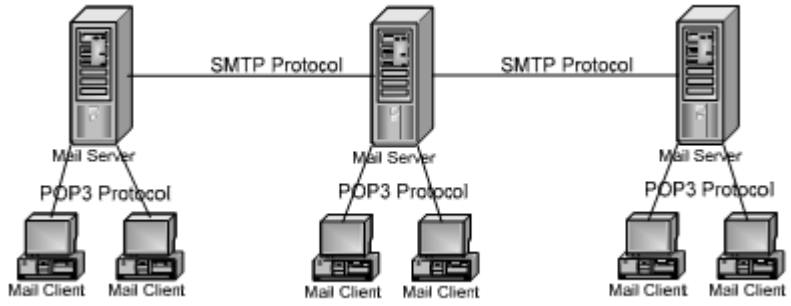
	của chương trình ftp.
Debug	Bật trạng thái Debugg on/off. Nếu là on thì mỗi lệnh gởi đến máy tính ở xa thì chương trình sẽ in ra các thông báo. Mặc định là trạng thái là off.
Delete	Xoá tập tin trên máy tính ở xa.
Dir	Hiển thị danh sách các tập tin và thư mục con trong thư mục hiện tại.
Disconnect	Tắt kết nối với máy tính ở xa và trở về dòng lệnh FTP.
Get	Chép một tập tin từ máy tính ở xa về máy tính cục bộ, dùng định dạng truyền file hiện tại.
Help	Hiển thị giúp đỡ của các lệnh Ftp.
Lcd	Thay đổi thư mục hiện trên máy tính cục bộ. Mặc định là thư mục đang làm việc trên máy tính cục bộ.
Ls	Hiển thị danh sách các tập tin và thư mục con trong thư mục hiện tại.
Mdelete	Xóa nhiều tập tin cùng trên một máy tính ở xa.
Mget	Chép nhiều tập tin từ máy tính ở xa về máy tính cục bộ dùng định dạng truyền file hiện tại.
mkdir	Tạo thư mục trên máy tính ở xa.
Mput	Chép nhiều tập tin ở máy tính cục bộ lên máy tính ở xa dùng định dạng truyền file hiện tại.
open	Mở một kết nối đến máy FTP Server.
Put	Chép một tập tin ở máy tính cục bộ lên máy tính ở xa dùng định dạng truyền file hiện tại.
Pwd	Hiển thị thư mục hiện hành trên máy tính ở xa.
Quit	Tắt kết nối với máy tính ở xa và thoát khỏi chương trình FTP.
Recv	Chép một tập tin từ máy tính ở xa về máy tính cục bộ, dùng định dạng truyền file hiện tại. Tương tự như lệnh
Rename	Đổi tên tập tin, thư mục trên máy tính ở xa.
Rmdir	Xóa một thư mục ở xa.
Send	Chép một tập tin ở máy tính cục bộ lên máy tính ở xa dùng định dạng truyền file hiện tại. Tương tự như Put.
Status	Hiển thị các trạng thái lựa chọn của kết nối FTP.

III. E-MAIL.

III.1. Mô hình hoạt động

E-mail (electronic mail) là thư điện tử, là một hình thức trao đổi thư từ nhưng thông qua mạng Internet. Dịch vụ này được sử dụng rất phổ biến và không đòi hỏi hai máy tính gởi và nhận thư phải kết nối online trên mạng..

Tại mỗi Mail Server thông thường gồm hai dịch vụ: POP3 (Post Office Protocol 3) làm nhiệm vụ giao tiếp mail giữa Mail Client và Mail Server, SMTP (Simple E-mail Transfer Protocol) làm nhiệm vụ giao tiếp mail giữa các máy Mail Server.



Hình VII-5 : Mô hình hoạt động của Mail Server.

Để sử dụng E-mail, người dùng cần có một account mail do nhà cung cấp dịch vụ Internet (ISP) cấp bao gồm các thông tin sau: địa chỉ mail (ví dụ: nvteo@hcm.vnn.vn), username, password và địa chỉ của Mail Server mà mình đăng ký. Sau đó chọn một chương trình Mail Client (Outlook Express, Eudora, Netscape...) và cấu hình các thông số trên vào chương trình đó. Từ đó bạn có thể sử dụng chương trình này để soạn thảo và gởi nhận mail một cách dễ dàng.

III.2. Các loại mail.

Thông thường có hai loại mail thông dụng là WebMail và POP Mail. Webmail là loại mail mà hình thức giao dịch mail giữa Client và Server dựa trên giao thức Web (http), thông thường Webmail là miễn phí. Còn POP Mail là loại mail mà các Mail Client tương tác với MAIL SERVER bằng giao thức POP3. Mail loại này tiện lợi và an toàn hơn nên thông thường là phải đăng ký thuê bao với nhà cung cấp dịch vụ.

III.3. Sử dụng WebMail.

Bạn muốn có một địa chỉ mail Internet để giao dịch với bạn bè trên thế giới, bạn có thể đến nhà cung cấp dịch vụ Internet để đăng ký hoặc tự tạo cho mình một địa chỉ mail miễn phí trên các Website nổi tiếng như Yahoo, Hotmail, Fpt, Vnn...

www.viet-ebook.co.cc

GIÁO TRÌNH

MẠNG MÁY TÍNH

Hà nội 11-2000

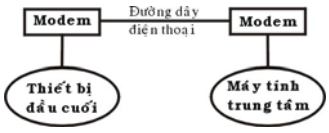
www.viet-ebook.co.cc

Chương 1

Sơ lược lịch sử phát triển của mạng máy tính

Vào giữa những năm 50 khi những thế hệ máy tính đầu tiên được đưa vào hoạt động thực tế với những bóng đèn điện tử thì chúng có kích thước rất cồng kềnh và tốn nhiều năng lượng. Hồi đó việc nhập dữ liệu vào các máy tính được thông qua các tấm bìa mà người viết chương trình đã đục lỗ sẵn. Mỗi tấm bìa tương đương với một dòng lệnh mà mỗi một cột của nó có chứa tất cả các ký tự cần thiết mà người viết chương trình phải đục lỗ vào ký tự mình lựa chọn. Các tấm bìa được đưa vào một "thiết bị" gọi là thiết bị đọc bìa mà qua đó các thông tin được đưa vào máy tính (hay còn gọi là trung tâm xử lý) và sau khi tính toán kết quả sẽ được đưa ra máy in. Như vậy các thiết bị đọc bìa và máy in được thể hiện như các thiết bị vào ra (I/O) đối với máy tính. Sau một thời gian các thế hệ máy mới được đưa vào hoạt động trong đó một máy tính trung tâm có thể được nối với nhiều thiết bị vào ra (I/O) mà qua đó nó có thể thực hiện liên tục hết chương trình này đến chương trình khác.

Cùng với sự phát triển của những ứng dụng trên máy tính các phương pháp nâng cao khả năng giao tiếp với máy tính trung tâm cũng đã được đầu tư nghiên cứu rất nhiều. Vào giữa những năm 60 một số nhà chế tạo máy tính đã nghiên cứu thành công những thiết bị truy cập từ xa tới máy tính của họ. Một trong những phương pháp thâm nhập từ xa được thực hiện bằng việc cài đặt một thiết bị đầu cuối ở một vị trí cách xa trung tâm tính toán, thiết bị đầu cuối này được liên kết với trung tâm bằng việc sử dụng đường dây điện thoại và với hai thiết bị xử lý tín hiệu (thường gọi là Modem) gắn ở hai đầu và tín hiệu được truyền thay vì trực tiếp thì thông qua dây điện thoại.



Hình 1.1. Mô hình truyền dữ liệu từ xa đầu tiên

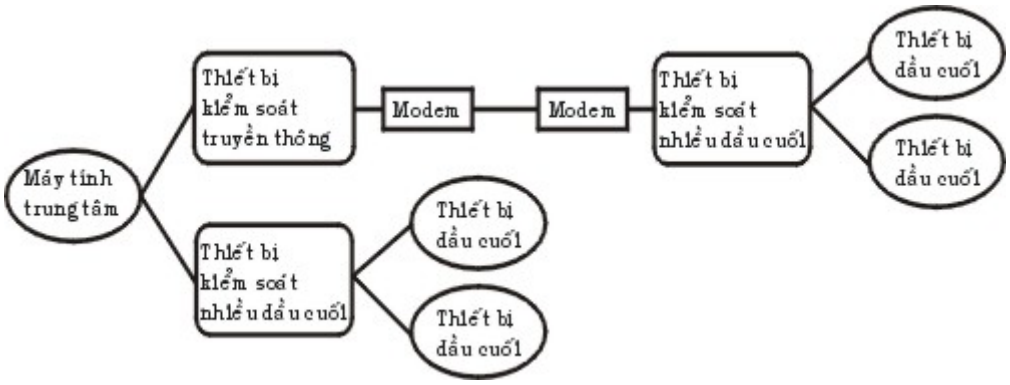
Những dạng đầu tiên của thiết bị đầu cuối bao gồm máy đọc bìa, máy in, thiết bị xử lý tín hiệu, các thiết bị cảm nhận. Việc liên kết từ xa đó có thể thực hiên thông qua những vùng khác nhau và đó là những dạng đầu tiên của hệ thống mạng.

Trong lúc đưa ra giới thiệu những thiết bị đầu cuối từ xa, các nhà khoa học đã triển khai một loạt những thiết bị điều khiển, những thiết bị đầu cuối đặc biệt cho phép người sử dụng nâng cao được khả năng tương tác với máy tính. Một trong những sản phẩm quan trọng đó là hệ thống thiết bị đầu cuối 3270 của IBM. Hệ thống đó bao gồm các màn hình, các hệ thống điều khiển, các thiết bị truyền thông được liên kết với các trung tâm tính toán. Hệ thống 3270 được giới thiệu vào năm 1971 và được sử dụng dùng để mở rộng khả năng tính toán của trung tâm máy tính tới các vùng xa. Để làm giảm nhiệm vụ truyền thông của máy tính trung tâm và số lượng các liên kết giữa máy tính trung tâm với các thiết bị đầu cuối, IBM và các công ty máy tính khác đã sản xuất một số các thiết bị sau:

- **Thiết bị kiểm soát truyền thông**: có nhiệm vụ nhận các bit tín hiệu từ các kênh truyền thông, gom chúng lại thành các byte dữ liệu và chuyển nhóm các byte đó tới máy tính trung tâm để xử lý, thiết bị này cũng thực hiện công việc ngược lại để chuyển tín hiệu trả lời của máy tính trung tâm tới các trạm ở xa. Thiết bị trên cho

phép giảm bớt được thời gian xử lý trên máy tính trung tâm và xây dựng các thiết bị logic đặc trưng.

- **Thiết bị kiểm soát nhiều đầu cuối**: cho phép cùng một lúc kiểm soát nhiều thiết bị đầu cuối. Máy tính trung tâm chỉ cần liên kết với một thiết bị như vậy là có thể phục vụ cho tất cả các thiết bị đầu cuối đang được gắn với thiết bị kiểm soát trên. Điều này đặc biệt có ý nghĩa khi thiết bị kiểm soát nằm ở cách xa máy tính vì chỉ cần sử dụng một đường điện thoại là có thể phục vụ cho nhiều thiết bị đầu cuối.



Hình 1.2: Mô hình trao đổi mạng của hệ thống 3270

Vào giữa những năm 1970, các thiết bị đầu cuối sử dụng những phương pháp liên kết qua đường cáp nằm trong một khu vực đã được ra đời. Với những ưu điểm từ nâng cao tốc độ truyền dữ liệu và qua đó kết hợp được khả năng tính toán của các máy tính lại với nhau. Để thực hiện việc nâng cao khả năng tính toán với nhiều máy tính các nhà sản xuất bắt đầu xây dựng các mạng phức tạp. Vào những năm 1980 các hệ thống đường truyền tốc độ cao đã được thiết lập ở Bắc Mỹ và Châu Âu và từ đó cũng xuất hiện các nhà cung cấp các dịnh vụ truyền thông với những đường truyền có tốc độ cao hơn nhiều lần so với đường dây điện thoại. Với những chi phí thuê bao chấp nhận được, người ta có thể sử dụng được các đường truyền này để liên kết máy tính lại với nhau và bắt đầu hình thành các mạng một cách rộng khắp. Ở đây các nhà cung cấp dịch vụ đã xây dựng những đường truyền dữ liệu liên kết giữa các thành phố và khu vực với nhau và sau đó cung cấp các dịch vụ truyền dữ liệu cho những người xây dựng mạng. ẩ gười xây dựng mạng lúc này sẽ không cần xây dựng lại đường truyền của mình mà chỉ cần sử dụng một phần các năng lực truyền thông của các nhà cung cấp.

Vào năm 1974 công ty IBM đã giới thiệu một loạt các thiết bị đầu cuối được chế tạo cho lĩnh vực ngân hàng và thương mại, thông qua các dây cáp mạng các thiết bị đầu cuối có thể truy cập cùng một lúc vào một máy tính dùng chung. Với việc liên kết các máy tính nằm ở trong một khu vực nhỏ như một tòa nhà hay là một khu nhà thì tiền chi phí cho các thiết bị và phần mềm là thấp. Từ đó việc nghiên cứu khả năng sử dụng chung môi trường truyền thông và các tài nguyên của các máy tính nhanh chóng được đầu tư.

Vào năm 1977, công ty Datapoint Corporation đã bắt đầu bán hệ điều hành mạng của mình là "Attached Resource Computer ẩ etwork" (hay gọi tắt là Arcnet) ra thị trường. Mạng Arcnet cho phép liên kết các máy tính và các trạm đầu cuối lại bằng dây cáp mạng, qua đó đã trở thành là hệ điều hành mạng cục bộ đầu tiên.

Từ đó đến nay đã có rất nhiều công ty đưa ra các sản phẩm của mình, đặc biệt khi các máy tính cá nhân được sử dụng một cánh rộng rãi. Khi số lượng máy vi tính trong một văn phòng hay cơ quan được tăng lên nhanh chóng thì việc kết nối chúng trở nên vô cùng cần thiết và sẽ mang lại nhiều hiệu quả cho người sử dụng.

ả gày nay với một lượng lớn về thông tin, nhu cầu xử lý thông tin ngày càng cao. Mạng máy tính hiện nay trở nên quá quen thuộc đối với chúng ta, trong mọi lĩnh vực như khoa học, quân sự, quốc phòng, thương mại, dịch vụ, giáo dục... Hiện nay ở nhiều nơi mạng đã trở thành một nhu cầu không thể thiếu được. ả gười ta thấy được việc kết nối các máy tính thành mạng cho chúng ta những khả năng mới to lớn như:

- **Sử dụng chung tài nguyên:** ả những tài nguyên của mạng (như thiết bị, chương trình, dữ liệu) khi được trở thành các tài nguyên chung thì mọi thành viên của mạng đều có thể tiếp cận được mà không quan tâm tới những tài nguyên đó ở đâu.

- **Tăng độ tin cậy của hệ thống**: ả gười ta có thể dễ dàng bảo trì máy móc và lưu trữ (backup) các dữ liệu chung và khi có trục trặc trong hệ thống thì chúng có thể được khôi phục nhanh chóng. Trong trường hợp có trục trặc trên một trạm làm việc thì người ta cũng có thể sử dụng những trạm khác thay thế.

- **Nâng cao chất lượng và hiệu quả khai thác thông tin:** Khi thông tin có thể được sữ dụng chung thì nó mang lại cho người sử dụng khả năng tổ chức lại các công việc với những thay đổi về chất như:

    - Đáp ứng những nhu cầu của hệ thống ứng dụng kinh doanh hiện đại.
    - Cung cấp sự thống nhất giữa các dữ liệu.
    - Tăng cường năng lực xử lý nhờ kết hợp các bộ phận phân tán.
    - Tăng cường truy nhập tới các dịch vụ mạng khác nhau đang được cung cấp trên thế giới.

Với nhu cầu đòi hỏi ngày càng cao của xã hội nên vấn đề kỹ thuật trong mạng là mối quan tâm hàng đầu của các nhà tin học. Ví dụ như làm thế nào để truy xuất thông tin một cách nhanh chóng và tối ưu nhất, trong khi việc xử lý thông tin trên mạng quá nhiều đôi khi có thể làm tắc nghẽn trên mạng và gây ra mất thông tin một cách đáng tiếc.

Hiện nay việc làm sao có được một hệ thống mạng chạy thật tốt, thật an toàn với lợi ích kinh tế cao đang rất được quan tâm. Một vấn đề đặt ra có rất nhiều giải pháp về công nghệ, một giải pháp có rất nhiều yếu tố cấu thành, trong mỗi yếu tố có nhiều cách lựa chọn. ả hư vậy để đưa ra một giải pháp hoàn chỉnh, phù hợp thì phải trải qua một quá trình chọn lọc dựa trên những ưu điểm của từng yếu tố, từng chi tiết rất nhỏ.

Để giải quyết một vấn đề phải dựa trên những yêu cầu đặt ra và dựa trên công nghệ để giải quyết. ả hưng công nghệ cao nhất chưa chắc là công nghệ tốt nhất, mà công nghệ tốt nhất là công nghệ phù hợp nhất.

www.viet-ebook.co.cc

Chương 2

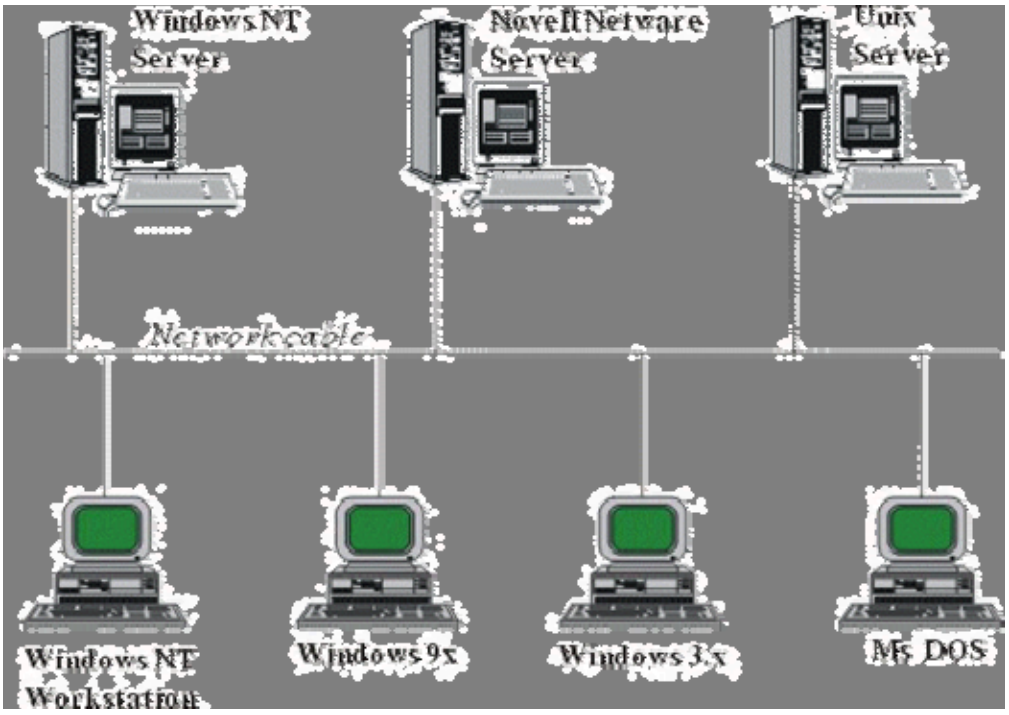
Những khái niệm cơ bản của mạng máy tính

Với sự phát triển của khoa học và kỹ thuật, hiện nay các mạng máy tính đã phát triển một cách nhanh chóng và đa dạng cả về quy mô, hệ điều hành và ứng dụng. Do vậy việc nghiên cứu chúng ngày càng trở nên phức tạp. Tuy nhiên các mạng máy tính cũng có cùng các điểm chung thông qua đó chúng ta có thể đánh giá và phân loại chúng.

I. Định nghĩa mạng máy tính

> *Mạng máy tính là một tập hợp các máy tính được nối với nhau bởi đường truyền theo một cấu trúc nào đó và thông qua đó các máy tính trao đổi thông tin qua lại cho nhau.*

Đường truyền là hệ thống các thiết bị truyền dẫn có dây hay không dây dùng để chuyển các tín hiệu điện tử từ máy tính này đến máy tính khác. Các tín hiệu điện tử đó biểu thị các giá trị dữ liệu dưới dạng các xung nhị phân (on - off). Tất cả các tín hiệu được truyền giữa các máy tính đều thuộc một dạng sóng điện từ. Tùy theo tần số của sóng điện từ có thể dùng các đường truyền vật lý khác nhau để truyền các tín hiệu. Ở đây đường truyền được kết nối có thể là dây cáp đồng trục, cáp xoắn, cáp quang, dây điện thoại, sóng vô tuyến ... Các đường truyền dữ liệu tạo nên cấu trúc của mạng. Hai khái niệm đường truyền và cấu trúc là những đặc trưng cơ bản của mạng máy tính.



Hình 2.1: Một mô hình liên kết các máy tính trong mạng

Với sự trao đổi qua lại giữa máy tính này với máy tính khác đã phân biệt mạng máy tính với các hệ thống thu phát một chiều như truyền hình, phát thông tin từ vệ tinh xuống các trạm thu thụ động... vì tại đây chỉ có thông tin một chiều từ nơi phát đến nơi thu mà không quan tâm đến có bao nhiêu nơi thu, có thu tốt hay không.

Đặc trưng cơ bản của đường truyền vật lý là giải thông. Giải thông của một đường chuyền chính là độ đo phạm vi tần số mà nó có thể đáp ứng được. Tốc độ truyền dữ liệu trên đường truyền còn được gọi là thông lượng của đường truyền - thường được tính bằng số lượng bit được truyền đi trong một giây (Bps). Thông lượng còn được đo bằng đơn vị khác là Baud (lấy từ tên nhà bác học - Emile Baudot). Baud biểu thị số lượng thay đổi tín hiệu trong một giây.

Ở đây Baud và Bps không phải bao giờ cũng đồng nhất. Ví dụ: nếu trên đường dây có 8 mức tín hiệu khác nhau thì mỗi mức tín hiệu tương ứng với 3 bit hay là 1 Baud tương ứng với 3 bit. Chỉ khi có 2 mức tín hiệu trong đó mỗi mức tín hiệu tương ứng với 1 bit thì 1 Baud mới tương ứng với 1 bit.

II. Phân loại mạng máy tính

Do hiện nay mạng máy tính được phát triển khắp nơi với những ứng dụng ngày càng đa dạng cho nên việc phân loại mạng máy tính là một việc rất phức tạp. ả gười ta có thể chia các mạng máy tính theo khoảng cách địa lý ra làm hai loại: Mạng diện rộng và Mạng cục bộ.

- **Mạng cục bộ (Local Area Networks - LAN)** là mạng được thiết lập để liên kết các máy tính trong một khu vực như trong một toà nhà, một khu nhà.

- **Mạng diện rộng (Wide Area Networks - WAN)** là mạng được thiết lập để liên kết các máy tính của hai hay nhiều khu vực khác nhau như giữa các thành phố hay các tỉnh.

Sự phân biệt trên chỉ có tính chất ước lệ, các phân biệt trên càng trở nên khó xác định với việc phát triển của khoa học và kỹ thuật cũng như các phương tiện truyền dẫn. Tuy nhiên với sự phân biệt trên phương diện địa lý đã đưa tới việc phân biệt trong nhiều đặc tính khác nhau của hai loại mạng trên, việc nghiên cứu các phân biệt đó cho ta hiểu rõ hơn về các loại mạng.

III. Sự phân biệt giữa mạng cục bộ và mạng diện rộng

Mạng cục bộ và mạng diện rộng có thể được phân biệt bởi: địa phương hoạt động, tốc độ đường truyền và tỷ lệ lỗi trên đường truyền, chủ quản của mạng, đường đi của thông tin trên mạng, dạng chuyển giao thông tin.

***Địa phương hoạt động*:** Liên quan đến khu vực địa lý thì mạng cục bộ sẽ là mạng liên kết các máy tính nằm ở trong một khu vực nhỏ. Khu vực có thể bao gồm một tòa nhà hay là một khu nhà... Điều đó hạn chế bởi khoảng cách đường dây cáp được dùng để liên kết các máy tính của mạng cục bộ (Hạn chế đó còn là hạn chế của khả năng kỹ thuật của đường truyền dữ liệu). ả gược lại mạng diện rộng là mạng có khả năng liên kết các máy tính trong một vùng rộng lớn như là một thành phố, một miền, một đất nước, mạng diện rộng được xây dựng để nối hai hoặc nhiều khu vực địa lý riêng biệt.

***Tốc độ đường truyền và tỷ lệ lỗi trên đường truyền*:** Do các đường cáp của mạng cục bộ đươc xây dựng trong một khu vực nhỏ cho nên nó ít bị ảnh hưởng bởi tác động của thiên nhiên (như là sấm chớp, ánh sáng...). Điều đó cho phép mạng cục bộ có thể truyền dữ liệu với tốc độ cao mà chỉ chịu một tỷ lệ lỗi nhỏ. ả gược lại với mạng diện rộng do phải

truyền ở những khoảng cách khá xa với những đường truyền dẫn dài có khi lên tới hàng ngàn km. Do vậy mạng diện rộng không thể truyền với tốc độ quá cao vì khi đó tỉ lệ lỗi sẽ trở nên khó chấp nhận được.

Mạng cục bộ thường có tốc độ truyền dữ liệu từ 4 đến 16 Mbps và đạt tới 100 Mbps nếu dùng cáp quang. Còn phần lớn các mạng diện rộng cung cấp đường truyền có tốc độ thấp hơn nhiều như T1 với 1.544 Mbps hay E1 với 2.048 Mbps.

(Ở đây bps (Bit Per Second) là một đơn vị trong truyền thông tương đương với 1 bit được truyền trong một giây, ví dụ như tốc độ đường truyền là 1 Mbps tức là có thể truyền tối đa 1 Megabit trong 1 giây trên đường truyền đó).

Thông thường trong mạng cục bộ tỷ lệ lỗi trong truyền dữ liệu vào khoảng $1/10^7$-10^8 còn trong mạng diện rộng thì tỷ lệ đó vào khoảng $1/10^6$ - 10^7

Chủ quản và điều hành của mạng: Do sự phức tạp trong việc xây dựng, quản lý, duy trì các đường truyền dẫn nên khi xây dựng mạng diện rộng người ta thường sử dụng các đường truyền được thuê từ các công ty viễn thông hay các nhà cung cấp dịch vụ truyền số liệu. Tùy theo cấu trúc của mạng những đường truyền đó thuộc cơ quan quản lý khác nhau như các nhà cung cấp đường truyền nội hạt, liên tỉnh, liên quốc gia. Các đường truyền đó phải tuân thủ các quy định của chính phủ các khu vực có đường dây đi qua như: tốc độ, việc mã hóa.

Còn đối với mạng cục bộ thì công việc đơn giản hơn nhiều, khi một cơ quan cài đặt mạng cục bộ thì toàn bộ mạng sẽ thuộc quyền quản lý của cơ quan đó.

Đường đi của thông tin trên mạng: Trong mạng cục bộ thông tin được đi theo con đường xác định bởi cấu trúc của mạng. Khi người ta xác định cấu trúc của mạng thì thông tin sẽ luôn luôn đi theo cấu trúc đã xác định đó. Còn với mạng diện rộng dữ liệu cấu trúc có thể phức tạp hơn nhiều do việc sử dụng các dịch vụ truyền dữ liệu. Trong quá trình hoạt động các điểm nút có thể thay đổi đường đi của các thông tin khi phát hiện ra có trục trặc trên đường truyền hay khi phát hiện có quá nhiều thông tin cần truyền giữa hai điểm nút nào đó. Trên mạng diện rộng thông tin có thể có các con đường đi khác nhau, điều đó cho phép có thể sử dụng tối đa các năng lực của đường truyền hay nâng cao điều kiện an toàn trong truyền dữ liệu.

Dạng chuyển giao thông tin: Phần lớn các mạng diện rộng hiện nay được phát triển cho việc truyền đồng thời trên đường truyền nhiều dạng thông tin khác nhau như: video, tiếng nói, dữ liệu... Trong khi đó các mạng cục bộ chủ yếu phát triển trong việc truyền dữ liệu thông thường. Điều này có thể giải thích do việc truyền các dạng thông tin như video, tiếng nói trong một khu vực nhỏ ít được quan tâm hơn như khi truyền qua những khoảng cách lớn.

Các hệ thống mạng hiện nay ngày càng phức tạp về chất lượng, đa dạng về chủng loại và phát triển rất nhanh về chất. Trong sự phát triển đó số lượng những nhà sản xuất từ phần mềm, phần cứng máy tính, các sản phẩm viễn thông cũng tăng nhanh với nhiều sản phẩm đa dạng. Chính vì vậy vai trò chuẩn hóa cũng mang những ý nghĩa quan trọng. Tại các nước các cơ quan chuẩn quốc gia đã đưa ra các những chuẩn về phần cứng và các quy định

về giao tiếp nhằm giúp cho các nhà sản xuất có thể làm ra các sản phẩm có thể kết nối với các sản phẩm do hãng khác sản xuất.

www.viet-ebook.co.cc

Chương 3

Mô hình truyền thông

I. Sự cần thiết phải có mô hình truyền thông

Để một mạng máy tính trở một môi trường truyền dữ liệu thì nó cần phải có những yếu tố sau:

- Mỗi máy tính cần phải có một địa chỉ phân biệt trên mạng.
- Việc chuyển dữ liệu từ máy tính này đến máy tính khác do mạng thực hiện thông qua những quy định thống nhất gọi là giao thức của mạng.

Khi các máy tính trao đổi dữ liệu với nhau thì một quá trình truyền giao dữ liệu đã được thực hiện hoàn chỉnh. Ví dụ như để thực hiện việc truyền một file giữa một máy tính với một máy tính khác cùng được gắn trên một mạng các công việc sau đây phải được thực hiện:

- Máy tính cần truyền cần biết địa chỉ của máy nhận.
- Máy tính cần truyền phải xác định được máy tính nhận đã saün sàng nhận thông tin
- Chương trình gửi file trên máy truyền cần xác định được rằng chương trình nhận file trên máy nhận đã saün sàng tiếp nhận file.
- ả ếu cấu trúc file trên hai máy không giống nhau thì một máy phải làm nhiệm vụ chuyển đổi file từ dạng này sang dạng kia.
- Khi truyền file máy tính truyền cần thông báo cho mạng biết địa chỉ của máy nhận để các thông tin được mạng đưa tới đích.

Điều trên đó cho thấy giữa hai máy tính đã có một sự phối hợp hoạt động ở mức độ cao. Bây giờ thay vì chúng ta xét cả quá trình trên như là một quá trình chung thì chúng ta sẽ chia quá trình trên ra thành một số công đoạn và mỗi công đoạn con hoạt động một cách độc lập với nhau. Ở đây chương trình truyền nhận file của mỗi máy tính được chia thành ba module là: Module truyền và nhận File, Module truyền thông và Module tiếp cận mạng. Hai module tương ứng sẽ thực hiện việc trao đổi với nhau trong đó:

- *Module truyền và nhận file* cần được thực hiện tất cả các nhiệm vụ trong các ứng dụng truyền nhận file. Ví dụ: truyền nhận thông số về file, truyền nhận các mẫu tin của file, thực hiện chuyển đổi file sang các dạng khác nhau nếu cần. Module truyền và nhận file không cần thiết phải trực tiếp quan tâm tới việc truyền dữ liệu trên mạng như thế nào mà nhiệm vụ đó được giao cho Module truyền thông.
- *Module truyền thông* quan tâm tới việc các máy tính đang hoạt động và saün sàng trao đổi thông tin với nhau. ả ó còn kiểm soát các dữ liệu sao cho những dữ liệu

này có thể trao đổi một cách chính xác và an toàn giữa hai máy tính. Điều đó có nghĩa là phải truyền file trên nguyên tắc đảm bảo an toàn cho dữ liệu, tuy nhiên ở đây có thể có một vài mức độ an toàn khác nhau được dành cho từng ứng dụng. Ở đây việc trao đổi dữ liệu giữa hai máy tính không phụ thuộc vào bản chất của mạng đang liên kết chúng. ả những yêu cầu liên quan đến mạng đã được thực hiện ở module thứ ba là module tiếp cận mạng và nếu mạng thay đổi thì chỉ có module tiếp cận mạng bị ảnh hưởng.

- *Module tiếp cận mạng* được xây dựng liên quan đến các quy cách giao tiếp với mạng và phụ thuộc vào bản chất của mạng. ả ó đảm bảo việc truyền dữ liệu từ máy tính này đến máy tính khác trong mạng.

ả hư vậy thay vì xét cả quá trình truyền file với nhiều yêu cầu khác nhau như một tiến trình phức tạp thì chúng ta có thể xét quá trình đó với nhiều tiến trình con phân biệt dựa trên việc trao đổi giữa các Module tương ứng trong chương trình truyền file. Cách này cho phép chúng ta phân tích kỹ quá trình file và dễ dàng trong việc viết chương trình.

Việc xét các module một cách độc lập với nhau như vậy cho phép giảm độ phức tạp cho việc thiết kế và cài đặt. Phương pháp này được sử dụng rộng rãi trong việc xây dựng mạng và các chương trình truyền thông và được gọi là phương pháp phân tầng (layer).

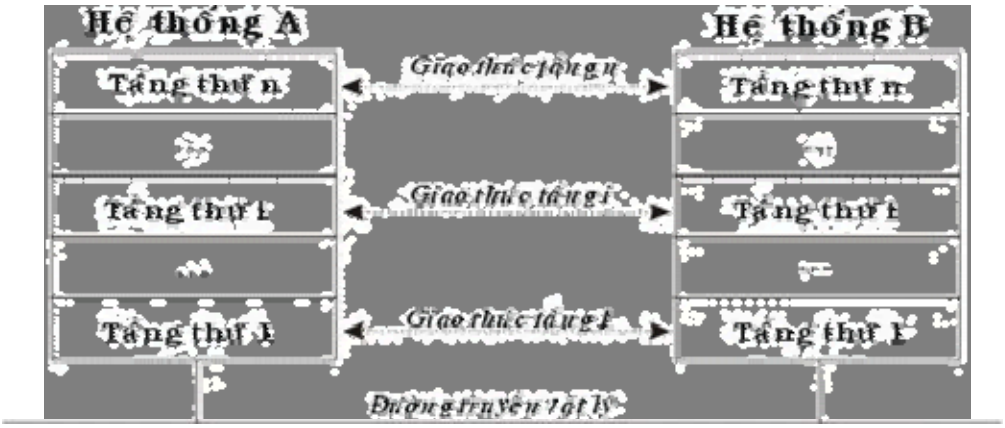
ả guyên tắc của phương pháp phân tầng là:

- Mỗi hệ thống thành phần trong mạng được xây dựng như một cấu trúc nhiều tầng và đều có cấu trúc giống nhau như: số lượng tầng và chức năng của mỗi tầng.

- Các tầng nằm chồng lên nhau, dữ liệu được chỉ trao đổi trực tiếp giữa hai tầng kề nhau từ tầng trên xuống tầng dưới và ngược lại.

- Cùng với việc xác định chức năng của mỗi tầng chúng ta phải xác định mối quan hệ giữa hai tầng kề nhau. Dữ liệu được truyền đi từ tầng cao nhất của hệ thống truyền lần lượt đến tầng thấp nhất sau đó truyền qua đường nối vật lý dưới dạng các bit tới tầng thấp nhất của hệ thống nhận, sau đó dữ liệu được truyền ngược lên lần lượt đến tầng cao nhất của hệ thống nhận.

- Chỉ có hai tầng thấp nhất có liên kết vật lý với nhau còn các tầng trên cùng thứ tư chỉ có các liên kết logic với nhau. Liên kết logic của một tầng được thực hiện thông qua các tầng dưới và phải tuân theo những quy định chặt chẽ, các quy định đó được gọi giao thức của tầng.



Hình 3.1: Mô hình phân tầng gồm N tầng

II. Mô hình truyền thông đơn giản 3 tầng

ả ói chung trong truyền thông có sự tham gia của các thành phần: các chương trình ứng dụng, các chương trình truyền thông, các máy tính và các mạng. Các chương trình ứng dụng là các chương trình của người sử dụng được thực hiện trên máy tính và có thể tham gia vào quá trình trao đổi thông tin giữa hai máy tính. Trên một máy tính với hệ điều hành đa nhiệm (như Windows, Uả IX) thường được thực hiện đồng thời nhiều ứng dụng trong đó có những ứng dụng liên quan đến mạng và các ứng dụng khác. Các máy tính được nối với mạng và các dữ liệu được trao đổi thông qua mạng từ máy tính này đến máy tính khác.

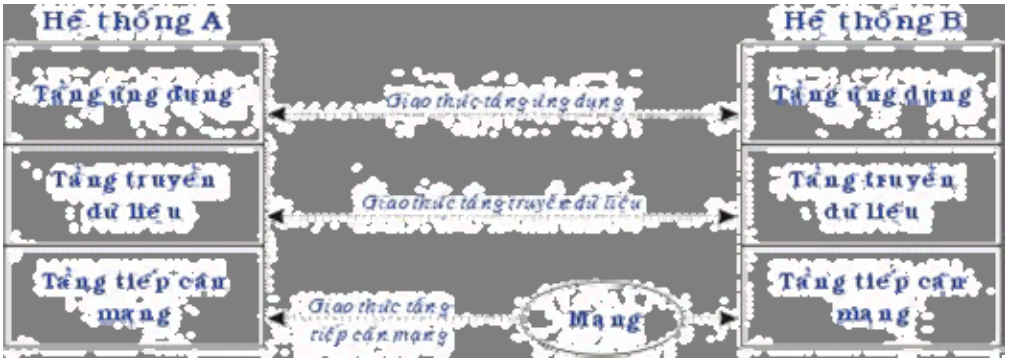
Việc gửi dữ liệu được thực hiện giữa một ứng dụng với một ứng dụng khác trên hai máy tính khác nhau thông qua mạng được thực hiện như sau: Ứng dụng gửi chuyển dữ liệu cho chương trình truyền thông trên máy tính của nó, chương trình truyền thông sẽ gửi chúng tới máy tính nhận. Chương trình truyền thông trên máy nhận sẽ tiếp nhận dữ liệu, kiểm tra nó trước khi chuyển giao cho ứng dụng đang chờ dữ liệu.

Với mô hình truyền thông đơn giản người ta chia chương trình truyền thông thành ba tầng không phụ thuộc vào nhau là: tầng ứng dụng, tầng chuyển vận và tầng tiếp cận mạng.

- ***Tầng tiếp cận mạng*** liên quan tới việc trao đổi dữ liệu giữa máy tính và mạng mà nó được nối vào. Để dữ liệu đến được đích máy tính gửi cần phải chuyển địa chỉ của máy tính nhận cho mạng và qua đó mạng sẽ chuyển các thông tin tới đích. ả goài ra máy gửi có thể sử dụng một số phục vụ khác nhau mà mạng cung cấp như gửi ưu tiên, tốc độ cao. Trong tầng này có thể có nhiều phần mềm khác nhau được sử dụng phụ thuộc vào các loại của mạng ví dụ như mạng chuyển mạch, mạng chuyển mạch gói, mạng cục bộ.

- ***Tầng truyền dữ liệu*** thực hiện quá trình truyền thông không liên quan tới mạng và nằm ở trên tầng tiếp cận mạng. Tầng truyền dữ liệu không quan tâm tới bản chất các ứng dụng đang trao đổi dữ liệu mà quan tâm tới làm sao cho các dữ liệu được trao đổi một cách an toàn. Tầng truyền dữ liệu đảm bảo các dữ liệu đến được đích và đến theo đúng thứ tự mà chúng được xử lý. Trong tầng truyền dữ liệu người ta phải có những cơ chế nhằm đảm bảo sự chính xác đó và rõ ràng các cơ chế này không phụ thuộc vào bản chất của từng ứng dụng và chúng sẽ phục vụ cho tất cả các ứng dụng.

- ***Tầng ứng dụng*** sẽ chứa các module phục vụ cho tất cả những ứng dụng của người sử dụng. Với các loại ứng dụng khác nhau (như là truyền file, truyền thư mục) cần các module khác nhau.



Hình 3.2 Mô hình truyền thông 3 tầng

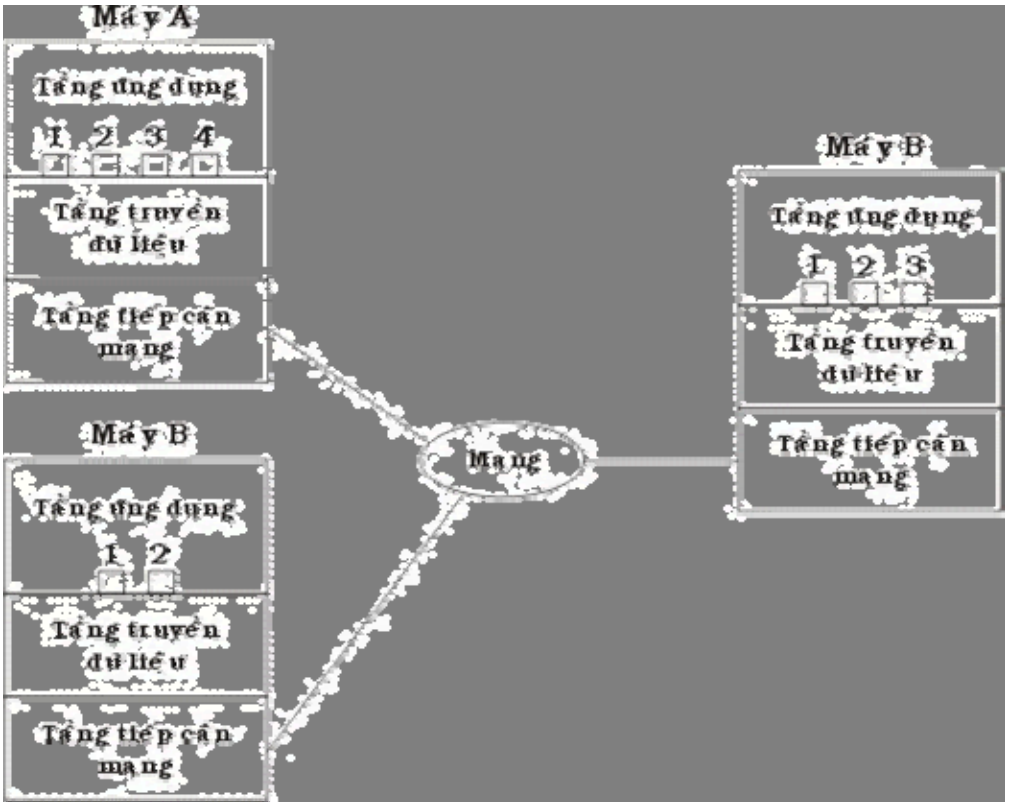
Trong một mạng với nhiều máy tính, mỗi máy tính một hay nhiều ứng dụng thực hiện đồng thời (Tại đây ta xét trên một máy tính trong một thời điểm có thể chạy nhiều ứng dụng và các ứng dụng đó có thể thực hiện đồng thời việc truyền dữ liệu qua mạng). Một ứng dụng khi cần truyền dữ liệu qua mạng cho một ứng dụng khác cần phải gọi 1 module tầng ứng dụng của chương trình truyền thông trên máy của mình, đồng thời ứng dụng kia cũng sẽ gọi 1 module tầng ứng dụng trên máy của nó. Hai module ứng dụng sẽ liên kết với nhau nhằm thực hiện các yêu cầu của các chương trình ứng dụng.

Các ứng dụng đó sẽ trao đổi với nhau thông qua mạng, tuy nhiên trong 1 thời điểm trên một máy có thể có nhiều ứng dụng cùng hoạt động và để việc truyền thông được chính xác thì các ứng dụng trên một máy cần phải có một địa chỉ riêng biệt. Rõ ràng cần có hai lớp địa chỉ:

- Mỗi máy tính trên mạng cần có một địa chỉ mạng của mình, hai máy tính trong cùng một mạng không thể có cùng địa chỉ, điều đó cho phép mạng có thể truyền thông tin đến từng máy tính một cách chính xác.

- Mỗi một ứng dụng trên một máy tính cần phải có địa chỉ phân biệt trong máy tính đo. ả ó cho phép tầng truyền dữ liệu giao dữ liệu cho đúng ứng dụng đang cần. Địa chỉ đó được gọi là điểm tiếp cận giao dịch. Điều đó cho thấy mỗi một ứng dụng sẽ tiếp cận các phục vụ của tầng truyền dữ liệu một cách độc lập.

- Các module cùng một tầng trên hai máy tính khác nhau sẽ trao đổi với nhau một cách chặt chẽ theo các qui tắc xác định trước được gọi là giao thức. Một giao thức được thể hiện một cách chi tiết bởi các chức năng cần phải thực hiện như các giá trị kiểm tra lỗi, việc định dạng các dữ liệu, các quy trình cần phải thực hiện để trao đổi thông tin.



Hình 3.3 Ví dụ mô hình truyền thông đơn giản

Chúng ta hãy xét trong ví dụ (như hình vẽ trên): giả sử có ứng dụng có điểm tiếp cận giao dịch 1 trên máy tính A muốn gửi thông tin cho một ứng dụng khác trên máy tính B có điểm tiếp cận giao dịch 2. Ứng dụng trên máy tính A chuyển các thông tin xuống tầng truyền dữ liệu của A với yêu cầu gửi chúng cho điểm tiếp cận giao dịch 2 trên máy tính B. Tầng truyền dữ liệu máy A sẽ chuyển các thông tin xuống tầng tiếp cận mạng máy A với yêu cầu chuyển chúng cho máy tính B (Chú ý rằng mạng không cần biết địa chỉ của điểm tiếp cận giao dịch mà chỉ cần biết địa chỉ của máy tính B). Để thực hiện quá trình này, các thông tin kiểm soát cũng sẽ được truyền cùng với dữ liệu.

Đầu tiên khi ứng dụng 1 trên máy A cần gửi một khối dữ liệu nó chuyển khối đó cho tầng vận chuyển. Tầng vận chuyển có thể chia khối đó ra thành nhiều khối nhỏ phụ thuộc vào yêu cầu của giao thức của tầng và đóng gói chúng thành các gói tin (packet). Mỗi một gói tin sẽ được bổ sung thêm các thông tin kiểm soát của giao thức và được gọi là phần đầu (Header) của gói tin. Thông thường phần đầu của gói tin cần có:

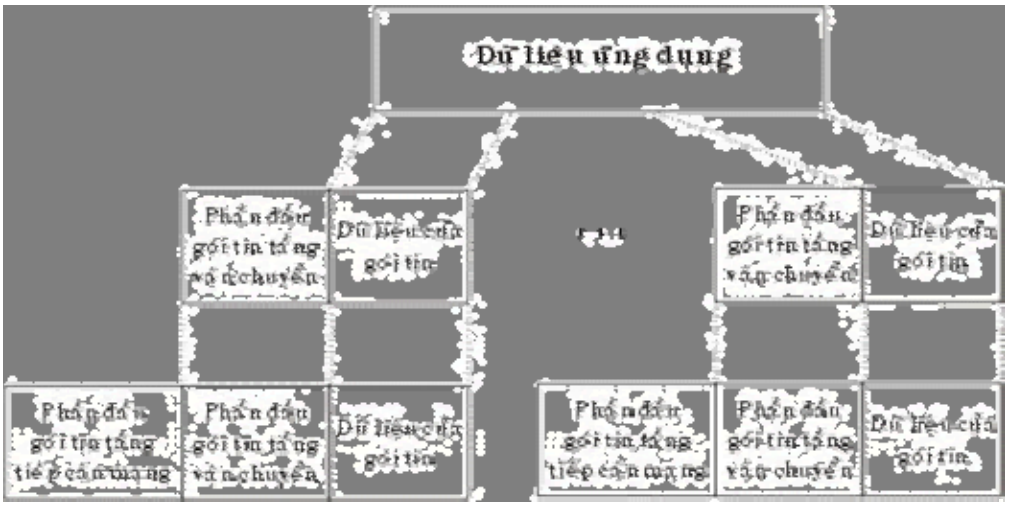
- ***Địa chỉ của điểm tiếp cận giao dịch nơi đến* (Ở đây là 3):** khi tầng vận chuyển của máy B nhận được gói tin thì nó biết được ứng dụng nào mà nó cần giao.

- ***Số thứ tự*** của gói tin, khi tầng vận chuyển chia một khối dữ liệu ra thành nhiều gói tin thì nó cần phải đánh số thứ tự các gói tin đó. ả ếu chúng đi đến đích nếu sai thứ tự thì tầng vận chuyển của máy nhận có thể phát hiện và chỉnh lại thứ tự. ả goài ra nếu có lỗi trên đường truyền thì tầng vận chuyển của máy nhận sẽ phát hiện ra và yêu cầu gửi lại một cách chính xác.

- ***Mã sửa lỗi*:** để đảm bảo các dữ liệu được nhận một cách chính xác thì trên cơ sở các dữ liệu của gói tin tầng vận chuyển sẽ tính ra một giá trị theo một công thức có

sãn và gửi nó đi trong phần đầu của gói tin. Tầng vận chuyển nơi nhận thông qua giá trị đó xác định được gói tin đó có bị lỗi trên đường truyền hay không.

Bước tiếp theo tầng vận chuyển máy A sẽ chuyển từng gói tin và địa chỉ của máy tính đích (ở đây là B) xuống tầng tiếp cận mạng với yêu cầu chuyển chúng đi. Để thực hiện được yêu cầu này tầng tiếp cận mạng cũng tạo các gói tin của mình trước khi truyền qua mạng. Tại đây giao thức của tầng tiếp cận mạng sẽ thêm các thông tin điều khiển vào phần đầu của gói tin mạng.



Hình 3.4: Mô hình thiết lập gói tin

Trong phần đầu gói tin mạng sẽ bao gồm địa chỉ của máy tính nhận, dựa trên địa chỉ này mạng truyền gói tin tới đích. ầ goài ra có thể có những thông số như là mức độ ưu tiên.

ầ hư vậy thông qua mô hình truyền thông đơn giản chúng ta cũng có thể thấy được phương thức hoạt động của các máy tính trên mạng, có thể xây dựng và thay đổi các giao thức trong cùng một tầng.

III. Các nhu cầu về chuẩn hóa đối với mạng

Trong phần trên chúng ta đã xem xét một mô hình truyền thông đơn giản, trong thực tế việc phân chia các tầng như trong mô hình trên thực sự chưa đủ. Trên thế giới hiện có một số cơ quan định chuẩn, họ đưa ra hàng loạt chuẩn về mạng tuy các chuẩn đó có tính chất khuyến nghị chứ không bắt buộc nhưng chúng rất được các cơ quan chuẩn quốc gia coi trọng.

Hai trong số các cơ quan chuẩn quốc tế là:

- **ISO (The International Standards Organization) -** Là tổ chức tiêu chuẩn quốc tế hoạt động dưới sự bảo trợ của Liên hợp Quốc với thành viên là các cơ quan chuẩn quốc gia với số lượng khoảng hơn 100 thành viên với mục đích hỗ trợ sự phát triển các chuẩn trên phạm vi toàn thế giới. Một trong những thành tựu của ISO trong lãnh vực truyền thông là mô hình hệ thống mở (Open Systems Interconnection - gọi tắt là OSI).

- **CCITT (Commité Consultatif International pour le Telegraphe et la Téléphone) -** Tổ chức tư vấn quốc tế về điện tín và điện thoại làm việc dưới sự bảo trợ của Liên Hiệp Quốc có trụ sở chính tại Geneva - Thụy sỹ. Các thành viên chủ yếu là các cơ quan bưu chính viễn thông các quốc gia. Tổ chức này có vai trò phát triển các khuyến nghị trong các lãnh vực viễn thông.

IV. Một số mô hình chuẩn hóa

1. Mô hình OSI (Open Systems Interconnection)

Mô hình OSI là một cơ sở dành cho việc chuẩn hoá các hệ thống truyền thông, nó được nghiên cứu và xây dựng bởi ISO. Việc nghiên cứu về mô hình OSI được bắt đầu tại ISO vào năm 1971 với mục tiêu nhằm tới việc nối kết các sản phẩm của các hãng sản xuất khác nhau và phối hợp các hoạt động chuẩn hoá trong các lĩnh vực viễn thông và hệ thống thông tin. Theo mô hình OSI chương trình truyền thông được chia ra thành 7 tầng với những chức năng phân biệt cho từng tầng. Hai tầng đồng mức khi liên kết với nhau phải sử dụng một giao thức chung. Trong mô hình OSI có hai loại giao thức chính được áp dụng: giao thức có liên kết (connection - oriented) và giao thức không liên kết (connectionless)

- ***Giao thức có liên kết*:** trước khi truyền dữ liệu hai tầng đồng mức cần thiết lập một liên kết logic và các gói tin được trao đổi thông qua liên kết náy, việc có liên kết logic sẽ nâng cao độ an toàn trong truyền dữ liệu.

- ***Giao thức không liên kết*:** trước khi truyền dữ liệu không thiết lập liên kết logic và mỗi gói tin được truyền độc lập với các gói tin trước hoặc sau nó.

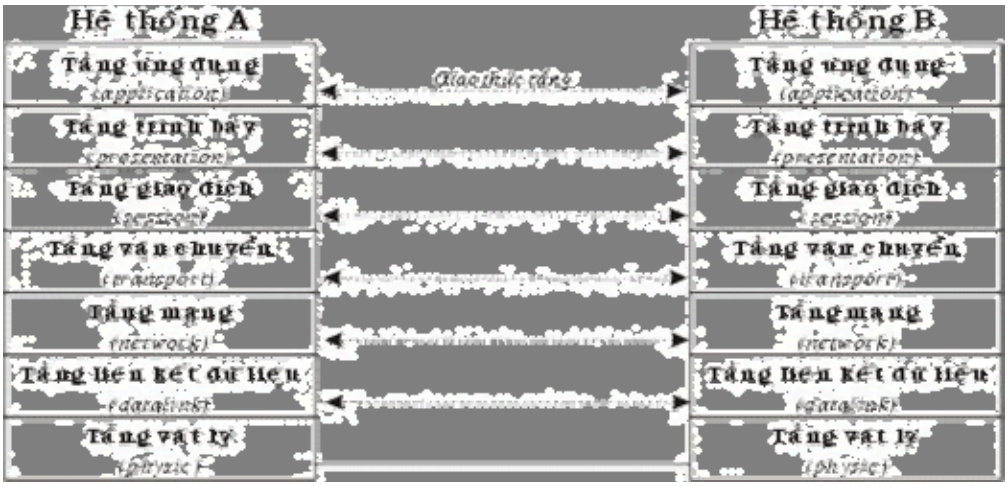
ả hiệm vụ của các tầng trong mô hình OSI:

- **Tầng ứng dụng (Application layer)**: tầng ứng dụng quy định giao diện giữa người sử dụng và môi trường OSI, nó cung cấp các phương tiện cho người sử dụng truy cập vả sử dụng các dịch vụ củ mô hình OSI.

- **Tầng trình bày (Presentation layer)**: tầng trình bày chuyển đổi các thông tin từ cú pháp người sử dụng sang cú pháp để truyền dữ liệu, ngoài ra nó có thể nén dữ liệu truyền và mã hóa chúng trước khi truyền đễ bảo mật.

- **Tầng giao dịch (Session layer)**: tầng giao dịch quy định một giao diện ứng dụng cho tầng vận chuyển sử dụng. ả ó xác lập ánh xa giữa các tên đặt địa chỉ, tạo ra các tiếp xúc ban đầu giữa các máy tính khác nhau trên cơ sở các giao dịch truyền thông. ả ó đặt tên nhất quán cho mọi thành phần muốn đối thoại riêng với nhau.

- **Tầng vận chuyển (Transport layer)**: tầng vận chuyển xác định địa chỉ trên mạng, cách thức chuyển giao gói tin trên cơ sở trực tiếp giữa hai đầu mút (end-to-end). Để bảo đảm được việc truyền ổn định trên mạng tầng vận chuyển thường đánh số các gói tin và đảm bảo chúng chuyển theo thứ tự.



Hình 3.5: Mô hình 7 tầng OSI

- **Tầng mạng (Network layer)**: tầng mạng có nhiệm vụ xác định việc chuyển hướng, vạch đường các gói tin trong mạng, các gói tin này có thể phải đi qua nhiều chặng trước khi đến được đích cuối cùng.

- **Tầng liên kết dữ liệu (Data link layer):** tầng liên kết dữ liệu có nhiệm vụ xác định cơ chế truy nhập thông tin trên mạng, các dạng thức chung trong các gói tin, đóng các gói tin...

- **Tầng vật lý (Phisical layer)**: tầng vật lý cung cấp phương thức truy cập vào đường truyền vật lý để truyền các dòng Bit không cấu trúc, ngoài ra nó cung cấp các chuẩn về điện, dây cáp, đầu nối, kỹ thuật nối mạch điện, điện áp, tốc độ cáp truyền dẫn, giao diện nối kết và các mức nối kết..

2. Mô hình SNA (Systems Netword Architecture)

Tháng 9/1973, Hãng IBM giới thiệu một kiến trúc mạng máy tính Sả A (System ả etwork Architecture). Đến năm 1977 đã có 300 trạm Sả A được cài đặt. Cuối năm 1978, số lượng đã tăng lên đến 1250, rồi cứ theo đà đó cho đến nayđã có 20.000 trạm Sả A đang được hoạt động. Qua con số này chúng ta có thể hình dung được mức độ quan trọng và tầm ảnh hưởng của Sả A trên toàn thế giới.

Cần lưu ý rằng Sả A không là một chuẩn quốc tế chính thức như OSI nhưng do vai trò to lớn của hãng IBM trên thị trường Cả TT nên Sả A trở thành một loại chuẩn thực tế và khá phổ biến. Sả A là một đặc tả gồm rất nhiều tài liệu mô tả kiến trúc của mạng xử lý dữ liệu phân tán. ả ó định nghĩa các quy tắc và các giao thức cho sự tương tác giữa các thành phần (máy tính, trạm cuối, phần mềm) trong mạng.

Sả A được tổ chức xung quanh khái niệm miền (domain). Một Sả A domain là một điểm điều khiển các dịch vụ hệ thống (Systems Services control point - SSCP) và nó sẽ điều khiển tất cả các tài nguyên đó, Các tài nguyên ở đây có thể là các đơn vị vật lý, các đơn vị logic, các liên kết dữ liệu và các thiết bị. Có thể ví SSCP như là "trái tim và khối óc" của Sả A. ả ó điều khiển Sả A domain bằng cách gói các lệnh tới một đơn vị vật lý, đơn vị vật lý này sau khi nhận được lệnh sẽ quản lý tất cả các tài nguyên trực tiếp với nó. đơn vị vật

lý thực sự là một "đối tác" của SSCP và chứa một tập con các khả năng của SSCP. Các Đơn vị vật lý đảm nhiệm việc quản lý của mỗi nút Sả A.

Sả A phân biệt giữa các nút miền con (Subarea node) và các nút ngoại vi (peripheral node).

- Một nút miền con có thể dẫn đường cho dữ liệu của người sử dụng qua toàn bộ mạng. ả ó dùng địa chỉ mạng và một số hiệu đường (router suember) để xác định đường truyền đi tới nút kế tiếp trong mạng.

- Một nút ngoại vi có tính cục bộ hơn. ả ó không dẫn đường giữa các nút miền con. Các nút được nối và điều khiển theo giao thức SDLC (Synchronous Data Link Control). Mỗi nút ngoại vi chỉ liên lạc được với nút miền con mà nó nối vào.

Mạng Sả A dựa trên cơ chế phân tầng, trước đây thì 2 hệ thống ngang hàng không được trao đổi trực tiếp. Sau này phát triển thành Sả A mở rộng: Lúc này hai tầng ngang hàng nhau có thể trao đổi trực tiếp. Với 6 tầng có tên gọi và chức năng tất như sau:

- **Tầng quản trị chức năng SNA (SNA Function Manegement)** Tầng này thật ra có thể chia tầng này làm hai tầng như sau:

- **Tầng dịch vụ giao tác (Transaction)** cung cấp các dịch vụ ứng dụng đến người dùng một mạng Sả A. ả những dịch vụ đó như : DIA cung cấp các tài liệu phân bố giữa các hệ thống văn phòng, Sả A DS (văn phòng dịch vụ phân phối) cho việc truyền thông bất đồng bộ giữa các ứng dụng phân tán và hệ thống văn phòng. Tầng dịch vụ giao tác cũng cung cấp các dịch vụ và cấu hình, các dịch vụ quản lý để điều khiển các hoạt động mạng.

- **Tầng dịch vụ trình diễn (Presentation Services):** tầng này thì liên quan với sự hiển thị các ứng dụng, người sử dụng đầu cuối và các dữ liệu hệ thống. Tầng này cũng định nghĩa các giao thức cho việc truyền thông giữa các chương trình và điều khiển truyền thông ở mức hội thoại.

- **Tầng kiểm soát luồng dữ liệu (Data flow control)** tầng này cung cấp các dịch vụ điều khiểnluồng lưu thông cho các phiên từ logic này đến đơn vị logic khác (LU - LU). ả ó thực hiện điều này bằng cách gán các số trình tự, các yêu cầu và đáp ứng, thực hiện các giao thức yêu cầu về đáp ứng giao dịch và hợp tác giữa các giao dịch gởi và nhận. ả ói chung nó yểm trợ phương thức khai thác hai chiều đồng thời (Full duplex).

- **Tầng kiểm soát truyền (Transmission control):** Tầng này cung cấp các điều khiển cơ bản của các phần tài nguyên truyền trong mạng, bằng cách xác định số trình tự nhận được, và quản lý việc theo dõi mức phiên. Tầng này cũng hỗ trợ cho việc mã hóa dữ liệu và cung cấp hệ thống hỗ trợ cho các nút ngoại vi.

- **Tầng kiểm soát đường dẫn (Path control):** Tầng này cung cấp các giao thức để tìm đường cho một gói tin qua mạng Sả A và để kết nối với các mạng Sả A khác, đồng thời nó cũng kiểm soát các đường truyền này.

- **Tầng kiểm soát liên kết dữ liệu (Data Link Control):** Tầng này cung cấp các giao thức cho việc truyền các gói tin thông qua đường truyền vật lý giữa hai node

và cũng cung cấp các điều khiển lưu thông và phục hồi lỗi, các hỗ trợ cho tầng này là các giao thức SDLC, System/370, X25, IEEE 802.2 và 802.5.

- **Tầng kiểm soát vật lý (Physical control):** Tầng này cung cấp một giao diện vật lý cho bất cứ môi trường truyền thông nào mà gắn với nó. Tầng nào định nghĩa các đặc trưng của tín hiệu cần để thiết lập, duy trì và kết thúc các đường nối vật lý cho việc hỗ trợ kết nối.

Hình 3.6: Tương ứng các tầng các kiến trúc SNI và OSI

www.viet-ebook.co.cc

Chương 4

Mô hình kết nối các hệ thống mở

Open Systems Interconection

Việc nghiên cứu về OSI được bắt đầu tại ISO vào năm 1971 với các mục tiêu nhằm nối kết các sản phẩm của các hãng sản xuất khác. Ưu điểm chính của OSI là ở chỗ nó hứa hẹn giải pháp cho vấn đề truyền thông giữa các máy tính không giống nhau. Hai hệ thống, dù có khác nhau đều có thể truyền thông với nhau một các hiệu quả nếu chúng đảm bảo những điều kiện chung sau đây:

Chúng cài đặt cùng một tập các chức năng truyền thông.

Các chức năng đó được tổ chức thành cùng một tập các tầng. các tầng đồng mức phải cung cấp các chức năng như nhau.

Các tầng đồng mức khi trao đổi với nhau sử dụng chung một giao thức

Mô hình OSI tách các mặt khác nhau của một mạng máy tính thành bảy tầng theo mô hình phân tầng. Mô hình OSI là một khung mà các tiêu chuẩn lập mạng khác nhau có thể khớp vào. Mô hình OSI định rõ các mặt nào của hoạt động của mạng có thể nhằm đến bởi các tiêu chuẩn mạng khác nhau. Vì vậy, theo một nghĩa nào đó, mô hình OSI là một loại tiêu chuẩn của các chuẩn.

I. Nguyên tắc sử dụng khi định nghĩa các tầng hệ thống mở:

Sau đây là các nguyên tắc mà ISO quy định dùng trong quá trình xây dựng mô hình OSI

- Không định nghĩa quá nhiều tầng để việc xác định và ghép nối các tầng không quá phức tạp.
- Tạo các ranh giới các tầng sao cho việc giải thích các phục vụ và số các tương tác qua lại hai tầng là nhỏ nhất.
- Tạo các tầng riêng biệt cho các chức năng khác biệt nhau hoàn toàn về kỹ thuật sử dụng hoặc quá trình thực hiên.
- Các chức năng giống nhau được đặt trong cùng một tầng.
- Lựa chọn ranh giới các tầng tại các điểm mà những thử nghiệm trong quá khứ thành công.
- Các chức năng được xác định sao cho chúng có thể dễ dàng xác định lại, và các nghi thức của chúng có thể thay đổi trên mọi hướng.
- Tạo ranh giới các tầng mà ở đó cần có những mức độ trừu tượng khác nhau trong việc sử dụng số liệu.

- Cho phép thay đổi các chức năng hoặc giao thức trong tầng không ảnh hưởng đến các tầng khác.

- Tạo các ranh giới giữa mỗi tầng với tầng trên và dưới nó.

II. Các giao thức trong mô hình OSI

Trong mô hình OSI có hai loại giao thức chính được áp dụng: giao thức có liên kết (connection - oriented) và giao thức không liên kết (connectionless).

- *Giao thức có liên kết*: trước khi truyền dữ liệu hai tầng đồng mức cần thiết lập một liên kết logic và các gói tin được trao đổi thông qua liên kết náy, việc có liên kết logic sẽ nâng cao độ an toàn trong truyền dữ liệu.

- *Giao thức không liên kết*: trước khi truyền dữ liệu không thiết lập liên kết logic và mỗi gói tin được truyền độc lập với các gói tin trước hoặc sau nó.

ả hư vậy với giao thức có liên kết, quá trình truyền thông phải gồm 3 giai đoạn phân biệt:

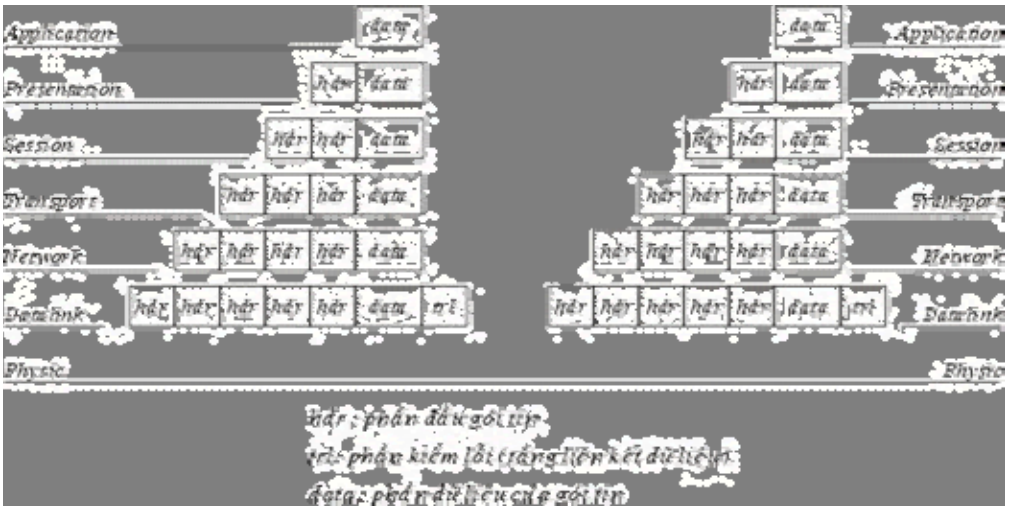
- *Thiết lập liên kết (logic):* hai thực thể đồng mức ở hai hệ thống thương lượng với nhau về tập các tham số sẽ sử dụng trong giai đoạn sau (truyền dữ liệu).

- *Truyền dữ liệu*: dữ liệu được truyền với các cơ chế kiểm soát và quản lý kèm theo (như kiểm soát lỗi, kiểm soát luồng dữ liệu, cắt/hợp dữ liệu...) để tăng cường độ tin cậy và hiệu quả của việc truyền dữ liệu.

- *Hủy bỏ liên kết (logic):* giải phóng tài nguyên hệ thống đã được cấp phát cho liên kết để dùng cho liên kết khác.

Đối với giao thức không liên kết thì chỉ có duy nhất một giai đoạn truyền dữ liệu mà thôi.

Gói tin của giao thức: Gói tin (Packet) được hiểu như là một đơn vị thông tin dùng trong việc liên lạc, chuyển giao dữ liệu trong mạng máy tính. ả hững thông điệp (message) trao đổi giữa các máy tính trong mạng, được tạo dạng thành các gói tin ở máy nguồn. Và những gói tin này khi đích sẽ được kết hợp lại thành thông điệp ban đầu. Một gói tin có thể chứa đựng các yêu cầu phục vụ, các thông tin điều khiển và dữ liệu.



Hình 4.1: Phương thức xác lập các gói tin trong mô hình OSI

Trên quan điểm mô hình mạng phân tầng tầng mỗi tầng chỉ thực hiện một chức năng là nhận dữ liệu từ tầng bên trên để chuyển giao xuống cho tầng bên dưới và ngược lại. Chức năng này thực chất là gắn thêm và gỡ bỏ phần đầu (header) đối với các gói tin trước khi chuyển nó đi. ả ói cách khác, từng gói tin bao gồm phần đầu (header) và phần dữ liệu. Khi đi đến một tầng mới gói tin sẽ được đóng thêm một phần đầu đề khác và được xem như là gói tin của tầng mới, công việc trên tiếp diễn cho tới khi gói tin được truyền lên đường dây mạng để đến bên nhận.

Tại bên nhận các gói tin được gỡ bỏ phần đầu trên từng tầng tướng ứng và đây cũng là nguyên lý của bất cứ mô hình phân tầng nào.

Chú ý: Trong mô hình OSI phần kiểm lỗi của gói tin tầng liên kết dữ liệu đặt ở cuối gói tin

III. Các chức năng chủ yếu của các tầng của mô hình OSI.

Tầng 1: Vật lý (Physical)

Tầng vật lý (Physical layer) là tầng dưới cùng của mô hình OSI là. ả ó mô tả các đặc trưng vật lý của mạng: Các loại cáp được dùng để nối các thiết bị, các loại đầu nối được dùng , các dây cáp có thể dài bao nhiêu v.v... Mặt khác các tầng vật lý cung cấp các đặc trưng điện của các tín hiệu được dùng để khi chuyển dữ liệu trên cáp từ một máy này đến một máy khác của mạng, kỹ thuật nối mạch điện, tốc độ cáp truyền dẫn.

Tầng vật lý không qui định một ý nghĩa nào cho các tín hiệu đó ngoài các giá trị nhị phân 0 và 1. Ở các tầng cao hơn của mô hình OSI ý nghĩa của các bit được truyền ở tầng vật lý sẽ được xác định.

Ví dụ: Tiêu chuẩn Ethernet cho cáp xoắn đôi 10 baseT định rõ các đặc trưng điện của cáp xoắn đôi, kích thước và dạng của các đầu nối, độ dài tối đa của cáp.

Khác với các tầng khác, tầng vật lý là không có gói tin riêng và do vậy không có phần đầu (header) chứa thông tin điều khiển, dữ liệu được truyền đi theo dòng bit. Một giao thức tầng vật lý tồn tại giữa các tầng vật lý để quy định về phương thức truyền (đồng bộ, phi đồng bộ), tốc độ truyền.

Các giao thức được xây dựng cho tầng vật lý được phân chia thành phân chia thành hai loại giao thức sử dụng phương thức truyền thông dị bộ (asynchronous) và phương thức truyền thông đồng bộ (synchronous).

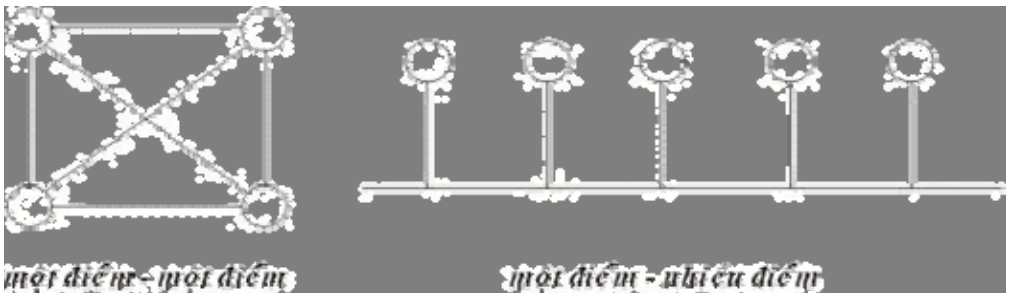
- *Phương thức truyền dị bộ*: không có một tín hiệu quy định cho sự đồng bộ giữa các bit giữa máy gửi và máy nhận, trong quá trình gửi tín hiệu máy gửi sử dụng các bit đặc biệt START và STOP được dùng để tách các xâu bit biểu diễn các ký tự trong dòng dữ liệu cần truyền đi. ả ó cho phép một ký tự được truyền đi bất kỳ lúc nào mà không cần quan tâm đến các tín hiệu đồng bộ trước đó.

- *Phương thức truyền đồng bộ*: sử dụng phương thức truyền cần có đồng bộ giữa máy gửi và máy nhận, nó chèn các ký tự đặc biệt như SYả (Synchronization), EOT (End Of Transmission) hay đơn giản hơn, một cái "cờ " (flag) giữa các dữ liệu của máy gửi để báo hiệu cho máy nhận biết được dữ liệu đang đến hoặc đã đến.

Tầng 2: Liên kết dữ liệu (Data link)

Tầng liên kết dữ liệu (data link layer) là tầng mà ở đó ý nghĩa được gán cho các bít được truyền trên mạng. Tầng liên kết dữ liệu phải quy định được các dạng thức, kích thước, địa chỉ máy gửi và nhận của mỗi gói tin được gửi đi. ả ó phải xác định cơ chế truy nhập thông tin trên mạng và phương tiện gửi mỗi gói tin sao cho nó được đưa đến cho người nhận đã định.

Tầng liên kết dữ liệu có hai phương thức liên kết dựa trên cách kết nối các máy tính, đó là phương thức "một điểm - một điểm" và phương thức "một điểm - nhiều điểm". Với phương thức "một điểm - một điểm" các đường truyền riêng biệt được thiết lập để nối các cặp máy tính lại với nhau. Phương thức "một điểm - nhiều điểm " tất cả các máy phân chia chung một đường truyền vật lý.



Hình 4.2: Các đường truyền kết nối kiểu "một điểm - một điểm" và "một điểm - nhiều điểm".

Tầng liên kết dữ liệu cũng cung cấp cách phát hiện và sửa lỗi cơ bản để đảm bảo cho dữ liệu nhận được giống hoàn toàn với dữ liệu gửi đi. ả ếu một gói tin có lỗi không sửa được, tầng liên kết dữ liệu phải chỉ ra được cách thông báo cho nơi gửi biết gói tin đó có lỗi để nó gửi lại.

Các giao thức tầng liên kết dữ liệu chia làm 2 loại chính là các giao thức hướng ký tư và các giao thức hướng bit. Các giao thức hướng ký tự được xây dựng dựa trên các ký tự đặc biệt của một bộ mã chuẩn nào đó (như ASCII hay EBCDIC), trong khi đó các giao thức hướng bit lại dùng các cấu trúc nhị phân (xâu bit) để xây dựng các phần tử của giao thức (đơn vị dữ liệu, các thủ tục.) và khi nhận, dữ liệu sẽ được tiếp nhận lần lượt từng bit một.

Tầng 3: Mạng (Network)

Tầng mạng (network layer) nhắm đến việc kết nối các mạng với nhau bằng cách tìm đường (routing) cho các gói tin từ một mạng này đến một mạng khác. ả ó xác định việc chuyển hướng, vạch đường các gói tin trong mạng, các gói này có thể phải đi qua nhiều chặng trước khi đến được đích cuối cùng. ả ó luôn tìm các tuyến truyền thông không tắc nghẽn để đưa các gói tin đến đích.

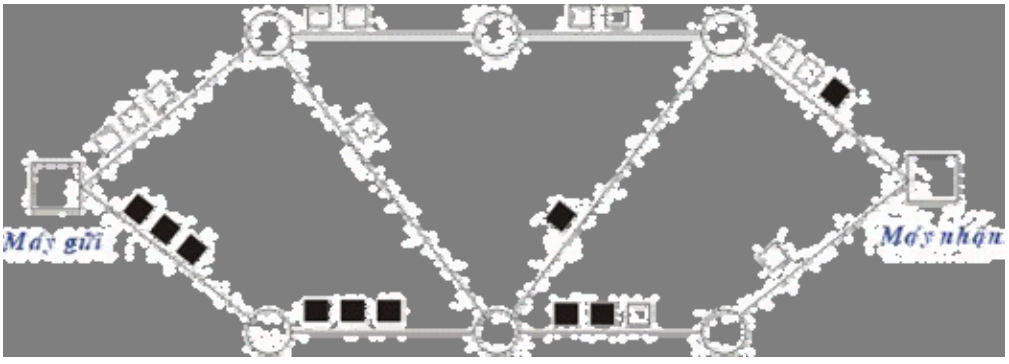
Tầng mạng cung các các phương tiện để truyền các gói tin qua mạng, thậm chí qua một mạng của mạng (network of network). Bởi vậy nó cần phải đáp ứng với nhiều kiểu mạng và nhiều kiểu dịch vụ cung cấp bởi các mạng khác nhau. hai chức năng chủ yếu của tầng mạng là chọn đường (routing) và chuyển tiếp (relaying). Tầng mạng là quan trọng nhất khi liên kết hai loại mạng khác nhau như mạng Ethernet với mạng Token Ring khi đó phải dùng một bộ tìm đường (quy định bởi tầng mạng) để chuyển các gói tin từ mạng này sang mạng khác và ngược lại.

Đối với một mạng chuyển mạch gói (packet - switched network) - gồm tập hợp các nút chuyển mạch gói nối với nhau bởi các liên kết dữ liệu. Các gói dữ liệu được truyền từ một hệ thống mở tới một hệ thống mở khác trên mạng phải được chuyển qua một chuỗi các nút. Mỗi nút nhận gói dữ liệu từ một đường vào (incoming link) rồi chuyển tiếp nó tới một đường ra (outgoing link) hướng đến đích của dữ liệu. ả hư vậy ở mỗi nút trung gian nó phải thực hiện các chức năng chọn đường và chuyển tiếp.

Việc chọn đường là sự lựa chọn một con đường để truyền một đơn vị dữ liệu (một gói tin chẳng hạn) từ trạm nguồn tới trạm đích của nó. Một kỹ thuật chọn đường phải thực hiện hai chức năng chính sau đây:

- Quyết định chọn đường tối ưu dựa trên các thông tin đã có về mạng tại thời điểm đó thông qua những tiêu chuẩn tối ưu nhất định.

- Cập nhật các thông tin về mạng, tức là thông tin dùng cho việc chọn đường, trên mạng luôn có sự thay đổi thường xuyên nên việc cập nhật là việc cần thiết.



Hình 4. 3: Mô hình chuyển vận các gói tin trong mạng chuyển mạch gói

ả gười ta có hai phương thức đáp ứng cho việc chọn đường là phương thức xử lý tập trung và xử lý tại chỗ.

- *Phương thức chọn đường xử lý tập trung* được đặc trưng bởi sự tồn tại của một (hoặc vài) trung tâm điều khiển mạng, chúng thực hiện việc lập ra các bảng đường đi tại từng thời điểm cho các nút và sau đó gửi các bảng chọn đường tới từng nút dọc theo con đường đã được chọn đó. Thông tin tổng thể của mạng cần dùng cho việc chọn đường chỉ cần cập nhập và được cất giữ tại trung tâm điều khiển mạng.

- *Phương thức chọn đường xử lý tại chỗ* được đặc trưng bởi việc chọn đường được thực hiện tại mỗi nút của mạng. Trong từng thời điểm, mỗi nút phải duy trì các thông tin của mạng và tự xây dựng bảng chọn đường cho mình. ả hư vậy các thông tin tổng thể của mạng cần dùng cho việc chọn đường cần cập nhập và được cất giữ tại mỗi nút.

Thông thường các thông tin được đo lường và sử dụng cho việc chọn đường bao gồm:

- Trạng thái của đường truyền.
- Thời gian trễ khi truyền trên mỗi đường dẫn.
- Mức độ lưu thông trên mỗi đường.
- Các tài nguyên khả dụng của mạng.

Khi có sự thay đổi trên mạng (ví dụ thay đổi về cấu trúc của mạng do sự cố tại một vài nút, phục hồi của một nút mạng, nối thêm một nút mới... hoặc thay đổi về mức độ lưu thông) các thông tin trên cần được cập nhật vào các cơ sở dữ liệu về trạng thái của mạng.

Hiện nay khi nhu cầu truyền thông đa phương tiện (tích hợp dữ liệu văn bản, đồ hoạ, hình ảnh, âm thanh) ngày càng phát triển đòi hỏi các công nghệ truyền dẫn tốc độ cao nên việc phát triển các hệ thống chọn đường tốc độ cao đang rất được quan tâm.

Tầng 4: Vận chuyển (Transport)

Tầng vận chuyển cung cấp các chức năng cần thiết giữa tầng mạng và các tầng trên. nó là tầng cao nhất có liên quan đến các giao thức trao đổi dữ liệu giữa các hệ thống mở. ả ó cùng các tầng dưới cung cấp cho người sử dụng các phục vụ vận chuyển.

Tầng vận chuyển (transport layer) là tầng cơ sở mà ở đó một máy tính của mạng chia sẻ thông tin với một máy khác. Tầng vận chuyển đồng nhất mỗi trạm bằng một địa chỉ duy nhất và quản lý sự kết nối giữa các trạm. Tầng vận chuyển cũng chia các gói tin lớn thành các gói tin nhỏ hơn trước khi gửi đi. Thông thường tầng vận chuyển đánh số các gói tin và đảm bảo chúng chuyển theo đúng thứ tự.

Tầng vận chuyển là tầng cuối cùng chịu trách nhiệm về mức độ an toàn trong truyền dữ liệu nên giao thức tầng vận chuyển phụ thuộc rất nhiều vào bản chất của tầng mạng. ả gười ta chia giao thức tầng mạng thành các loại sau:

- Mạng loại A: Có tỷ suất lỗi và sự cố có báo hiệu chấp nhận được (tức là chất lượng chấp nhận được). Các gói tin được giả thiết là không bị mất. Tầng vận chuyển không cần cung cấp các dịch vụ phục hồi hoặc sắp xếp thứ tự lại.

- Mạng loại B: Có tỷ suất lỗi chấp nhận được nhưng tỷ suất sự cố có báo hiệu lại không chấp nhận được. Tầng giao vận phải có khả năng phục hồi lại khi xẩy ra sự cố.

- Mạng loại C: Có tỷ suất lỗi không chấp nhận được (không tin cậy) hay là giao thức không liên kết. Tầng giao vận phải có khả năng phục hồi lại khi xảy ra lỗi và sắp xếp lại thứ tự các gói tin.

Trên cơ sở loại giao thức tầng mạng chúng ta có 5 lớp giao thức tầng vận chuyển đó là:

- *Giao thức lớp 0 (Simple Class - lớp đơn giản)*: cung cấp các khả năng rất đơn giản để thiết lập liên kết, truyền dữ liệu và hủy bỏ liên kết trên mạng "có liên kết" loại A. ả ó có khả năng phát hiện và báo hiệu các lỗi nhưng không có khả năng phục hồi.

- *Giao thức lớp 1 (Basic Error Recovery Class - Lớp phục hồi lỗi cơ bản)* dùng với các loại mạng B, ở đây các gói tin (TPDU) được đánh số. ả goài ra giao thức còn có khả năng báo nhận cho nơi gửi và truyền dữ liệu khẩn. So với giao thức lớp 0 giao thức lớp 1 có thêm khả năng phục hồi lỗi.

- *Giao thức lớp 2 (Multiplexing Class - lớp dồn kênh)* là một cải tiến của lớp 0 cho phép dồn một số liên kết chuyển vận vào một liên kết mạng duy nhất, đồng thời có thể kiểm soát luồng dữ liệu để tránh tắc nghẽn. Giao thức lớp 2 không có khả năng phát hiện và phục hồi lỗi. Do vậy nó cần đặt trên một tầng mạng loại A.

- *Giao thức lớp 3 (Error Recovery and Multiplexing Class - lớp phục hồi lỗi cơ bản và dồn kênh)* là sự mở rộng giao thức lớp 2 với khả năng phát hiện và phục hồi lỗi, nó cần đặt trên một tầng mạng loại B.

- *Giao thức lớp 4 (Error Detection and Recovery Class - Lớp phát hiện và phục hồi lỗi)* là lớp có hầu hết các chức năng của các lớp trước và còn bổ sung thêm một số khả năng khác để kiểm soát việc truyền dữ liệu.

Tầng 5: Giao dịch (Session)

Tầng giao dịch (session layer) thiết lập "các giao dịch" giữa các trạm trên mạng, nó đặt tên nhất quán cho mọi thành phần muốn đối thoại với nhau và lập ánh xa giữa các tên với địa chỉ của chúng. Một giao dịch phải được thiết lập trước khi dữ liệu được truyền trên mạng, tầng giao dịch đảm bảo cho các giao dịch được thiết lập và duy trì theo đúng qui định.

Tầng giao dịch còn cung cấp cho người sử dụng các chức năng cần thiết để quản trị các giao dịnh ứng dụng của họ, cụ thể là:

- Điều phối việc trao đổi dữ liệu giữa các ứng dụng bằng cách thiết lập và giải phóng (một cách lôgic) các phiên (hay còn gọi là các hội thoại - dialogues)

- Cung cấp các điểm đồng bộ để kiểm soát việc trao đổi dữ liệu.
- Áp đặt các qui tắc cho các tương tác giữa các ứng dụng của người sử dụng.
- Cung cấp cơ chế "lấy lượt" (nắm quyền) trong quá trình trao đổi dữ liệu.

Trong trường hợp mạng là hai chiều luân phiên thì nẩy sinh vấn đề: hai người sử dụng luân phiên phải "lấy lượt" để truyền dữ liệu. Tầng giao dịch duy trì tương tác luân phiên bằng cách báo cho mỗi người sử dụng khi đến lượt họ được truyền dữ liệu. Vấn đề đồng bộ hóa trong tầng giao dịch cũng được thực hiện như cơ chế kiểm tra/phục hồi, dịch vụ này cho phép người sử dụng xác định các điểm đồng bộ hóa trong dòng dữ liệu đang chuyển vận và khi cần thiết có thể khôi phục việc hội thoại bắt đầu từ một trong các điểm đó

Ở một thời điểm chỉ có một người sử dụng đó quyền đặc biệt được gọi các dịch vụ nhất định của tầng giao dịch, việc phân bổ các quyền này thông qua trao đổi thẻ bài (token). Ví dụ: Ai có được token sẽ có quyền truyền dữ liệu, và khi người giữ token trao token cho người khác thi cũng có nghĩa trao quyền truyền dữ liệu cho người đó.

Tầng giao dịch có các hàm cơ bản sau:

- *Give Token* cho phép người sử dụng chuyển một token cho một người sử dụng khác của một liên kết giao dịch.
- *Please Token* cho phép một người sử dụng chưa có token có thể yêu cầu token đó.
- *Give Control* dùng để chuyển tất cả các token từ một người sử dụng sang một người sử dụng khác.

Tầng 6: Trình bày (Presentation)

Trong giao tiếp giữa các ứng dụng thông qua mạng với cùng một dữ liệu có thể có nhiều cách biểu diễn khác nhau. Thông thường dạng biểu diễn dùng bởi ứng dụng nguồn và dạng biểu diễn dùng bởi ứng dụng đích có thể khác nhau do các ứng dụng được chạy trên các hệ thống hoàn toàn khác nhau (như hệ máy Intel và hệ máy Motorola). Tầng trình bày (Presentation layer) phải chịu trách nhiệm chuyển đổi dữ liệu gửi đi trên mạng từ một loại biểu diễn này sang một loại khác. Để đạt được điều đó nó cung cấp một dạng biểu diễn chung dùng để truyền thông và cho phép chuyển đổi từ dạng biểu diễn cục bộ sang biểu diễn chung và ngược lại.

Tầng trình bày cũng có thể được dùng kĩ thuật mã hóa để xáo trộn các dữ liệu trước khi được truyền đi và giải mã ở đầu đến để bảo mật. ả goài ra tầng biểu diễn cũng có thể dùng các kĩ thuật nén sao cho chỉ cần một ít byte dữ liệu để thể hiện thông tin khi nó được truyền ở trên mạng, ở đầu nhận, tầng trình bày bung trở lại để được dữ liệu ban đầu.

Tầng 7: Ứng dụng (Application)

Tầng ứng dụng (Application layer) là tầng cao nhất của mô hình OSI, nó xác định giao diện giữa người sử dụng và môi trường OSI và giải quyết các kỹ thuật mà các chương trình ứng dụng dùng để giao tiếp với mạng.

Để cung cấp phương tiện truy nhập môi trường OSI cho các tiến trình ứng dụng, ả gười ta thiết lập các thực thể ứng dụng (AE), các thực thể ứng dụng sẽ gọi đến các phần tử dịch vụ ứng dụng (Application Service Element - viết tắt là ASE) của chúng. Mỗi thực thể ứng dụng có thể gồm một hoặc nhiều các phần tử dịch vụ ứng dụng. Các phần tử dịch vụ ứng dụng được phối hợp trong môi trường của thực thể ứng dụng thông qua các liên kết (association) gọi là đối tượng liên kết đơn (Single Association Object - viết tắt là SAO). SAO điều khiển việc truyền thông trong suốt vòng đời của liên kết đó cho phép tuần tự hóa các sự kiện đến từ các ASE thành tố của nó.

www.viet-ebook.co.cc

Chương 5

Các đặc tính kỹ thuật của mạng cục bộ

Trên thực tế mạng cục bộ là một hệ thống truyền dữ liệu giữa các máy tính với một khoảng cách tương đối hẹp, điều đó cho phép có những lựa chọn đa dạng về thiết bị . Tuy nhiên những lựa chọn đa dạng này lại bị hạn chế bởi các đặc tính kỹ thuật của mạng cục bộ, đó làø tập hợp các quy tắc chuẩn đã được quy ước mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để đảm bảo cho mạng hoạt động tốt. Các đặc tính chính của mạng cục bộ mà chúng ta nói tới sau đây là:

- Cấu trúc của mạng (hay topology của mạng mà qua đó thể hiện cách nối các mạng máy tính với nhau ra sao).
- Các nghi thức truyền dữ liệu trên mạng (các thủ tục hướng dẫn trạm làm việc làm thế nào và lúc nào có thể thâm nhập vào đường dây cáp để gửi các gói thông tin).
- Các loại đường truyền và các chuẩn của chúng .
- Các phương thức tín hiệu

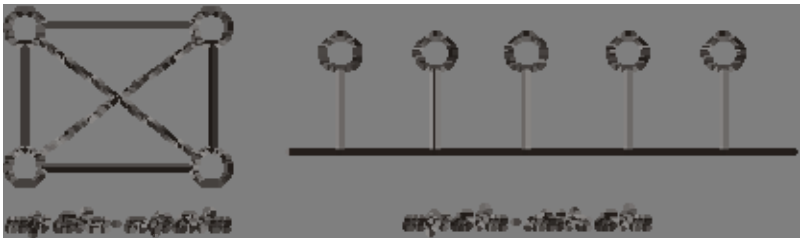
I. Cấu trúc của mạng (Topology)

Hình trạng của mạng cục bộ thể hiện qua cấu trúc hay hình dáng hình học cuả các đường dây cáp mạng dùng để liên kết các máy tính thuộc mạng với nhau. Các mạng cục bộ thường hoạt động dựa trên cấu trúc đã định saün liên kết các máy tính và các thiết bị có liên quan.

Trước hết chúng ta xem xét hai phương thức nối mạng chủ yếu được sử dụng trong việc liên kết các máy tính là "một điểm - một điểm" và "một điểm - nhiều điểm ".

Với phương thức "một điểm - một điểm" các đường truyền riêng biệt được thiết lập để nối các cặp máy tính lại với nhau. Mỗi máy tính có thể truyền và nhận trực tiếp dữ liệu hoặc có thể làm trung gian như lưu trữ những dữ liệu mà nó nhận được rồi sau đó chuyển tiếp dữ liệu đi cho một máy khác để dữ liệu đó đạt tới đích.

Theo phương thức "một điểm - nhiều điểm " tất cả các trạm phân chia chung một đường truyền vật lý. Dữ liệu được gửi đi từ một máy tính sẽ có thể được tiếp nhận bởi tất cả các máy tính còn lại, bởi vậy cần chỉ ra địa chỉ đích của dữ liệu để mỗi máy tính căn cứ vào đó kiểm tra xem dữ liệu có phải dành cho mình không nếu đúng thì nhận còn nếu không thì bỏ qua.



Hình 5.1: Các phương thức liên kết mạng

Tùy theo cấu trúc của mỗi mạng chúng sẽ thuộc vào một trong hai phương thức nối mạng và mỗi phương thức nối mạng sẽ có những yêu cầu khác nhau về phần cứng và phần mềm.

II. Những cấu trúc chính của mạng cục bộ

1. Dạng đường thẳng (Bus)

Trong dạng đường thẳng các máy tính đều được nối vào một đường dây truyền chính (bus). Đường truyền chính này được giới hạn hai đầu bởi một loại đầu nối đặc biệt gọi là *terminator* (dùng để nhận biết là đầu cuối để kết thúc đường truyền tại đây). Mỗi trạm được nối vào bus qua một đầu nối chữ T (T_connector) hoặc một bộ thu phát (transceiver). Khi một trạm truyền dữ liệu, tín hiệu được truyền trên cả hai chiều của đường truyền theo từng gói một, mỗi gói đều phải mang địa chỉ trạm đích. Các trạm khi thấy dữ liệu đi qua nhận lấy, kiểm tra, nếu đúng với địa chỉ của mình thì nó nhận lấy còn nếu không phải thì bỏ qua.

Sau đây là vài thông số kỹ thuật của topology bus. Theo chuẩn IEEE 802.3 (cho mạng cục bộ) với cách đặt tên qui ước theo thông số: tốc độ truyền tính hiệu (1,10 hoặc 100 Mb/s); BASE (nếu là Baseband) hoặc BROAD (nếu là Broadband).

- 10BASE5: Dùng cáp đồng trục đường kính lớn (10mm) với trở kháng 50 Ohm, tốc độ 10 Mb/s, phạm vi tín hiệu 500m/segment, có tối đa 100 trạm, khoảng cách giữa 2 tranceiver tối thiểu 2,5m (Phương án này còn gọi là Thick Ethernet hay Thicknet)

- 10BASE2: tương tự như Thicknet nhưng dùng cáp đồng trục nhỏ (RG 58A), có thể chạy với khoảng cách 185m, số trạm tối đa trong 1 segment là 30, khoảng cách giữa hai máy tối thiểu là 0,5m.

Dạng kết nối này có ưu điểm là ít tốn dây cáp, tốc độ truyền dữ liệu cao tuy nhiên nếu lưu lượng truyền tăng cao thì dễ gây ách tắc và nếu có trục trặc trên hành lang chính thì khó phát hiện ra.

Hiện nay các mạng sử dụng hình dạng đường thẳng là mạng Ethernet và G-net.

2. Dạng vòng tròn (Ring)

Các máy tính được liên kết với nhau thành một vòng tròn theo phương thức "một điểm - một điểm ", qua đó mỗi một trạm có thể nhận và truyền dữ liệu theo vòng một chiều và dữ liệu được truyền theo từng gói một. Mỗi gói dữ liệu đều có mang địa chỉ trạm đích, mỗi trạm khi nhận được một gói dữ liệu nó kiểm tra nếu đúng với địa chỉ của mình thì nó nhận lấy còn nếu không phải thì nó sẽ phát lại cho trạm kế tiếp, cứ như vậy gói dữ liệu đi được đến đích. Với dạng kết nối này có ưu điểm là không tốn nhiều dây cáp, tốc độ truyền dữ liệu cao, không gây ách tắc tuy nhiên các giao thức để truyền dữ liệu phức tạp và nếu có trục trặc trên một trạm thì cũng ảnh hưởng đến toàn mạng.

Hiện nay các mạng sử dụng hình dạng vòng tròn là mạng Tocken ring của IBM.

3. Dạng hình sao (Star)

Ở dạng hình sao, tất cả các trạm được nối vào một thiết bị trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển tín hiệu đến trạm đích với phương thức kết nối là phương thức "một điểm - một điểm ". Thiết bị trung tâm hoạt động giống như một tổng đài cho phép thực hiện việc nhận và truyền dữ liệu từ trạm này tới các trạm khác. Tùy theo yêu cầu truyền thông trong mạng , thiết bị trung tâm có thể là một bộ chuyển mạch (switch), một bộ chọn đường (router) hoặc đơn giản là một bộ phân kênh (Hub). Có nhiều cổng ra và mỗi cổng nối với một máy. Theo chuẩn IEEE 802.3 mô hình dạng Star thường dùng:

- 10BASE-T: dùng cáp UTP, tốc độ 10 Mb/s, khoảng cách từ thiết bị trung tâm tới trạm tối đa là 100m.

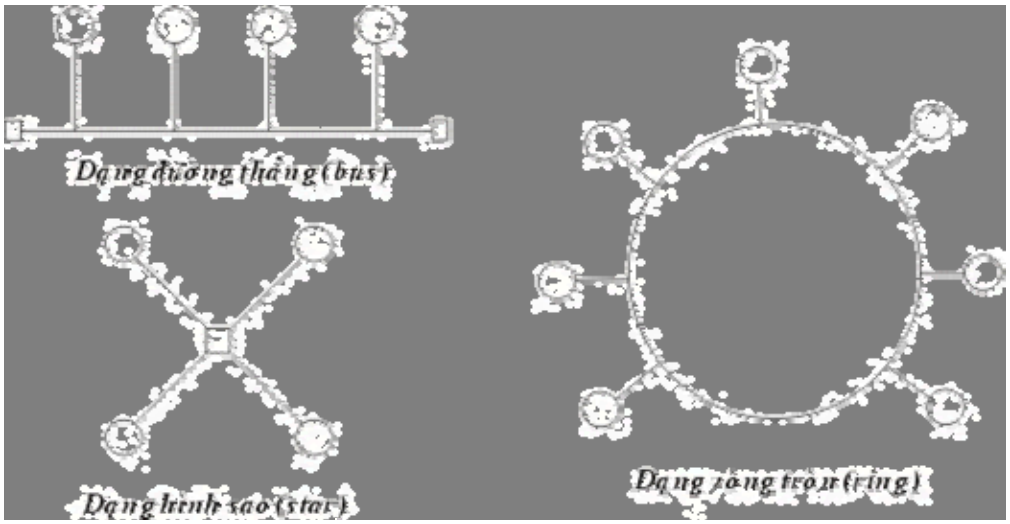
- 100BASE-T tương tự như 10BASE-T nhưng tốc độ cao hơn 100 Mb/s.

Ưu và khuyết điểm

- *Ưu điểm:* Với dạng kết nối này có ưu điểm là không đụng độ hay ách tắc trên đường truyền, lắp đặt đơn giản, dễ dàng cấu hình lại (thêm, bớt trạm). ấ ếu có trục trặc trên một trạm thì cũng không gây ảnh hưởng đến toàn mạng qua đó dễ dàng kiểm soát và khắc phục sự cố.

- *Nhược điểm:* Độ dài đường truyền nối một trạm với thiết bị trung tâm bị hạn chế (trong vòng 100 m với công nghệ hiện đại) tốn đường dây cáp nhiều, tốc độ truyền dữ liệu không cao.

Hiện nay các mạng sử dụng hình dạng hình sao là mạng STARLAấ của AT&T và S-ấ ET của ấ ovell.



Hình 5.2 : Các loại cấu trúc chính của mạng cục bộ.

	Đường thẳng	Vòng Tròn	Hình sao
Ứng dụng	Tốt cho trường hợp mạng nhỏ và mạng có giao thông thấp và	Tốt cho trường hợp mạng có số trạm ít hoạt động với tốc độ cao,không cách nhau xa lắm hoặc mạng có lưu lượng dữ	hiên nay mạng sao là cách tốt nhất cho trường hợp phải tích hợp dữ liệu và tín hiệutiếng.Các

	lưu lượng dữ liệu thấp	liệu phân bố không đều.	mạng đện thoại công cộng có cấu trúc này
Độ phức tạp	Tương đối không phức tạp	Đòi hỏi thiết bị tương đối phức tạp .Mặt khác việc đưa thông điệp đi trên tuyến là đơn giản, vì chỉ có 1 con đường, trạm phát chỉ cần biết địa chỉ của trạm nhận , các thông tin để dẫn đường khác thì không cần thiết	Mạng sao được xem là khá phức tạp . Các trạm được nối với thiết bị trung tâm và lần lượt hoạt động như thiết bị trung tâm hoặc nối được tới các dây dẫn truyền từ xa
Hiệu suất	Rất tốt dưới tải thấp có thể giảm hiệu suất rất mau khi tải tăng	Có hiệu quả trong trường hợp lượng lưu thông cao và khá ổn định nhờ sự tăng chậm thời gian trễ và sự xuống cấp so với các mạng khác	Tốt cho trường hợp tải vừa tuy nhiên kích thước và khả năng , suy ra hiệu suất của mạng phụ thuộc trực tiếp vào sức mạnh của thiết bị trung tâm.
Tổng phí	Tương đối thấp đặc biệt do nhiều thiết bị đã phát triển hòa chỉnh và bán sảm phẩm ở thị trường .Sự dư thừa kênh truyền được khuyến để giảm bớt nguy cơ xuất hiện sự cố trên mạng	Phải dự trù gấp đôi nguồn lực hoặc phải có 1 phương thức thay thế khi 1 nút không hoạt động nếu vẫn muốn mạng hoạt động bình thường	Tổng phí rất cao khi làm nhiêm vụ của thiết bị trung tâm, thiết bị trung tâm ï không được dùng vào việc khác .Số lượng dây riêng cũng nhiều.
ẩ guy cơ	Một trạm bị hỏng không ảnh hưởng đến cả mạng. Tuy nhiên mạng sẽ có nguy cơ bị tổn hại khi sự cố trên đường dây dẫn chính hoặc có vấn đề với tuyến. Vấn đề trên rất khó xác định được lại rất dễ sửa chữa	Mơt trạm bị hỏng có thể ảnh hưởng đến cả hệ thống vì các trạm phục thuộc vào nhau. Tìm 1 repeater hỏng rất khó ,vả lại việc sửa chữa thẳng hay dùng mưu mẹo xác định điểm hỏng trên mạng có địa bàn rôăng rất khó	Độ tin cậy của hệ thống phụ thuộc vào thiết bị trung tâm, .nếu bị hỏng thì mạng ngưng hoạt động Sự ngưng hoạt động tại thiết bị trung tâm thường không ảnh hươdng đến toàn bộ hệ thống .
Khả năng mở rộng	Việc thêm và định hình lại mạng này rất dễ.Tuy nhiên việc kết nối giữa các máy tính và thiết bị của các hãng khác nhau khó có thể vì chúng phải có thể nhận cùng	Tương đối dễ thêm và bớt các trạm làm việc mà không phải nối kết nhiều cho mỗi thay đổi Giá thành cho việc thay đổi tương đối thấp	Khả năng mở rộâng hạn chế, đa số các thiết bị trung tâm chỉ chịu đựng nối 1 số nhất định liên kết. Sự hạn chế về tốc độ truyền dữ liệu và băng tần thường được đòi hỏi ở mỗi người sử dụng. Các hạn chế này giúp cho các chức năng xử lý trung tâm không

	địa chỉ và dữ liệu		bị quá tải bởi tốc độ thu nạp tại tại cổng truyền và giá thành mỗi cổng truyền của thiết bị trung tâm thấp .

Hình 6.4 : Bảng so sánh tính năng giữa các cấu trúc của mạng LAN

III. Phương thức truyền tín hiệu

Thông thường có hai phương thức truyền tín hiệu trong mạng cục bộ là dùng băng tần cơ sở (baseband) và băng tần rộng (broadband). Sự khác nhau chủ yếu giữa hai phương thức truyền tín hiệu này là băng tầng cơ sở chỉ chấp nhận một kênh dữ liệu duy nhất trong khi băng rộng có thể chấp nhận đồng thời hai hoặc nhiều kênh truyền thông cùng phân chia giải thông của đường truyền.

Hầu hết các mạng cục bộ sử dụng phương thức băng tần cơ sở. Với phương thức truyền tín hiệu này này tín hiệu có thể được truyền đi dưới cả hai dạng: tương tự (analog) hoặc số (digital). Phương thức truyền băng tần rộng chia giải thông (tần số) của đường truyền thành nhiều giải tần con trong đó mỗi dải tần con đó cung cấp một kênh truyền dữ liệu tách biệt nhờ sử dụng một cặp modem đặc biệt gọi là bộ giải / Điều biến RF cai quản việc biến đổi các tín hiệu số thành tín hiệu tương tự có tần số vô tuyến (RF) bằng kỹ thuật ghép kênh.

IV. Các giao thức truy cập đường truyền trên mạng LAN

Để truyền được dữ liệu trên mạng người ta phải có các thủ tục nhằm hướng dẫn các máy tính của mạng làm thế nào và lúc nào có thể thâm nhập vào đường dây cáp để gửi các gói dữ kiện. Ví dụ như đối với các dạng bus và ring thì chỉ có một đường truyền duy nhất nối các trạm với nhau, cho nên cần phải có các quy tắc chung cho tất cả các trạm nối vào mạng để đảm bảo rằng đường truyền được truy nhập và sử dụng một cách hợp lý.

Có nhiều giao thức khác nhau để truy nhập đường truyền vật lý nhưng phân thành hai loại: các giao thức truy nhập ngẫu nhiên và các giao thức truy nhập có điều khiển.

1. Giao thức chuyển mạch (yêu cầu và chấp nhận)

Giao thức chuyển mạch là loại giao thức hoạt động theo cách thức sau: một máy tính của mạng khi cần có thể phát tín hiệu thâm nhập vào mạng, nếu vào lúc này đường cáp không bận thì mạch điều khiển sẽ cho trạm này thâm nhập vào đường cáp còn nếu đường cáp đang bận, nghĩa là đang có giao lưu giữa các trạm khác, thì việc thâm nhập sẽ bị từ chối.

2. Giao thức đường dây đa truy cập với cảm nhận va chạm (Carrier Sense Multiple Access with Collision Detection hay CSMA/CD)

Giao thức đường dây đa truy cập cho phép nhiều trạm thâm nhập cùng một lúc vào mạng, giao thức này thường dùng trong sơ đồ mạng dạng đường thẳng. Mọi trạm đều có thể được truy nhập vào đường dây chung một cách ngẫu nhiên và do vậy có thể dẫn đến xung đột (hai hoặc nhiều trạm đồng thời cùng truyền dữ liệu). Các trạm phải kiểm tra đường truyền gói dữ liệu đi qua có phải của nó hay không. Khi một trạm muốn truyền dữ liệu nó phải

kiểm tra đường truyền xem có rảnh hay không để gửi gói dữ liệu của, nếu đường truyền đang bận trạm phải chờ đợi chỉ được truyền khi thấy đường truyền rảnh. ả ếu cùng một lúc có hai trạm cùng sử dụng đường truyền thì giao thức phải phát hiện điều này và các trạm phải ngưng thâm nhập, chờ đợi lần sau các thời gian ngẫu nhiên khác nhau.

Khi đường cáp đang bận trạm phải chờ đợi theo một trong ba phương thức sau:

- Trạm tạm chờ đợi một thời gian ngẫu nhiên nào đó rồi lại bắt đầu kiểm tra đường truyền.
- Trạm tiếp tục kiểm tra đường truyền đến khi đường truyền rảnh thì truyền dữ liệu đi.
- Trạm tiếp tục kiểm tra đường truyềnđến khi đường truyền rảnh thì truyền dữ liệu đi với xác suất p xác định trước ($0 < p < 1$).

Tại đây phương thức 1 có hiệu quả trong việc tránh xung đột vì hai trạm cần truyền khi thấy đường truyền bận sẽ cùng rút lui và chờ đợi trong các thời gian ngẫu nhiên khác nhau. ả gược lại phương thức 2 cố gắng giảm thời gian trống của đường truyền bằng các cho phép trạm có thể truyền ngay sau khi một cuộc truyền kết thúc song nếu lúc đó có thêm một trạm khác đang đợi thì khả năng xảy ra xung đột là rất cao. Phương thức 3 với giá trị p phải lựa chọn hợp lý có thể tối thiểu hóa được khả năng xung đột lẫn thời gian trống của đường truyền.

Khi lưu lượng các gói dữ liệu cần di chuyển trên mạng quá cao, thì việc đụng độ có thể xảy ra với sồ lượng lớn có gây tắc nghẽn đường truyền dẫn đến làm chậm tốc độ truyền tin của hệ thống.

3. Giao thức dùng thẻ bài vòng (Token ring)

Đây là giao thức truy nhập có điều khiển chủ yếu dùng kỹ thuật chuyển thẻ bài (token) để cấp phát quyền truy nhập đường truyền tức là quyền được truyền dữ liệu đi. Thẻ bài ở đay là một đơn vị dữ liệu đặc biệt, có kích thuóc và nội dung (gồm các thông tin điều khiển) được quy định riêng cho mỗi giao thức. Theo giao thức dùng thẻ bài vòng trong đường cáp liên tục có một thẻ bài chạy quanh trong mạng Thẻ bài là một đơn vị dữ liệu đặc biệt trong đó có một bit biểu diễn trạng thái sử dụng của nó (bận hoặc rỗi). Một trạm muốn truyền dữ liệu thì phải đợi đến khi nhận được một thẻ bài rảnh. Khi đó trạm sẽ đổi bit trạng thái của thẻ bài thành bận, nép gói dữ liệu có kèm theo địa chỉ nơi nhận vào thẻ bài và truyền đi theo chiều của vòng.

Vì thẻ bài chạy vòng quang trong mạng kín và chỉ có một thẻ nên việc đụng độ dữ liệu không thể xẩy ra, do vậy hiệu suất truyền dữ liệu của mạng không thay đổi.

Trong các giao thức này cần giải quyết hai vấn đề có thể dẫn đến phá vỡ hệ thống. Một là việc mất thẻ bài làm cho trên vòng không còn thẻ bài lưu chuyển nữa. Hai là một thẻ bài bận lưu chuyển không dừng trên vòng.

4. Giao thức dung thẻ bài cho dạng đường thẳng (Token bus)

Đây là giao thức truy nhập có điều khiển trong để cấp phát quyền truy nhập đường truyền cho các trạm đang có nhu cầu truyền dữ liệu, một thẻ bài được lưu chuyển trên một vòng logic thiết lập bởi các trạm đó. Khi một trạm có thẻ bài thì nó có quyền sử dụng đường truyền trong một thời gian xác định trước. Khi đã hết dữ liệu hoặc hết thời đoạn cho phép, trạm chuyển thẻ bài đến trạm tiếp theo trong vòng logic.

ả hư vậy trong mạng phải thiết lập được vòng logic (hay còn gọi là vòng ảo) bao gồm các trạm đang hoạt động nối trong mạng được xác định vị trí theo một chuỗi thứ tự mà trạm cuối cùng của chuỗi sẽ tiếp liền sau bởi trạm đầu tiên. Mỗi trạm được biết địa chỉ của các trạm kề trước và sau nó trong đó thứ tự của các trạm trên vòng logic có thể độc lập với thứ tự vật lý. Cùng với việc thiết lập vòng thì giao thức phải luôn luôn theo dõi sự thay đổi theo trạng thái thực tế của mạng.

V. Đường cáp truyền mạng

Đường cáp truyền mạng là cơ sở hạ tầng của một hệ thống mạng, nên nó rất quan trọng và ảnh hưởng rất nhiều đến khả năng hoạt động của mạng. Hiện nay người ta thường dùng 3 loại dây cáp là cáp xoắn cặp, cáp đồng trục và cáp quang.

1. Cáp xoắn cặp

Đây là loại cáp gồm hai đường dây dẫn đồng được xoắn vào nhau nhằm làm giảm nhiễu điện từ gây ra bởi môi trường xung quanh và giữa chúng với nhau.

Hiện nay có hai loại cáp xoắn là cáp có bọc kim loại (STP - Shield Twisted Pair) và cáp không bọc kim loại (UTP -Unshield Twisted Pair).

- Cáp có bọc kim loại (STP): Lớp bọc bên ngoài có tác dụng chống nhiễu điện từ, có loại có một đôi giây xoắn vào nhau và có loại có nhiều đôi giây xoắn với nhau.

- Cáp không bọc kim loại (UTP): Tính tương tự như STP nhưng kém hơn về khả năng chống nhiễu và suy hao vì không có vỏ bọc.

STP và UTP có các loại (Category - Cat) thường dùng:

- Loại 1 & 2 (Cat 1 & Cat 2): Thường dùng cho truyền thoại và những đường truyền tốc độ thấp (nhỏ hơn 4Mb/s).

- Loại 3 (Cat 3): tốc độ truyền dữ liệu khoảng 16 Mb/s , nó là chuẩn cho hầu hết các mạng điện thoại.

- Loại 4 (Cat 4): Thích hợp cho đường truyền 20Mb/s.

- Loại 5 (Cat 5): Thích hợp cho đường truyền 100Mb/s.

- Loại 6 (Cat 6): Thích hợp cho đường truyền 300Mb/s.

Đây là loại cáp rẻ, dễ cài đặt tuy nhiên nó dễ bị ảnh hưởng của môi trường.

2. Cáp đồng trục

Cáp đồng trục có hai đường dây dẫn và chúng có cùng một trục chung, một dây dẫn trung tâm (thường là dây đồng cứng) đường dây còn lại tạo thành đường ống bao xung quanh dây dẫn trung tâm (dây dẫn này có thể là dây bện kim loại và vì nó có chức năng chống nhiễu nên còn gọi là lớp bọc kim). Giữa hai dây dẫn trên có một lớp cách ly, và bên ngoài cùng là lớp vỏ plastic để bảo vệ cáp.

Các loại cáp	*Dây xoắn cặp*	*Cáp đồng trục mỏng*	*Cáp đồng trục dày*	*Cáp quang*
Chi tiết	Bằng đồng, có 4 và 25 cặp dây (loại 3, 4, 5)	Bằng đồng, 2 dây, đường kính 5mm	Bằng đồng, 2 dây, đường kính 10mm	Thủy tinh, 2 sợi
Loại kết nối	RJ-25 hoặc 50-pin telco	Bả C	ả -series	ST
Chiều dài đoạn tối đa	100m	185m	500m	1000m
Số đầu nối tối đa trên 1 đoạn	2	30	100	2
Chạy 10 Mbit/s	Được	Được	Được	Được
Chạy 100 Mbit/s	Được	Không	Không	Được
Chống nhiễu	Tốt	Tốt	Rất tốt	Hoàn toàn
Bảo mật	Trung bình	Trung bình	Trung bình	Hoàn toàn
Độ tin cậy	Tốt	Trung bình	Tốt	Tốt
Lắp đặt	Dễ dàng	Trung bình	Khó	Khó
Khắc phục lỗi	Tốt	Dở	Dở	Tốt
Quản lý	Dễ dàng	Khó	Khó	Trung bình
Chi phí cho 1 trạm	Rất thấp	Thấp	Trung bình	Cao
Ứng dụng tốt nhất	Hệ thống Workgroup	Đường backbone	Đường backbone trong tủ mạng	Đường backbone dài trong tủ mạng hoặc các tòa nhà

Hình 5.3: Tính năng kỹ thuật của một số loại cáp mạng

Cáp đồng trục có độ suy hao ít hơn so với các loại cáp đồng khác (ví dụ như cáp xoắn đôi) do ít bị ảnh hưởng của môi trường. Các mạng cục bộ sử dụng cáp đồng trục có thể có kích thước trong phạm vi vài ngàn mét, cáp đồng trục được sử dụng nhiều trong các mạng dạng đường thẳng. Hai loại cáp thường được sử dụng là cáp đồng trục mỏng và cáp đồng trục dày trong đường kính cáp đồng trục mỏng là 0,25 inch, cáp đồng trục dày là 0,5 inch. Cả hai loại cáp đều làm việc ở cùng tốc độ nhưng cáp đồng trục mỏng có độ hao suy tín hiệu lớn hơn

Hiện nay có cáp đồng trục sau:

- RG -58,50 ohm: dùng cho mạng Thin Ethernet
- RG -59,75 ohm: dùng cho truyền hình cáp
- RG -62,93 ohm: dùng cho mạng ARCnet

Các mạng cục bộ thường sử dụng cáp đồng trục có dải thông từ 2,5 - 10 Mb/s, cáp đồng trục có độ suy hao ít hơn so với các loại cáp đồng khác vì nó có lớp vỏ bọc bên ngoài, độ dài thông thường của một đoạn cáp nối trong mạng là 200m, thường sử dụng cho dạng Bus.

3. Cáp sợi quang (Fiber - Optic Cable)

Cáp sợi quang bao gồm một dây dẫn trung tâm (là một hoặc một bó sợi thủy tinh có thể truyền dẫn tín hiệu quang) được bọc một lớp vỏ bọc có tác dụng phản xạ các tín hiệu trở lại để giảm sự mất mát tín hiệu. Bên ngoài cùng là lớp vỏ plastic để bảo vệ cáp. ẩ hư vậy cáp sợi quang không truyền dẫn các tín hiệu điện mà chỉ truyền các tín hiệu quang (các tín hiệu dữ liệu phải được chuyển đổi thành các tín hiệu quang và khi nhận chúng sẽ lại được chuyển đổi trở lại thành tín hiệu điện).

Cáp quang có đường kính từ 8.3 - 100 micron, Do đường kính lõi sợi thuỷ tinh có kích thước rất nhỏ nên rất khó khăn cho việc đấu nối, nó cần công nghệ đặc biệt với kỹ thuật cao đòi hỏi chi phí cao.

Dải thông của cáp quang có thể lên tới hàng Gbps và cho phép khoảng cách đi cáp khá xa do độ suy hao tín hiệu trên cáp rất thấp. ẩ goài ra, vì cáp sợi quang không dùng tín hiệu điện từ để truyền dữ liệu nên nó hoàn toàn không bị ảnh hưởng của nhiễu điện từ và tín hiệu truyền không thể bị phát hiện và thu trộm bởi các thiết bị điện tử của người khác.

Chỉ trừ nhược điểm khó lắp đặt và giá thành còn cao , nhìn chung cáp quang thích hợp cho mọi mạng hiện nay và sau này.

4. Các yêu cầu cho một hệ thống cáp

- An toàn, thẩm mỹ: tất cả các dây mạng phải được bao bọc cẩn thận, cách xa các nguồn điện, các máy có khả năng phát sóng để tránh trường hợp bị nhiễu. Các đầu nối phải đảm bảo chất lượng, tránh tình trạng hệ thống mạng bị chập chờn.

- Đúng chuẩn: hệ thống cáp phải thực hiện đúng chuẩn, đảm bảo cho khả năng nâng cấp sau này cũng như dễ dàng cho việc kết nối các thiết bị khác nhau của các nhà sản xuất khác nhau. Tiêu chuẩn quốc tế dùng cho các hệ thống mạng hiện nay là EIA/TIA 568B.

- Tiết kiệm và "linh hoạt" (flexible): hệ thống cáp phải được thiết kế sao cho kinh tế nhất, dễ dàng trong việc di chuyển các trạm làm việc và có khả năng mở rộng sau này.

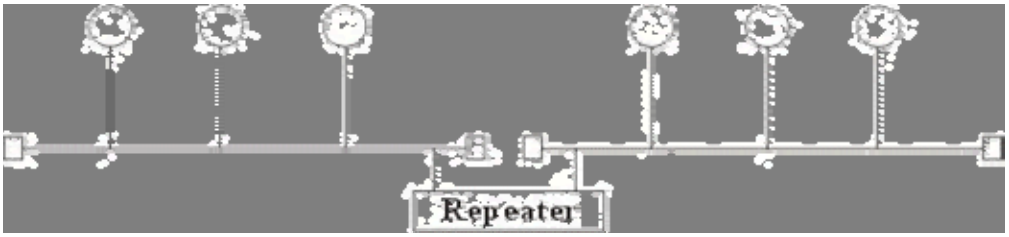
www.viet-ebook.co.cc

Chương 6

Các thiết bị liên kết mạng

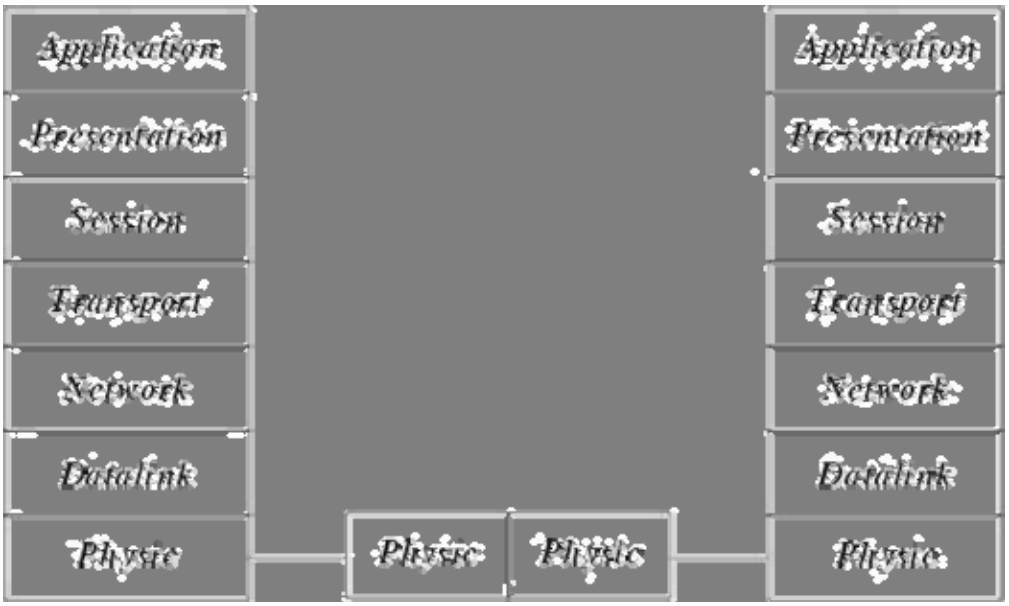
I. Repeater (Bộ tiếp sức)

Repeater là loại thiết bị phần cứng đơn giản nhất trong các thiết bị liên kết mạng, nó được hoạt động trong tầng vật lý của mô hình hệ thống mở OSI. Repeater dùng để nối 2 mạng giống nhau hoặc các phần một mạng cùng có một nghi thức và một cấu hình. Khi Repeater nhận được một tín hiệu từ một phía của mạng thì nó sẽ phát tiếp vào phía kia của mạng.



Hình 6.1: Mô hình liên kết mạng của Repeater.

Repeater không có xử lý tín hiệu mà nó chỉ loại bỏ các tín hiệu méo, nhiễu, khuếch đại tín hiệu đã bị suy hao (vì đã được phát với khoảng cách xa) và khôi phục lại tín hiệu ban đầu. Việc sử dụng Repeater đã làm tăng thêm chiều dài của mạng.



Hình 6.2: Hoạt động của bộ tiếp sức trong mô hình OSI

Hiện nay có hai loại Repeater đang được sử dụng là Repeater điện và Repeater điện quang.

- ***Repeater điện*** nối với đường dây điện ở cả hai phía của nó, nó nhận tín hiệu điện từ một phía và phát lại về phía kia. Khi một mạng sử dụng Repeater điện để nối các phần của mạng lại thì có thể làm tăng khoảng cách của mạng, nhưng khoảng cách đó luôn bị hạn chế bởi một khoảng cách tối đa do độ trễ của tín hiệu. Ví dụ với

mạng sử dụng cáp đồng trục 50 thì khoảng cách tối đa là 2.8 km, khoảng cách đó không thể kéo thêm cho dù sử dụng thêm Repeater.

- ***Repeater điện quang*** liên kết với một đầu cáp quang và một đầu là cáp điện, nó chuyển một tín hiệu điện từ cáp điện ra tín hiệu quang để phát trên cáp quang và ngược lại. Việc sử dụng Repeater điện quang cũng làm tăng thêm chiều dài của mạng.

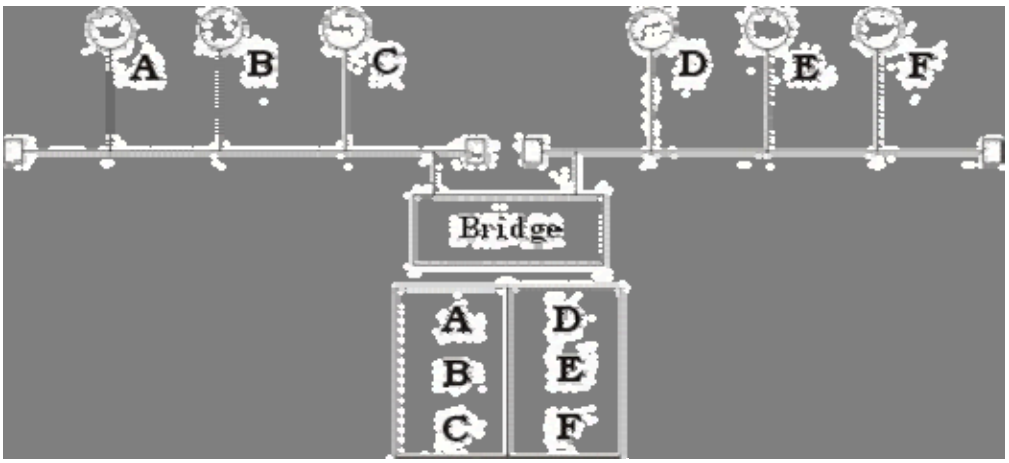
Việc sử dụng Repeater không thay đổi nội dung các tín hiện đi qua nên nó chỉ được dùng để nối hai mạng có cùng giao thức truyền thông (như hai mạng Ethernet hay hai mạng Token ring) nhưng không thể nối hai mạng có giao thức truyền thông khác nhau (như một mạng Ethernet và một mạng Token ring). Thêm nữa Repeater không làm thay đổi khối lượng chuyển vận trên mạng nên việc sử dụng không tính toán nó trên mạng lớn sẽ hạn chế hiệu năng của mạng. Khi lựa chọn sử dụng Repeater cần chú ý lựa chọn loại có tốc độ chuyển vận phù hợp với tốc độ của mạng.

II. Bridge (Cầu nối)

Bridge là một thiết bị có xử lý dùng để nối hai mạng giống nhau hoặc khác nhau, nó có thể được dùng với các mạng có các giao thức khác nhau. Cầu nối hoạt động trên tầng liên kết dữ liệu nên không như bộ tiếp sức phải phát lại tất cả những gì nó nhận được thì cầu nối đọc được các gói tin của tầng liên kết dữ liệu trong mô hình OSI và xử lý chúng trước khi quyết định có chuyển đi hay không.

Khi nhận được các gói tin Bridge chọn lọc và chỉ chuyển những gói tin mà nó thấy cần thiết. Điều này làm cho Bridge trở nên có ích khi nối một vài mạng với nhau và cho phép nó hoạt động một cách mềm dẻo.

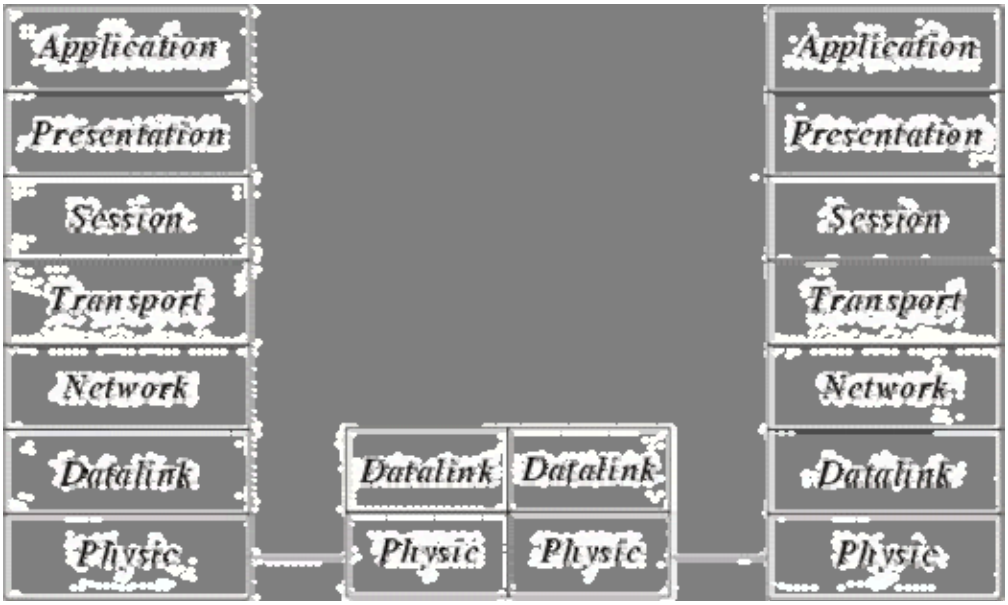
Để thực hiện được điều này trong Bridge ở mỗi đầu kết nối có một bảng các địa chỉ các trạm được kết nối vào phía đó, khi hoạt động cầu nối xem xét mỗi gói tin nó nhận được bằng cách đọc địa chỉ của nơi gửi và nhận và dựa trên bảng địa chỉ phía nhận được gói tin nó quyết định gửi gói tin hay không và bổ xung bảng địa chỉ.



Hình 6.3: Hoạt động của Bridge

Khi đọc địa chỉ nơi gửi Bridge kiểm tra xem trong bảng địa chỉ của phần mạng nhận được gói tin có địa chỉ đó hay không, nếu không có thì Bridge tự động bổ xung bảng địa chỉ (cơ chế đó được gọi là tự học của cầu nối).

Khi đọc địa chỉ nơi nhận Bridge kiểm tra xem trong bảng địa chỉ của phần mạng nhận được gói tin có địa chỉ đó hay không, nếu có thì Bridge sẽ cho rằng đó là gói tin nội bộ thuộc phần mạng mà gói tin đến nên không chuyển gói tin đó đi, nếu ngược lại thì Bridge mới chuyển sang phía bên kia. Ở đây chúng ta thấy một trạm không cần thiết chuyển thông tin trên toàn mạng mà chỉ trên phần mạng có trạm nhận mà thôi.



Hình 6.4: Hoạt động của Bridge trong mô hình OSI

Để đánh giá một Bridge người ta đưa ra hai khái niệm : Lọc và chuyển vận. Quá trình xử lý mỗi gói tin được gọi là quá trình lọc trong đó tốc độ lọc thể hiện trực tiếp khả năng hoạt động của Bridge. Tốc độ chuyển vận được thể hiện số gói tin/giây trong đó thể hiện khả năng của Bridge chuyển các gói tin từ mạng này sang mạng khác.

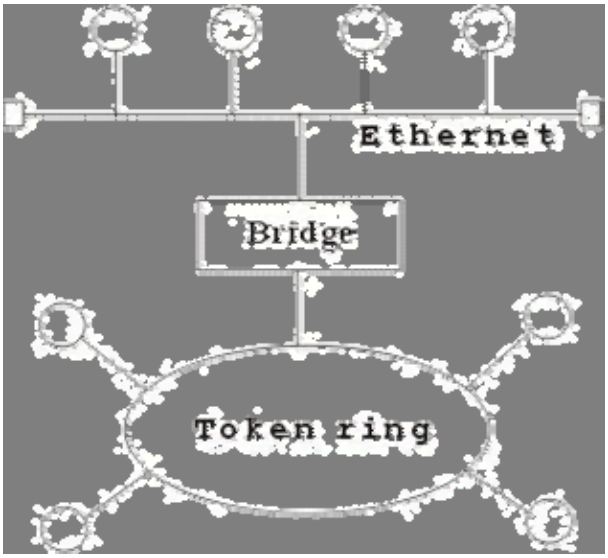
Hiện nay có hai loại Bridge đang được sử dụng là Bridge vận chuyển và Bridge biên dịch. Bridge *vận chuyển* dùng để nối hai mạng cục bộ cùng sử dụng một giao thức truyền thông của tầng liên kết dữ liệu, tuy nhiên mỗi mạng có thể sử dụng loại dây nối khác nhau. Bridge vận chuyển không có khả năng thay đổi cấu trúc các gói tin mà nó nhận được mà chỉ quan tâm tới việc xem xét và chuyển vận gói tin đó đi.

Bridge biên dịch dùng để nối hai mạng cục bộ có giao thức khác nhau nó có khả năng chuyển một gói tin thuộc mạng này sang gói tin thuộc mạng kia trước khi chuyển qua

Ví dụ : Bridge biên dịch nối một mạng Ethernet và một mạng Token ring. Khi đó Cầu nối thực hiện như một nút token ring trên mạng Token ring và một nút Enthernet trên mạng Ethernet. Cầu nối có thể chuyển một gói tin theo chuẩn đang sử dụng trên mạng Enthernet sang chuẩn đang sử dụng trên mạng Token ring.

Tuy nhiên chú ý ở đây cầu nối không thể chia một gói tin ra làm nhiều gói tin cho nên phải hạn chế kích thước tối đa các gói tin phù hợp với cả hai mạng. Ví dụ như kích thước tối đa

của gói tin trên mạng Ethernet là 1500 bytes và trên mạng Token ring là 6000 bytes do vậy nếu một trạm trên mạng token ring gửi một gói tin cho trạm trên mạng Ethernet với kích thước lớn hơn 1500 bytes thì khi qua cầu nối số lượng byte dư sẽ bị chặt bỏ.



Hình 6.5: Ví dụ về Bridge biên dịch

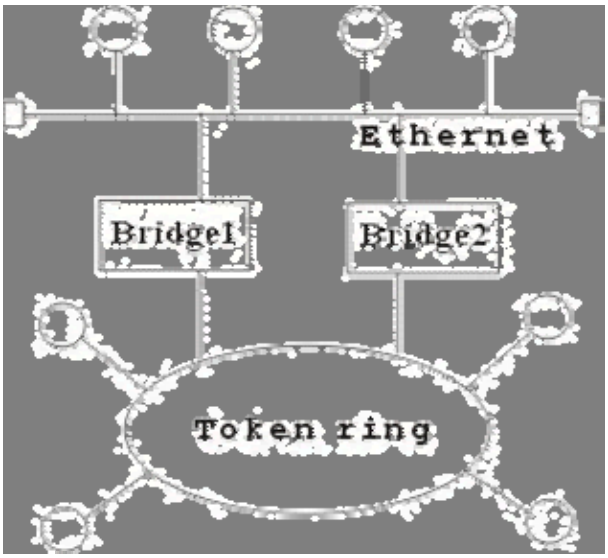
ầ gười ta sử dụng Bridge trong các trường hợp sau :

- Mở rộng mạng hiện tại khi đã đạt tới khoảng cách tối đa do Bridge sau khi sử lý gói tin đã phát lại gói tin trên phần mạng còn lại nên tín hiệu tốt hơn bộ tiếp sức.

- Giảm bớt tắc nghẽn mạng khi có quá nhiều trạm bằng cách sử dụng Bridge, khi đó chúng ta chia mạng ra thành nhiều phần bằng các Bridge, các gói tin trong nội bộ tùng phần mạng sẽ không được phép qua phần mạng khác.

- Để nối các mạng có giao thức khác nhau.

Một vài Bridge còn có khả năng lựa chọn đối tượng vận chuyển. ầ ó có thể chỉ chuyển vận những gói tin của nhưng địa chỉ xác định. Ví dụ : cho phép gói tin của máy A, B qua Bridge 1, gói tin của máy C, D qua Bridge 2.

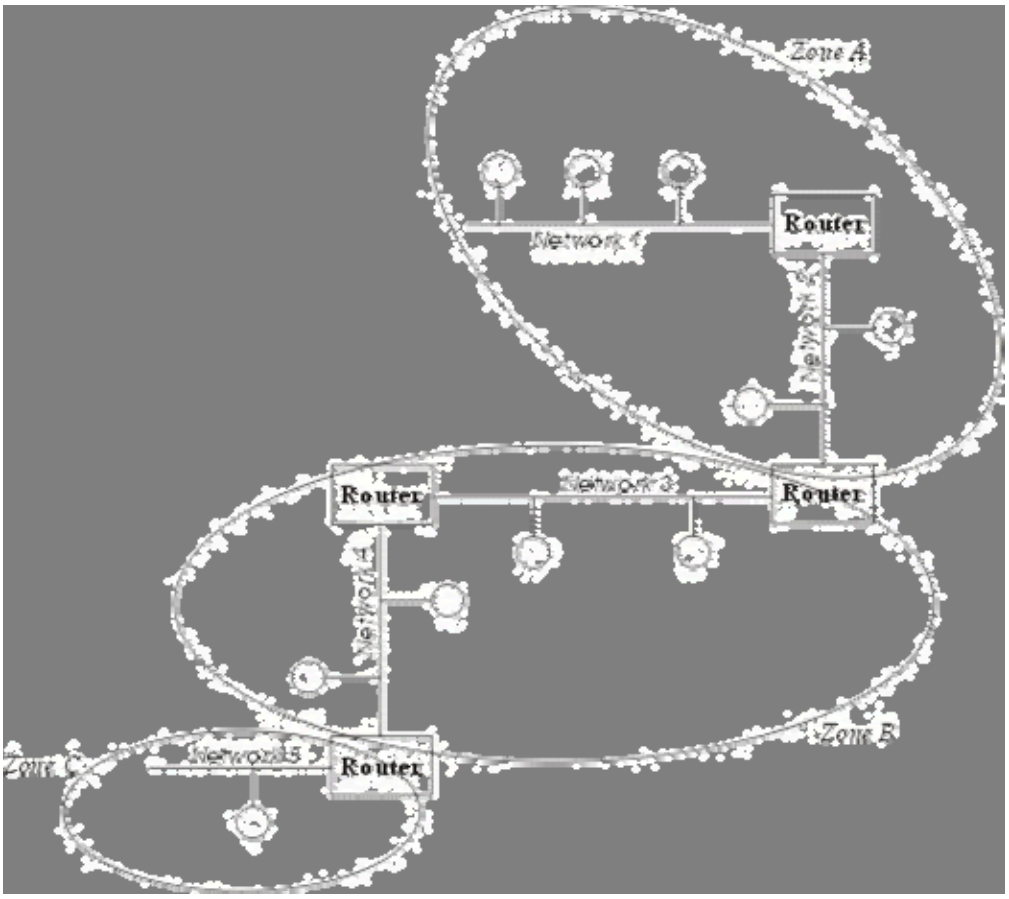


Hình 6.6 : Liên kết mạng với 2 Bridge

Một số Bridge được chế tạo thành một bộ riêng biệt, chỉ cần nối dây và bật. Các Bridge khác chế tạo như card chuyên dùng cắïm vào máy tính, khi đó trên máy tính sẽ sử dụng phần mềm Bridge. Việc kết hợp phần mềm với phần cứng cho phép uyển chuyển hơn trong hoạt động của Bridge.

III. Router (Bộ tìm đường)

Router là một thiết bị hoạt động trên tầng mạng, nó có thể tìm được đường đi tốt nhất cho các gói tin qua nhiều kết nối để đi từ trạm gửi thuộc mạng đầu đến trạm nhận thuộc mạng cuối. Router có thể được sử dụng trong việc nối nhiều mạng với nhau và cho phép các gói tin có thể đi theo nhiều đường khác nhau để tới đích.



Hình 6.7: Hoạt động của Router.

Khác với Bridge hoạt động trên tầng liên kết dữ liệu nên Bridge phải xử lý mọi gói tin trên đường truyền thì Router có địa chỉ riêng biệt và nó chỉ tiếp nhận và xử lý các gói tin gửi đến nó mà thôi. Khi một trạm muốn gửi gói tin qua Router thì nó phải gửi gói tin với địa chỉ trực tiếp của Router (Trong gói tin đó phải chứa các thông tin khác về đích đến) và khi gói tin đến Router thì Router mới xử lý và gửi tiếp.

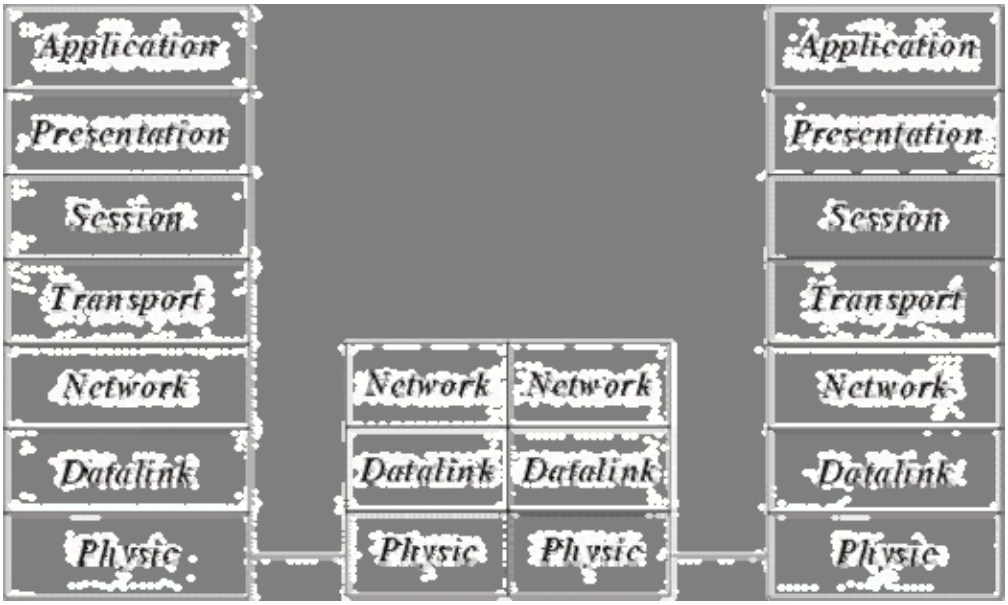
Khi xử lý một gói tin Router phải tìm được đường đi của gói tin qua mạng. Để làm được điều đó Router phải tìm được đường đi tốt nhất trong mạng dựa trên các thông tin nó có về mạng, thông thường trên mỗi Router có một bảng chỉ đường (Router table). Dựa trên dữ liệu về Router gần đó và các mạng trong liên mạng, Router tính được bảng chỉ đường (Router table) tối ưu dựa trên một thuật toán xác định trước.

ả gười ta phân chia Router thành hai loại là Router có phụ thuộc giao thức (The protocol dependent routers) và Router không phụ thuộc vào giao thức (The protocol independent router) dựa vào phương thức xử lý các gói tin khi qua Router.

- *Router có phụ thuộc giao thức:* Chỉ thực hiện việc tìm đường và truyền gói tin từ mạng này sang mạng khác chứ không chuyển đổi phương cách đóng gói của gói tin cho nên cả hai mạng phải dùng chung một giao thức truyền thông.

- *Router không phụ thuộc vào giao thức*: có thể liên kết các mạng dùng giao thức truyền thông khác nhau và có thể chuyển đôiø gói tin của giao thức này sang gói tin

của giao thức kia, Router cũng ù chấp nhận kích thước các gói tin khác nhau (Router có thể chia nhỏ một gói tin lớn thành nhiều gói tin nhỏ trước truyền trên mạng).



Hình 6.8: Hoạt động của Router trong mô hình OSI

Để ngăn chặn việc mất mát số liệu Router còn nhận biết được đường nào có thể chuyển vận và ngừng chuyển vận khi đường bị tắc.

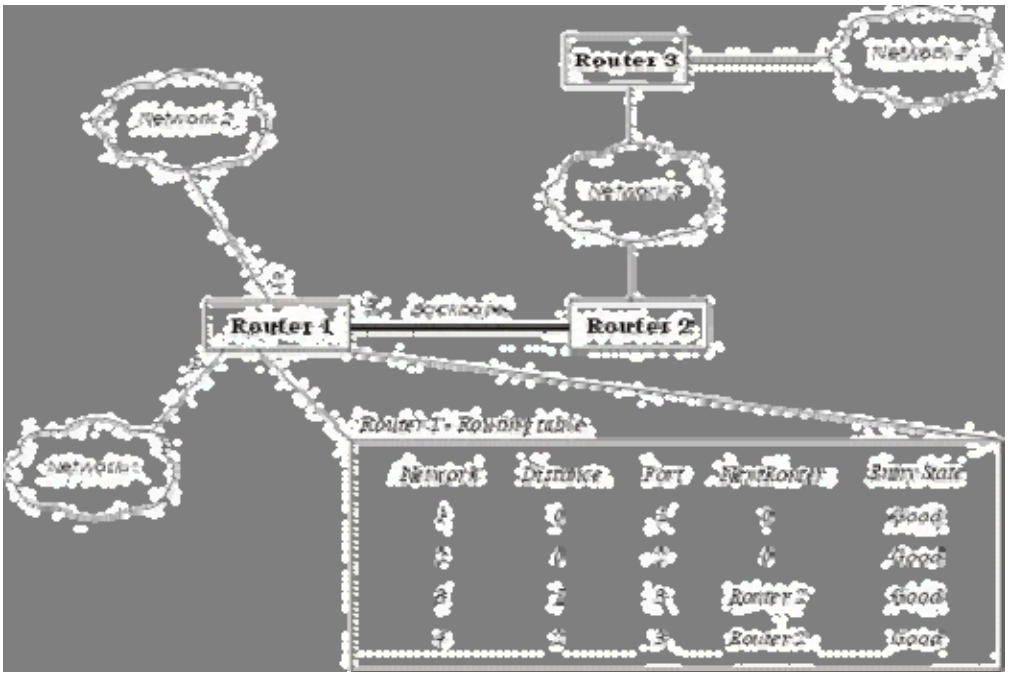
Các lý do sử dụng Router :

- Router có các phần mềm lọc ưu việt hơn là Bridge do các gói tin muốn đi qua Router cần phải gửi trực tiếp đến nó nên giảm được số lượng gói tin qua nó. Router thường được sử dụng trong khi nối các mạng thông qua các đường dây thuê bao đắt tiền do nó không truyền dư lên đường truyền.

- Router có thể dùng trong một liên mạng có nhiều vùng, mỗi vùng có giao thức riêng biệt.

- Router có thể xác định được đường đi an toàn và tốt nhất trong mạng nên độ an toàn của thông tin được đảm bảo hơn.

- Trong một mạng phức hợp khi các gói tin luân chuyển các đường có thể gây nên tình trạng tắc nghẽn của mạng thì các Router có thể được cài đặt các phương thức nhằm tránh được tắc nghẽn.



Hình 6.9: Ví dụ về bảng chỉ đường (Routing table) của Router.

Các phương thức hoạt động của Router

Đó là phương thức mà một Router có thể nối với các Router khác để qua đó chia sẻ thông tin về mạng hiện co. Các chương trình chạy trên Router luôn xây dựng bảng chỉ đường qua việc trao đổi các thông tin với các Router khác.

- Phương thức véc tơ khoảng cách : mỗi Router luôn luôn truyền đi thông tin về bảng chỉ đường của mình trên mạng, thông qua đó các Router khác sẽ cập nhật lên bảng chỉ đường của mình.

- Phương thức trạng thái tĩnh : Router chỉ truyền các thông báo khi có phát hiện có sự thay đổi trong mạng vàchỉ khi đó các Routerkhác ù cập nhật lại bảng chỉ đường, thông tin truyền đi khi đó thường là thông tin về đường truyền.

Một số giao thức hoạt động chính của Router

- *RIP(Routing Information Protocol)* được phát triển bởi Xerox ả etwork system và sử dụng SPX/IPX và TCP/IP. RIP hoạt động theo phương thức véc tơ khoảng cách.

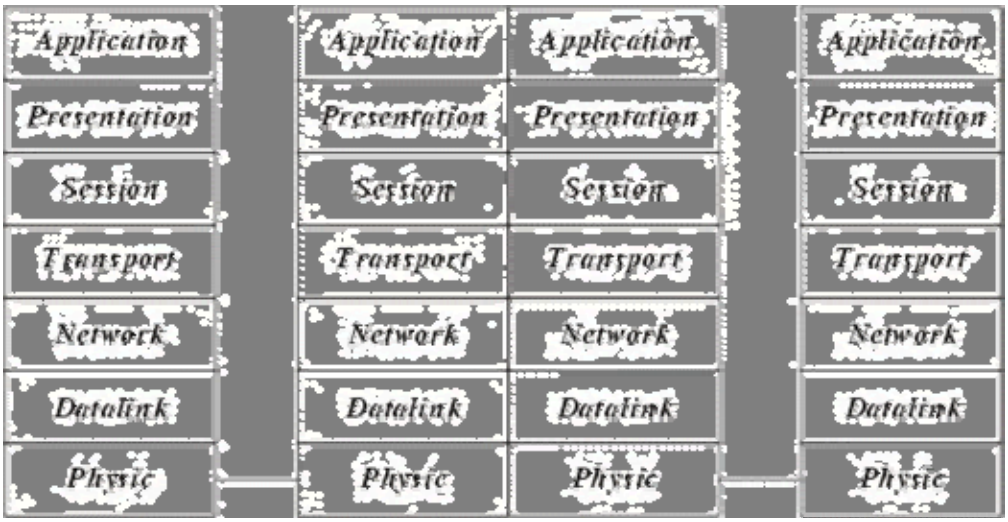
- *NLSP (Netware Link Service Protocol)* được phát triển bởi ả ovell dùng để thay thế RIP hoạt động theo phương thức véctơ khoảng cách, mỗi Router được biết cấu trúc của mạng và việc truyền các bảng chỉ đường giảm đi..

- *OSPF (Open Shortest Path First)* là một phần của TCP/IP với phương thức trạng thái tĩnh, trong đó có xét tới ưu tiên, giá đường truyền, mật độ truyền thông...

- *OSPF-IS (Open System Interconnection Intermediate System to Intermediate System)* là một phần của TCP/IP với phương thức trạng thái tĩnh, trong đó có xét tới ưu tiên, giá đường truyền, mật độ truyền thông...

IV. Gateway (cổng nối)

Gateway dùng để kết nối các mạng không thuần nhất chẳng hạn như các mạng cục bộ và các mạng máy tính lớn (Mainframe), do các mạng hoàn toàn không thuần nhất nên việc chuyển đổi thực hiện trên cả 7 tầng của hệ thống mở OSI. Thường được sử dụng nối các mạng LAå vào máy tính lớn. Gateway có các giao thức xác định trước thường là nhiều giao thức, một Gateway đa giao thức thường được chế tạo như các Card có chứa các bộ xử lý riêng và cài đặt trên các máy tính hoặc thiết bị chuyên biệt.



Hình 6.10: Hoạt động của Gateway trong mô hình OSI

Hoạt động của Gateway thông thường phức tạp hơn là Router nên thông suất của nó thường chậm hơn và thường không dùng nối mạng LAå -LAå .

V. Hub (Bộ tập trung)

Hub thường được dùng để nối mạng, thông qua những đầu cắm của nó người ta liên kết với các máy tính dưới dạng hình sao.

å gười ta phân biệt các Hub thành 3 loại như sau sau :

- ***Hub bị động (Passive Hub) :*** Hub bị động không chứa các linh kiện điện tử và cũng không xử lý các tín hiệu dữ liệu, nó có chức năng duy nhất là tổ hợp các tín hiệu từ một số đoạn cáp mạng. Khoảng cách giữa một máy tính và Hub không thể lớn hơn một nửa khoảng cách tối đa cho phép giữa 2 máy tính trên mạng (ví dụ khoảng cách tối đa cho phép giữa 2 máy tính của mạng là 200m thì khoảng cách tối đa giữa một máy tính và hub là 100m). Các mạng ARCnet thường dùng Hub bị động.

- ***Hub chủ động (Active Hub) :*** Hub chủ động có các linh kiện điện tử có thể khuyếch đại và xử lý các tín hiệu điện tử truyền giữa các thiết bị của mạng. Qúa trình xử lý tín hiệu được gọi là tái sinh tín hiệu, nó làm cho tín hiệu trở nên tốt hơn, ít nhạy cảm với lỗi do vậy khoảng cách giữa các thiết bị có thể tăng lên. Tuy nhiên những ưu điểm đó cũng kéo theo giá thành của Hub chủ động cao hơn nhiều so với Hub bị động. Các mạng Token ring có xu hướng dùng Hub chủ động.

www.viet-ebook.co.cc

- ***Hub thông minh (Intelligent Hub):*** cũng là Hub chủ động nhưng có thêm các chức năng mới so với loại trước, nó có thể có bộ vi xử lý của mình và bộ nhớ mà qua đó nó không chỉ cho phép điều khiển hoạt động thông qua các chương trình quản trị mạng mà nó có thể hoạt động như bộ tìm đường hay một cầu nối. ả ó có thể cho phép tìm đường cho gói tin rất nhanh trên các cổng của nó, thay vì phát lại gói tin trên mọi cổng thì nó có thể chuyển mạch để phát trên một cổng có thể nối tới trạm đích.

www.viet-ebook.co.cc

Chương 7

Giao thức TCP/IP

Giao thức TCP/IP được phát triển từ mạng ARPAả ET và Internet và được dùng như giao thức mạng và vận chuyển trên mạng Internet. TCP (Transmission Control Protocol) là giao thức thuộc tầng vận chuyển và IP (Internet Protocol) là giao thức thuộc tầng mạng của mô hình OSI. Họ giao thức TCP/IP hiện nay là giao thức được sử dụng rộng rãi nhất để liên kết các máy tính và các mạng.

Hiện nay các máy tính của hầu hết các mạng có thể sử dụng giao thức TCP/IP để liên kết với nhau thông qua nhiều hệ thống mạng với kỹ thuật khác nhau. Giao thức TCP/IP thực chất là một họ giao thức cho phép các hệ thống mạng cùng làm việc với nhau thông qua việc cung cấp phương tiện truyền thông liên mạng.

I. Giao thức IP

1. Tổng quát

ả hiệm vụ chính của giao thức IP là cung cấp khả năng kết nối các mạng con thành liên kết mạng để truyền dữ liệu, vai trò của IP là vai trò của giao thức tầng mạng trong mô hình OSI. Giao thức IP là một giao thức kiểu không liên kết (connectionlees) có nghĩa là không cần có giai đoạn thiết lập liên kết trước khi truyền dữ liệu.

Sơ đồ địa chỉ hóa để định danh các trạm (host) trong liên mạng được gọi là địa chỉ IP 32 bits (32 bit IP address). Mỗi giao diện trong 1 máy có hỗ trợ giao thức IP đều phải được gán 1 địa chỉ IP (một máy tính có thể gắn với nhiều mạng do vậy có thể có nhiều địa chỉ IP). Địa chỉ IP gồm 2 phần: địa chỉ mạng (netid) và địa chỉ máy (hostid). Mỗi địa chỉ IP có độ dài 32 bits được tách thành 4 vùng (mỗi vùng 1 byte), có thể biểu thị dưới dạng thập phân, bát phân, thập lục phân hay nhị phân. Cách viết phổ biến nhất là dùng ký pháp thập phân có dấu chấm (dotted decimal notation) để tách các vùng. Mục đích của địa chỉ IP là để định danh duy nhất cho một máy tính bất kỳ trên liên mạng.

Do tổ chức và độ lớn của các mạng con (subnet) của liên mạng có thể khác nhau, người ta chia các địa chỉ IP thành 5 lớp, ký hiệu là A, B, C, D và E. Trong lớp A, B, C chứa địa chỉ có thể gán được. Lớp D dành riêng cho lớp kỹ thuật multicasting. Lớp E được dành những ứng dụng trong tương lai.

ả etid trong địa chỉ mạng dùng để nhận dạng từng mạng riêng biệt. Các mạng liên kết phải có địa chỉ mạng (netid) riêng cho mỗi mạng. Ở đây các bit đầu tiên của byte đầu tiên được dùng để định danh lớp địa chỉ (0 - lớp A, 10 - lớp B, 110 - lớp C, 1110 - lớp D và 11110 - lớp E).

Ơû đây ta xét cấu trúc của các lớp địa chỉ có thể gán được là lớp A, lớp B, lớp C

Cấu trúc của các địa chỉ IP như sau:

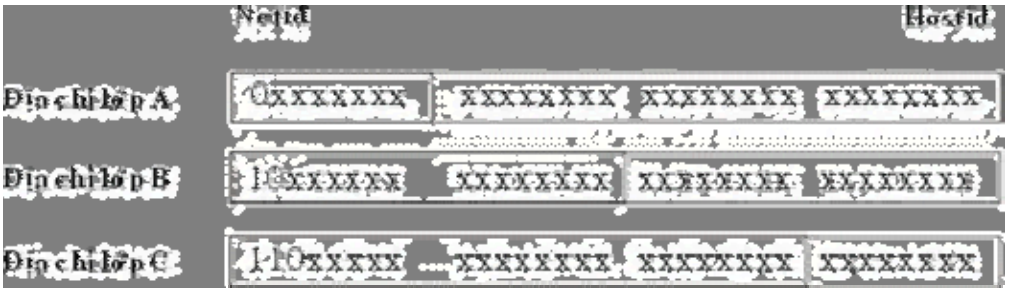
- Mạng lớp A: địa chỉ mạng (netid) là 1 Byte và địa chỉ host (hostid) là 3 byte.

- Mạng lớp B: địa chỉ mạng (netid) là 2 Byte và địa chỉ host (hostid) là 2 byte.
- Mạng lớp C: địa chỉ mạng (netid) là 3 Byte và địa chỉ host (hostid) là 1 byte.

Lớp A cho phép định danh tới 126 mạng, với tối đa 16 triệu host trên mỗi mạng. Lớp này được dùng cho các mạng có số trạm cực lớn.

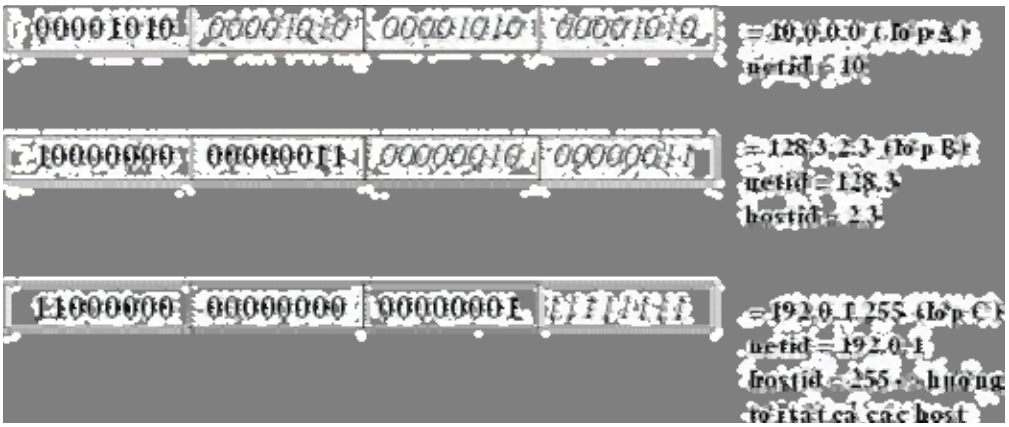
Lớp B cho phép định danh tới 16384 mạng, với tối đa 65534 host trên mỗi mạng.

Lớp C cho phép định danh tới 2 triệu mạng, với tối đa 254 host trên mỗi mạng. Lớp này được dùng cho các mạng có ít trạm.



Hình 7.1: Cấu trúc các lớp địa chỉ IP

Một số địa chỉ có tính chất đặc biệt: Một địa chỉ có hostid = 0 được dùng để hướng tới mạng định danh bởi vùng netid. ả gược lại, một địa chỉ có vùng hostid gồm toàn số 1 được dùng để hướng tới tất cả các host nối vào mạng netid, và nếu vùng netid cũng gồm toàn số 1 thì nó hướng tới tất cả các host trong liên mạng

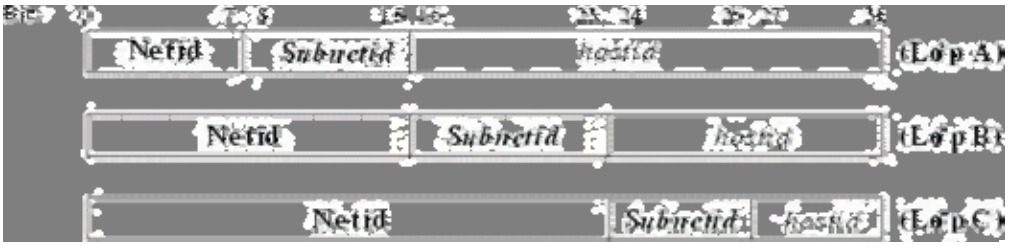


Hình 7.2: Ví dụ cấu trúc các lớp địa chỉ IP

Cần lưu ý rằng các địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, và chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm trên đó một mạng cục bộ (Ethernet, Token Ring.).

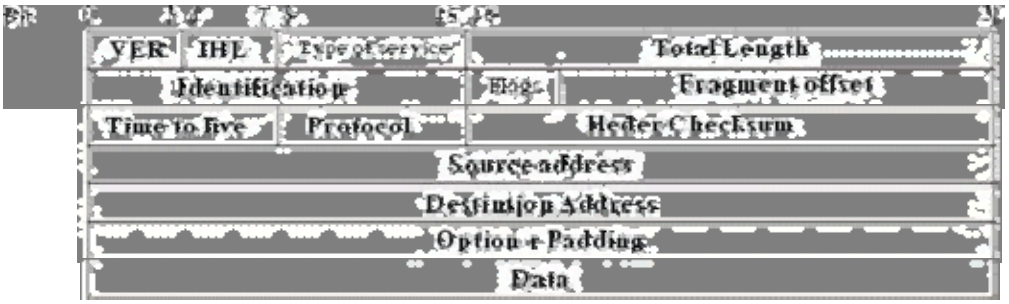
Trong nhiều trường hợp, một mạng có thể được chia thành nhiều mạng con (subnet), lúc đó có thể đưa thêm các vùng subnetid để định danh các mạng con. Vùng subnetid được lấy từ vùng hostid, cụ thể đối với lớp A, B, C như ví dụ sau:

www.viet-ebook.co.cc



Hình 7.3: Ví dụ địa chỉ khi bổ sung vùng subnetid

Đơn vị dữ liệu dùng trong IP được gọi là gói tin (datagram), có khuôn dạng



Hình 7.4: Dạng thức của gói tin IP

Ý nghĩa của thông số như sau:

- *VER* (4 bits): chỉ version hiện hành của giao thức IP hiện được cài đặt, Việc có chỉ số version cho phép có các trao đổi giữa các hệ thống sử dụng version cũ và hệ thống sử dụng version mới.

- *IHL* (4 bits): chỉ độ dài phần đầu (Internet header Length) của gói tin datagram, tính theo đơn vị từ (32 bits). Trường này bắt buột phải có vì phần đầu IP có thể có độ dài thay đổi tùy ý. Độ dài tối thiểu là 5 từ (20 bytes), độ dài tối đa là 15 từ hay là 60 bytes.

- *Type of service* (8 bits): đặc tả các tham số về dịch vụ nhằm thông báo cho mạng biết dịch vụ nào mà gói tin muốn được sử dụng, chẳng hạn ưu tiên, thời hạn chậm trễ, năng suất truyền và độ tin cậy. Hình sau cho biết ý nghĩ của trường 8 bits này.

- Precedence (3 bit): chỉ thị về quyền ưu tiên gửi datagram, nó có giá trị từ 0 (gói tin bình thường) đến 7 (gói tin kiểm soát mạng).

- D (Delay) (1 bit): chỉ độ trễ yêu cầu trong đó

  - D = 0 gói tin có độ trễ bình thường

  - D = 1 gói tin độ trễ thấp

- T (Throughput) (1 bit): chỉ độ thông lượng yêu cầu sử dụng để truyền gói tin với lựa chọn truyền trên đường thông suất thấp hay đường thông suất cao.
    - T = 0 thông lượng bình thường và
    - T = 1 thông lượng cao
- R (Reliability) (1 bit): chỉ độ tin cậy yêu cầu
    - R = 0 độ tin cậy bình thường
    - R = 1 độ tin cậy cao
- *Total Length* (16 bits): chỉ độ dài toàn bộ gói tin, kể cả phần đầu tính theo đơn vị byte với chiều dài tối đa là 65535 bytes. Hiện nay giới hạn trên là rất lớn nhưng trong tương lai với những mạng Gigabit thì các gói tin có kích thước lớn là cần thiết.
- *Identification* (16 bits): cùng với các tham số khác (như Source Address và Destination Address) tham số này dùng để định danh duy nhất cho một datagram trong khoảng thời gian nó vẫn còn trên liên mạng.
- *Flags (3 bits):* liên quan đến sự phân đoạn (fragment) các datagram, Các gói tin khi đi trên đường đi có thể bị phân thành nhiều gói tin nhỏ, trong trường hợp bị phân đoạn thì trường Flags được dùng điều khiển phân đoạn và tái lắp ghép bó dữ liệu. Tùy theo giá trị của Flags sẽ có ý nghĩa là gói tin sẽ không phân đoạn, có thể phân đoạn hay là gói tin phân đoạn cuối cùng. Trường ***Fragment Offset*** cho biết vị trí dữ liệu thuộc phân đoạn tương ứng với đoạn bắt đầu của gói dữ liệu gốc. Ý nghĩa cụ thể của trường Flags là:

    - bit 0: reserved - chưa sử dụng, luôn lấy giá trị 0.
    - bit 1: (DF) = 0 (May Fragment) = 1 (Don't Fragment)
    - bit 2: (MF) = 0 (Last Fragment) = 1 (More Fragments)
- *Fragment Offset (13 bits)*: chỉ vị trí của đoạn (fragment) ở trong datagram tính theo đơn vị 8 bytes, có nghĩa là phần dữ liệu mỗi gói tin (trừ gói tin cuối cùng) phải chứa một vùng dữ liệu có độ dài là bội số của 8 bytes. Điều này có ý nghĩa là phải nhân giá trị của Fragment offset với 8 để tính ra độ lệch byte.
- *Time to Live (8 bits)*: qui định thời gian tồn tại (tính bằng giây) của gói tin trong mạng để tránh tình trạng một gói tin bị quẩn trên mạng. Thời gian này được cho bởi trạm gửi và được giảm đi (thường qui ước là 1 đơn vị) khi datagram đi qua mỗi router của liên mạng. Thời lượng này giảm xuống tại mỗi router với mục đích giới hạn thời gian tồn tại của các gói tin và kết thúc những lần lặp lại vô hạn trên mạng. Sau đây là 1 số điều cần lưu ý về trường ***Time To Live***:

- ả út trung gian của mạng không được gởi 1 gói tin mà trường này có giá trị= 0.

- Một giao thức có thể ấn định ***Time To Live*** để thực hiện cuộc ra tìm tài nguyên trên mạng trong phạm vi mở rộng.

- Một giá trị cố định tối thiểu phải đủ lớn cho mạng hoạt động tốt.

- *Protocol (8 bits):* chỉ giao thức tầng trên kế tiếp sẽ nhận vùng dữ liệu ở trạm đích (hiện tại thường là TCP hoặc UDP được cài đặt trên IP). Ví dụ: **TCP** có giá trị trường **Protocol** là 6, **UDP** có giá trị trường **Protocol** là 17

- *Header Checksum (16 bits)*: Mã kiểm soát lỗi của header gói tin IP.

- *Source Address (32 bits)*: Địa chỉ của máy nguồn.

- *Destination Address (32 bits)*: địa chỉ của máy đích

- *Options (độ dài thay đổi)*: khai báo các lựa chọn do người gửi yêu cầu (tuỳ theo từng chương trình).

- *Padding (độ dài thay đổi)*: Vùng đệm, được dùng để đảm bảo cho phần header luôn kết thúc ở một mốc 32 bits.

- *Data (độ dài thay đổi)*: Trên một mạng cục bộ như vậy, hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. ả hư vậy vấn đề đặt ra là phải thực hiện ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý (48 bits) của một trạm.

2. Các giao thức trong mạng IP

Để mạng với giao thức IP hoạt động được tốt người ta cần một số giao thức bổ sung, các giao thức này đều không phải là bộ phận của giao thức IP và giao thức IP sẽ dùng đến chúng khi cần.

- *Giao thức ARP (Address Resolution Protocol):* Ở đây cần lưu ý rằng các địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, và chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm trên đó một mạng cục bộ (Ethernet, Token Ring.). Trên một mạng cục bộ hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. ả hư vậy vấn đề đặt ra là phải tìm được ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý của một trạm. *Giao thức ARP* đã được xây dựng để tìm địa chỉ vật lý từ địa chỉ IP khi cần thiết.

- *Giao thức RARP (Reverse Address Resolution Protocol)*: Là giao thức ngược với *giao thức ARP*. Giao thức RARP được dùng để tìm địa chỉ IP từ địa chỉ vật lý.

- *Giao thức ICMP (Internet Control Message Protocol):* Giao thức này thực hiện truyền các thông báo điều khiển (báo cáo về các tình trạng các lỗi trên mạng.) giữa các gateway hoặc một nút của liên mạng. Tình trạng lỗi có thể là: một gói tin IP không thể tới đích của nó, hoặc một router không đủ bộ nhớ đệm để lưu và chuyển một gói tin IP, Một thông báo ICMP được tạo và chuyển cho IP. IP sẽ "bọc"

(encapsulate) thông báo đó với một IP header và truyền đến cho router hoặc trạm đích.

3. Các bước hoạt động của giao thức IP

Khi giao thức IP được khởi động nó trở thành một thực thể tồn tại trong máy tính và bắt đầu thực hiện những chức năng của mình, lúc đó thực thể IP là cấu thành của tầng mạng, nhận yêu cầu từ các tầng trên nó và gửi yêu cầu xuống các tầng dưới nó.

Đối với thực thể IP ở máy nguồn, khi nhận được một yêu cầu gửi từ tầng trên, nó thực hiện các bước sau đây:

- Tạo một IP datagram dựa trên tham số nhận được.
- Tính checksum và ghép vào header của gói tin.
- Ra quyết định chọn đường: hoặc là trạm đích nằm trên cùng mạng hoặc một gateway sẽ được chọn cho chặng tiếp theo.
- Chuyển gói tin xuống tầng dưới để truyền qua mạng.

Đối với router, khi nhận được một gói tin đi qua, nó thực hiện các động tác sau:

1) Tính chesksum, nếu sai thì loại bỏ gói tin.

2) Giảm giá trị tham số Time - to Live. nếu thời gian đã hết thì loại bỏ gói tin.

3) Ra quyết định chọn đường.

4) Phân đoạn gói tin, nếu cần.

5) Kiến tạo lại IP header, bao gồm giá trị mới của các vùng Time - to -Live, Fragmentation và Checksum.

6) Chuyển datagram xuống tầng dưới để chuyển qua mạng.

Cuối cùng khi một datagram nhận bởi một thực thể IP ở trạm đích, nó sẽ thực hiện bởi các công việc sau:

1) Tính checksum. ả ếu sai thì loại bỏ gói tin.

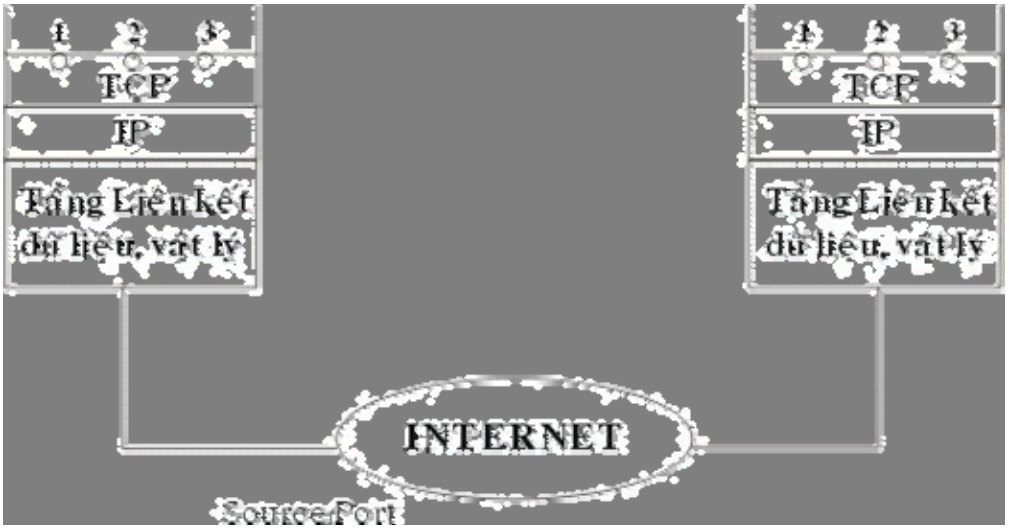
2) Tập hợp các đoạn của gói tin (nếu có phân đoạn)

3) Chuyển dữ liệu và các tham số điều khiển lên tầng trên.

II. Giao thức điều khiển truyền dữ liệu TCP

TCP là một giao thức "có liên kết" (connection - oriented), nghĩa là cần phải thiết lập liên kết giữa hai thực thể TCP trước khi chúng trao đổi dữ liệu với nhau. Một tiến trình ứng

dụng trong một máy tính truy nhập vào các dịch vụ của giao thức TCP thông qua một cổng (port) của TCP. Số hiệu cổng TCP được thể hiện bởi 2 bytes.



Hình 7.5: Cổng truy nhập dịch vụ TCP

Một cổng TCP kết hợp với địa chỉ IP tạo thành một đầu nối TCP/IP (socket) duy nhất trong liên mạng. Dịch vụ TCP được cung cấp nhờ một liên kết logic giữa một cặp đầu nối TCP/IP. Một đầu nối TCP/IP có thể tham gia nhiều liên kết với các đầu nối TCP/IP ở xa khác nhau. Trước khi truyền dữ liệu giữa 2 trạm cần phải thiết lập một liên kết TCP giữa chúng và khi không còn nhu cầu truyền dữ liệu thì liên kết đó sẽ được giải phóng.

Các thực thể của tầng trên sử dụng giao thức TCP thông qua các hàm gọi (function calls) trong đó có các hàm yêu cầu để yêu cầu, để trả lời. Trong mỗi hàm còn có các tham số dành cho việc trao đổi dữ liệu.

Các bước thực hiện để thiết lập một liên kết TCP/IP: Thiết lập một liên kết mới có thể được mở theo một trong 2 phương thức: chủ động (active) hoặc bị động (passive).

- Phương thức bị động, người sử dụng yêu cầu TCP chờ đợi một yêu cầu liên kết gửi đến từ xa thông qua một đầu nối TCP/IP (tại chỗ). ả gười sử dụng dùng hàm passive Open có khai báo cổng TCP và các thông số khác (mức ưu tiên, mức an toàn)

- Với phương thức chủ động, người sử dụng yêu cầu TCP mở một liên kết với một một đầu nối TCP/IP ở xa. Liên kết sẽ được xác lập nếu có một hàm Passive Open tương ứng đã được thực hiện tại đầu nối TCP/IP ở xa đó.

Bảng liệt kê một vài cổng TCP phổ biến.

Số hiệu cổng	Mô tả
0	Reserved

5	Remote job entry
7	Echo
9	Discard
11	Systat
13	Daytime
15	ả estat
17	Quotd (quote odd day
20	ftp-data
21	ftp (control)
23	Telnet
25	SMTP
37	Time
53	ả ame Server
102	ISO - TSAP
103	X.400
104	X.400 Sending
111	Sun RPC
139	ả et BIOS Session source
160 - 223	Reserved

Khi người sử dụng gửi đi một yêu cầu mở liên kết sẽ được nhận hai thông số trả lời từ TCP.

- Thông số Open ID được TCP trả lời ngay lập tức để gán cho một liên kết cục bộ (local connection name) cho liên kết được yêu cầu. Thông số này về sau được dùng để tham chiếu tới liên kết đó. (Trong trường hợp nếu TCP không thể thiết lập được liên kết yêu cầu thì nó phải gửi tham số Open Failure để thông báo.)

- Khi TCP thiết lập được liên kết yêu cầu nó gửi tham số Open Sucsess được dùng để thông báo liên kết đã được thiết lập thành công. Thông báo này dược chuyển đến trong cả hai trường hợp bị động và chủ động. Sau khi một liên kết được mở, việc truyền dữ liệu trên liên kết có thể được thực hiện.

Các bước thực hiện khi truyền và nhận dữ liệu: Sau khi xác lập được liên kết người sữ dụng gửi và nhận dữ liệu. Việc gửi và nhận dữ liệu thông qua các hàm Send và receive.

- *Hàm Send:* Dữ liệu được gửi xuống TCP theo các khối (block). Khi nhận được một khối dữ liệu, TCP sẽ lưu trữ trong bộ đệm (buffer). ả ếu cờ PUSH được dựng thì toàn bộ dữ liệu trong bộ đệm được gửi, kể cả khối dữ liệu mới đến sẽ được gửi đi. ả gược lại cờ PUSH không được dựng thì dữ liệu được giữ lại trong bộ đệm và sẽ gửi đi khi có cơ hội thích hợp (chẳng hạn chờ thêm dữ liệu nữa để gữi đi với hiệu quả hơn).

- *Hàm reveive*: Ở trạm đích dữ liệu sẽ được TCP lưu trong bộ đệm gắn với mỗi liên kết. ả ếu dữ liệu được đánh dấu với một cờ PUSH thì toàn bộ dữ liệu trong bộ đệm (kể cả các dữ liệu được lưu từ trước) sẽ được chuyển lên cho người sữ dụng. Còn nếu dữ liệu đến không được đánh dấu với cờ PUSH thì TCP chờ tới khi thích hợp mới chuyển dữ liệu với mục tiêu tăng hiệu quả hệ thống.

ả ói chung việc nhận và giao dữ liệu cho người sử dụng đích của TCP phụ thuộc vào việc cài đặt cụ thể. Trường hợp cần chuyển gấp dữ liệu cho người sử dụng thì có thể dùng cờ URGEả T và đánh dấu dữ liệu bằng bit URG để báo cho người sử dụng cần phải sử lý khẩn cấp dữ liệu đó.

Các bước thực hiện khi đóng một liên kết: Việc đóng một liên kết khi không cần thiết được thực hiên theo một trong hai cách: *dùng hàm Close* hoặc *dùng hàm Abort*.

- *Hàm Close*: yêu cầu đóng liên kết một cách bình thường. Có nghĩa là việc truyền dữ liệu trên liên kết đó đã hoàn tất. Khi nhận được một *hàm Close* TCP sẽ truyền đi tất cả dữ liệu còn trong bộ đệm thông báo rằng nó đóng liên kết. Lưu ý rằng khi một người sử dụng đã gửi đi một *hàm Close* thì nó vẫn phải tiếp tục nhận dữ liệu đến trên liên kết đó cho đến khi TCP đã báo cho phía bên kia biết về việc đóng liên kết và chuyển giao hết tất cả dữ liệu cho người sử dụng của mình.

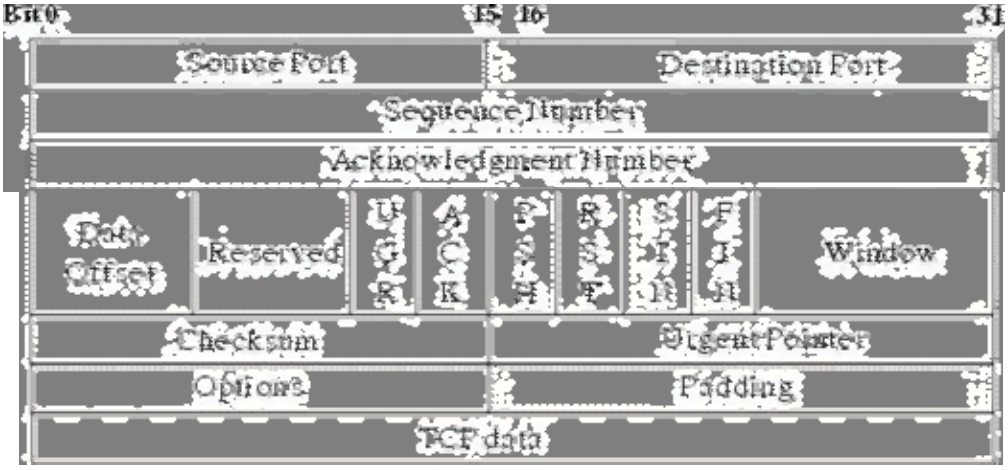
- *Hàm Abort*: ả gười sử dụng có thể đóng một liên kết bất và sẽ không chấp nhận dữ liệu qua liên kết đó nữa. Do vậy dữ liệu có thể bị mất đi khi đang được truyền đi. TCP báo cho TCP ở xa biết rằng liên kết đã được hủy bỏ và TCP ở xa sẽ thông báo cho người sử dụng cũa mình.

Một số hàm khác của TCP:

- *Hàm Status*: cho phép người sử dụng yêu cầu cho biết trạng thái của một liên kết cụ thể, khi đó TCP cung cấp thông tin cho người sử dụng.

- *Hàm Error*: thông báo cho người sử dụng TCP về các yêu cầu dịch vụ bất hợp lệ liên quan đến một liên kết có tên cho trước hoặc về các lỗi liên quan đến môi trường.

Đơn vị dữ liệu sử dụng trong TCP được gọi là segment (đoạn dữ liệu), có các tham số với ý nghĩa như sau:



Hình 7.5: Dạng thức của segment TCP

- Source Por (16 bits): Số hiệu cổng TCP của trạm nguồn.
- Destination Port (16 bit): Số hiệu cổng TCP của trạm đích.
- Sequence ả umber (32 bit): số hiệu của byte đầu tiên của segment trừ khi bit SYả được thiết lập. ả ếy bit SYả được thiết lập thì Sequence ả umber là số hiệu tuần tự khởi đầu (ISả) và byte dữ liệu đầu tiên là ISả +1.
- Acknowledgment ả umber (32 bit): số hiệu của segment tiếp theo mà trạm nguồn đang chờ để nhận. ả gầm ý báo nhận tốt (các) segment mà trạm đích đã gửi cho trạm nguồn.
- Data offset (4 bit): số lượng bội của 32 bit (32 bit words) trong TCP header (tham số này chỉ ra vị trí bắt đầu của nguồn dữ liệu).
- Reserved (6 bit): dành để dùng trong tương lai
- Control bit (các bit điều khiển):
    - URG: Vùng con trỏ khẩn (Ucgent Poiter) có hiệu lực.
    - ACK: Vùng báo nhận (ACK number) có hiệu lực.
    - PSH: Chức năng PUSH.
    - RST: Khởi động lại (reset) liên kết.
    - SYả : Đồng bộ hóa số hiệu tuần tự (sequence number).
    - FIả : Không còn dữ liệu từ trạm nguồn.

- Window (16 bit): cấp phát credit để kiểm soát nguồn dữ liệu (cơ chế cửa sổ). Đây chính là số lượng các byte dữ liệu, bắt đầu từ byte được chỉ ra trong vùng ACK number, mà trạm nguồn đã saün sàng để nhận.

- Checksum (16 bit): mã kiểm soát lỗi cho toàn bộ segment (header + data)

- Urgemt Poiter (16 bit): con trỏ này trỏ tới số hiệu tuần tự của byte đi theo sau dữ liệu khẩn. Vùng này chỉ có hiệu lực khi bit URG được thiết lập.

- Options (độ dài thay đổi): khai báo các option của TCP, trong đó có độ dài tối đa của vùng TCP data trong một segment.

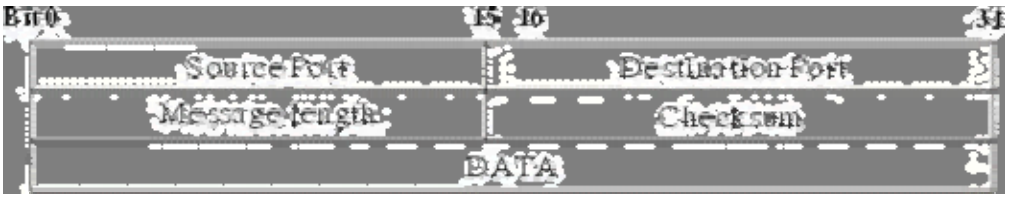
- Paddinh (độ dài thay đổi): phần chèn thêm vào header để đảm bảo phần header luôn kết thúc ở một mốc 32 bit. Phần thêm này gồm toàn số 0.

- TCP data (độ dài thay đổi): chứa dữ liệu của tầng trên, có độ dài tối đa ngầm định là 536 byte. Giá trị này có thể điều chỉnh bằng cách khai báo trong vùng options.

III. Giao thức UDP (User Datagram Protocol)

UDP (User Datagram Protocol) là giao thức theo phương thức không liên kết được sử dụng thay thế cho TCP ở trên IP theo yêu cầu của từng ứng dụng. Khác với TCP, UDP không có các chức năng thiết lập và kết thúc liên kết. Tương tự như IP, nó cũng không cung cấp cơ chế báo nhận (acknowledgment), không sắp xếp tuần tự các gói tin (datagram) đến và có thể dẫn đến tình trạng mất hoặc trùng dữ liệu mà không có cơ chế thông báo lỗi cho người gửi. Qua đó ta thấy UDP cung cấp các dịch vụ vận chuyển không tin cậy như trong TCP.

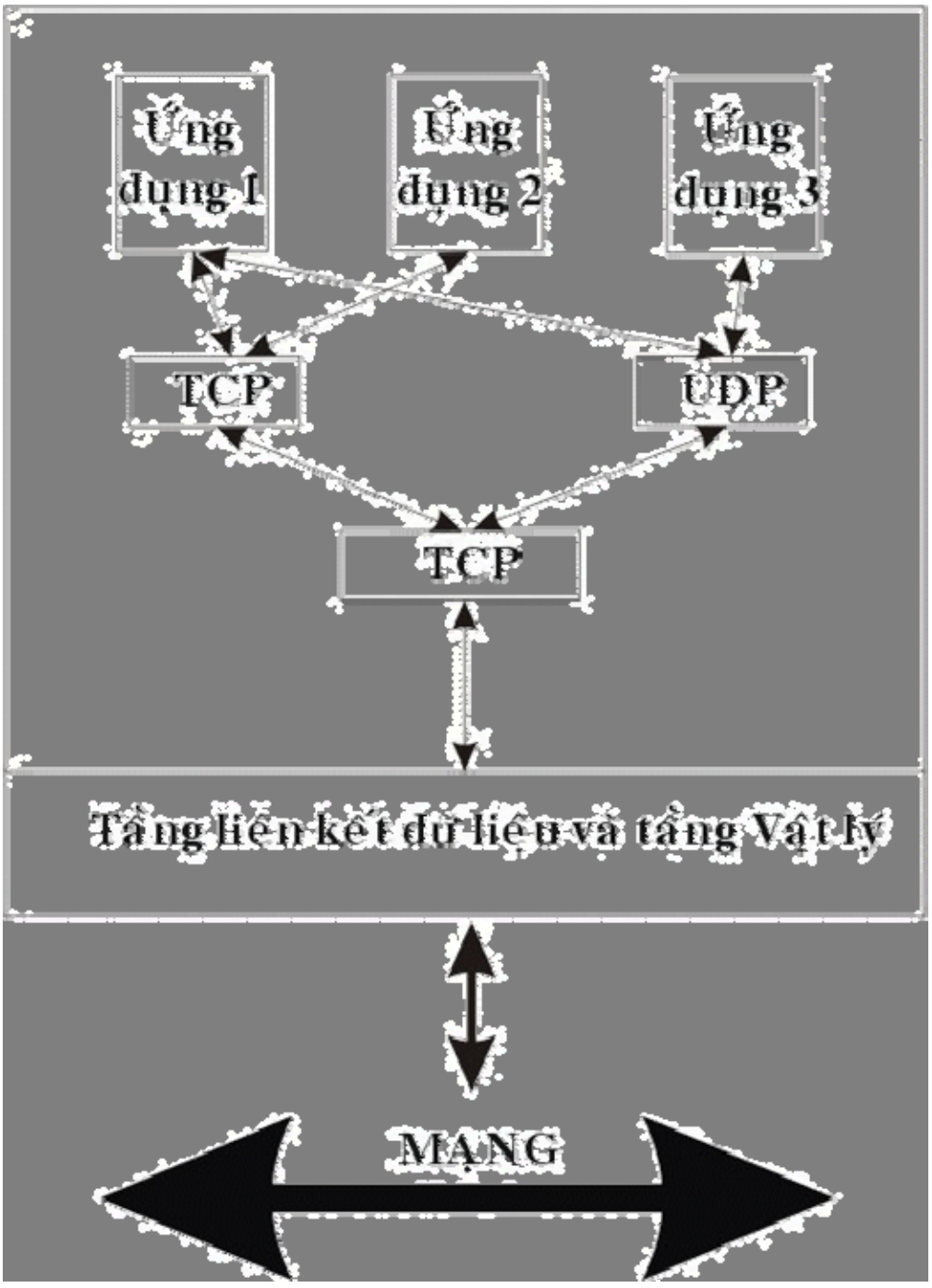
Khuôn dạng UDP datagram được mô tả với các vùng tham số đơn giản hơn nhiều so với TCP segment.



Hình 7.7: Dạng thức của gói tin UDP

UDP cũng cung cấp cơ chế gán và quản lý các số hiệu cổng (port number) để định danh duy nhất cho các ứng dụng chạy trên một trạm của mạng. Do ít chức năng phức tạp nên UDP thường có xu thế hoạt động nhanh hơn so với TCP. ả ó thường được dùng cho các ứng không đòi hỏi độ tin cậy cao trong giao vận.

www.viet-ebook.co.cc



Hình 7.8: Mô hình quan hệ họ giao thức TCP/IP

www.viet-ebook.co.cc

Chương 8

Các dịch vụ của mạng diện rộng (WAN)

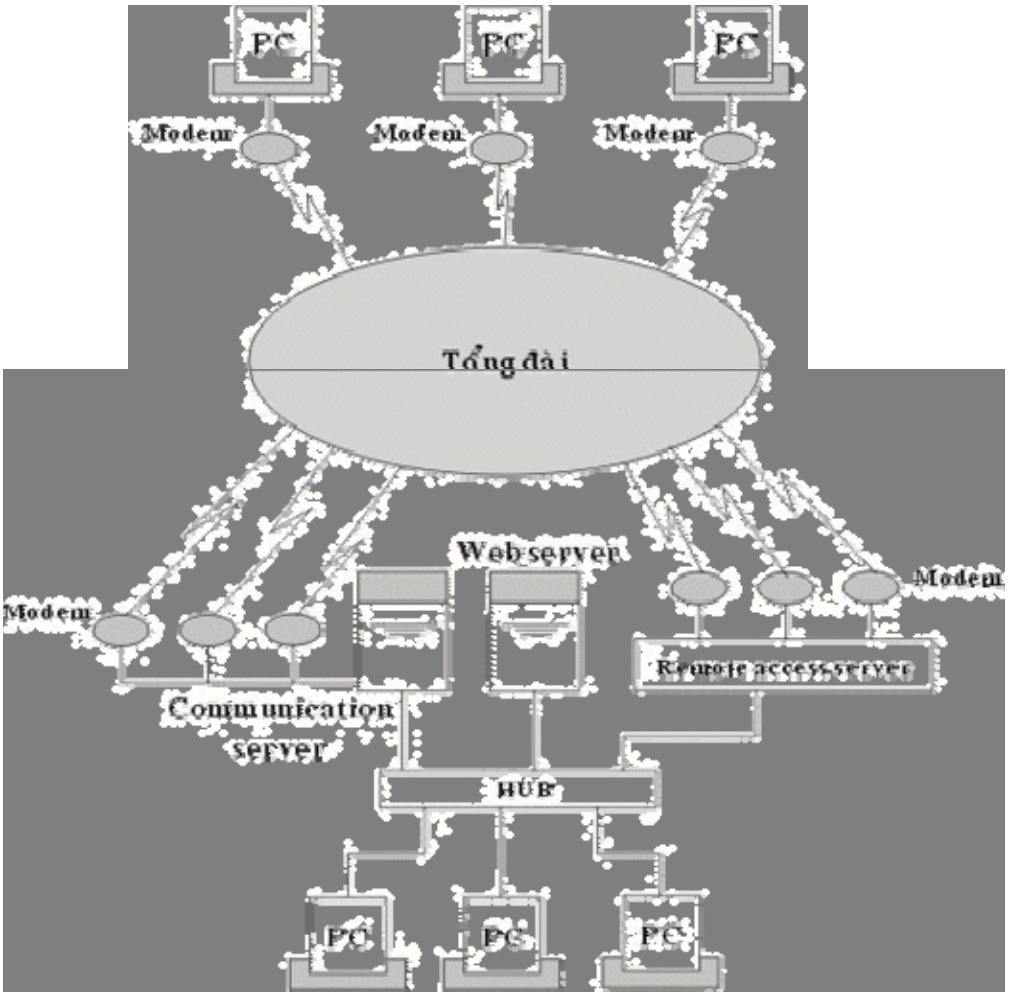
Hiện nay trên thế giới có nhiều dịch vụ dành cho việc chuyển thông tin từ khu vực này sang khu vực khác nhằm liên kết các mạng LAả của các khu vực khác nhau lại. Để có được những liên kết như vậy người ta thường sử dụng các dịch vụ của các mạng diện rộng. Hiện nay trong khi giao thức truyền thông cơ bản của LAả là Ethernet, Token Ring thì giao thức dùng để tương nối các LAả thông thường dựa trên chuẩn TCP/IP. ả gày nay khi các dạng kết nối có xu hướng ngày càng đa dạng và phân tán cho nên các mạng WAả đang thiên về truyền theo đơn vị tập tin thay vì truyền một lần xử lý.

Có nhiều cách phân loại mạng diện rộng, ở đây nếu phân loại theo phương pháp truyền thông tin thì có thể chia thành 3 loại mạng như sau:

- Mạng chuyển mạch (Circuit Swiching ả etwork)
- Mạng thuê bao (Leased lines ả etwork)
- Mạng chuyển gói tin (Packet Switching ả etwork)

I. Mạng chuyển mạch (Circuit Swiching Network)

Để thực hiện được việc liên kết giữa hai điểm nút, một đường nối giữa điểm nút này và điêm nút kia được thiết lập trong mạng thể hiện dưới dạng cuộc gọi thông qua các thiết bị chuyển mạch.



Hình 8.1: Mô hình mạng chuyển mạch

Một ví dụ của mạng chuyển mạch là hoạt động của mạng điện thoại, các thuê bao khi biết số của nhau có thể gọi cho nhau và có một đường nối vật lý tạm thời được thiết lập giữa hai thuê bao.

Với mô hình này mọi đường đều có thể một đường bất kỳ khác, thông qua những đường nối và các thiết bị chuyên dùng người ta có thể liên kết một đường tạm thời từ nơi gửi tới nơi nhận một đường nối vật lý, đường nối trên duy trì trong suốt phiên làm việc và chỉ giải phóng sau khi phiên làm việc kết thúc. Để thực hiện một phiên làm việc cần có các thủ tục đầy đủ cho việc thiết lập liên kết trong đó có việc thông báo cho mạng biết địa chỉ của nút nhận.

Hiện nay có 2 loại mạng chuyển mạch là chuyển mạch tương tự (analog) và chuyển mạch số (digital)

- *Chuyển mạch tương tự (Analog):* Việc chuyển dữ liệu qua mạng chuyển mạch tương tự được thực hiện qua mạng điện thoại. Các trạm sử dụng một thiết bị có tên là modem, thiết bị này sẽ chuyển các tín hiệu số từ máy tính sao tín hiệu tuần tự có trể truyền đi trên mạng điện thoại và ngược lại.

Hình 8.2: Mô hình chuyển mạch tương tự

Khi sử dụng đường truyền điện thoại để truyền số liệu thì các chuẩn của modem và các tính chất của nó sẽ quyết định tốc độ của đường truyền. Cùng với các kỹ thuật chuyển đổi tín hiệu các tính năng mới như nén tín hiệu cho phép nâng tốc độ truyền dữ liệu lên rất cao.

Loại	Tốc độ (bps)ä	Loại nén	Tốc độ thực tế (bps)
Bell 212A	1200		
CCITT V22	1200		
CCITT V22 bis	2400	Mả P Class 5	2400 - 3600
CCITT V32	9600	Mả P Class 5, V42 bis	9600 - 19200
CCITT V32 bis	14400	Mả P Class 5, V42 bis	14400 - 33600

Hình 8.3: Bảng kỹ thuật modem

Các kỹ thuật nén thường dùng là Mả P Class 5 và V42 bis, Mả P Class 5 cho phép nén với tỷ lệ 1.5:1 và V42 bis nén với tỷ lệ 2:1. Tuy nhiên trên thực tế tỷ lệ nén có thể thay đổi dựa vào dạng dữ liệu được truyền.

- *Chuyển mạch số (Digital):* Đường truyền chuyển mạch số lần đầu tiên được AT&T thiệu vào cuối 1980 khi AT&T giới thiệu mạng chuyển mạch số Acnet với đường truyền 56 kbs. Việc sử dụng đường chuyển mạch số cũng đòi hỏi sử dụng thiết bị phục vụ truyền dữ liệu số (Data Service Unit - DSU) vào vị trí modem trong chuyển mạch tương tự. Thiết bị phục vụ truyền dữ liệu số có nhiệm vụ chuyển các tín hiệu số đơn chiều (unipolar) từ máy tính ra thành tín hiệu số hai chiều (bipolar) để truyền trên đường truyền.

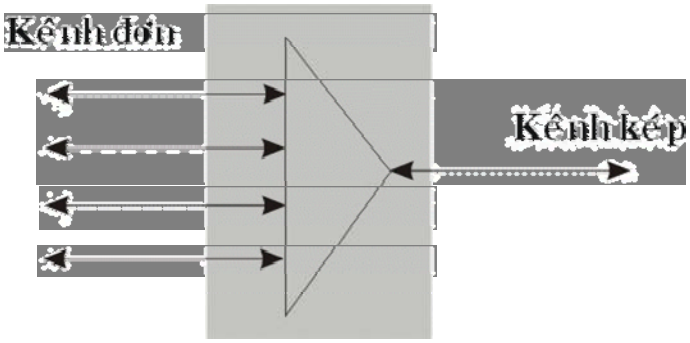
Hình 8.3: Mô hình chuyển mạch số

Mạng chuyển mạch số cho phép người sử dụng nâng cao tốc độ truyền (ở đây do khác biệt giữa kỹ thuật truyền số và kỹ thuật truyền tương tự nên hiệu năng của truyền mạch số cao hơn nhiều so với truyền tương tự cho dù cùng tốc độ), độ an toàn.

Vào năm 1991 AT&T giới thiệu mạng chuyển mạch số có tốc độ 384 Kbps. ả gười ta có thể dùng mạng chuyển mạch số để tạo các liên kết giữa các mạng LAả và làm các đường truyền dự phòng.

II. Mạng thuê bao (Leased line Network)

Với kỹ thuật chuyển mạch giữa các nút của mạng (tương tự hoặc số) có một số lượng lớn đường dây truyền dữ liệu, với mỗi đường dây trong một thời điểm chỉ có nhiều nhất một phiên giao dịch, khi số lượng các trạm sử dụng tăng cao người ta nhận thấy việc sử dụng mạng chuyển mạch trở nên không kinh tế. Để giảm bớt số lượng các đường dây kết nối giữa các nút mạng người ta đưa ra một kỹ thuật gọi là ghép kênh.



Hình 8.4: Mô hình ghép kênh

Mô hình đó được mô tả như sau: tại một nút người ta tập hợp các tín hiệu trên của nhiều người sử dụng ghép lại để truyền trên một kênh nối duy nhất đến các nút khác, tại nút cuối người ta phân kênh ghép ra thành các kênh riêng biệt và truyền tới các người nhận.

Có hai phương thức ghép kênh chính là ghép kênh theo tần số và ghép kênh theo thời gian, hai phương thức này tương ứng với mạng thuê bao tuần tự và mạng thuê bao kỹ thuật số. trong thời gian hiện nay mạng thuê bao kỹ thuật số sử dụng kỹ thuật ghép kênh theo thời gian với đường truyền T đang được sử dụng ngày một rộng rãi và dần dần thay thế mạng thuê bao tuần tự.

1. Phương thức ghép kênh theo tần số

Để sử dụng phương thức ghép kênh theo tần số giữa các nút của mạng được liên kết bởi đường truyền băng tần rộng. Băng tần này được chia thành nhiều kênh con được phân biệt bởi tần số khác nhau. Khi truyền dử liệu, mỗi kênh truyền từ người sử dụng đến nút sẽ được chuyển thành một kênh con với tần số xác định và được truyền thông qua bộ ghép kênh đến nút cuối và tại đây nó được tách ra thành kênh riêng biệt để truyền tới người nhận. Theo các chuẩn của CCITT có các phương thức ghép kênh cho phép ghép 12, 60, 300 kênh đơn.

ả gười ta có thể dùng đường thuê bao tuần tự (Analog) nối giữa máy của người sử dụng tới nút mạng thuê bao gần nhất. Khi máy của người sử dụng gửi dữ liệu thì kênh dữ liệu được

ghép với các kênh khác và truyền trên đường truyền tới nút đích và được phân ra thành kênh riêng biệt trước khi gửi tới máy của người sử dụng. Đường nối giữa máy trạm của người sử dụng tới nút mạng thuê bao cũng giống như mạng chuyển mạch tuần tự sử dụng đường dây điện thoại với các kỹ thuật chuyển đổi tín hiệu như V22, V22 bis, V32, V32 bis, các kỹ thuật nén V42 bis, Mả P class 5.

2.Phương thức ghép kênh theo thời gian:

Khác với phương thức ghép kênh theo tần số, phương thức ghép kênh theo thời gian chia một chu kỳ thời gian hoạt động của đường truyền trục thành nhiều khoảng nhỏ và mỗi kênh tuyền dữ liệu được một khoảng. Sau khi ghép kênh lại thành một kênh chung dữ liệu được truyền đi tương tự như phương thức ghép kênh theo tần số. ả gười ta dùng đường thuê bao là đường truyền kỹ thuật số nối giữa máy của người sử dụng tới nút mạng thuê bao gần nhất.

Hiện nay người ta có các đường truyền thuê bao như sau :

Đường T1 với tốc độ 1.544 Mbps nó bao gồm 24 kênh vớp tốc độ 64 kbps và 8000 bits điều khiển trong 1 giây.

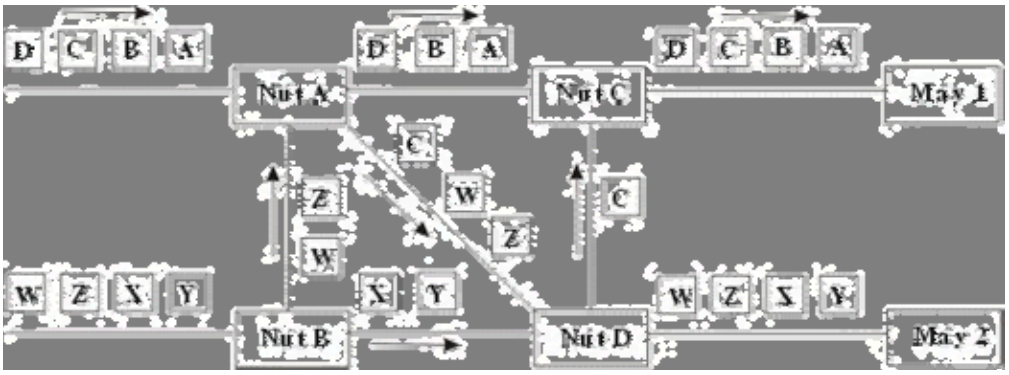
III. Mạng chuyển gói tin (Packet Switching NetWork)

Mạng chuyển mạch gói hoạt động theo nguyên tắc sau : Khi một trạm trên mạng cần gửi dữ liệu nó cần phải đóng dữ liệu thành từng gói tin, các gói tin đó được đi trên mạng từ nút này tới nút khác tới khi đến được đích. Do việc sử dụng kỹ thuật trên nên khi một trạm không gửi tin thì mọi tài nguyên của mạng sẽ dành cho các trạm khác, do vậy mạng tiết kiệm được các tài nguyên và có thể sử dụng chúng một cách tốt nhất.

ả gười ta chia các phương thức chuyển mạch gói ra làm 2 phương thức:

- Phương thức chuyển mạch gói theo sơ đồ rời rạc.
- Phương thức chuyển mạch gói theo đường đi xác định.

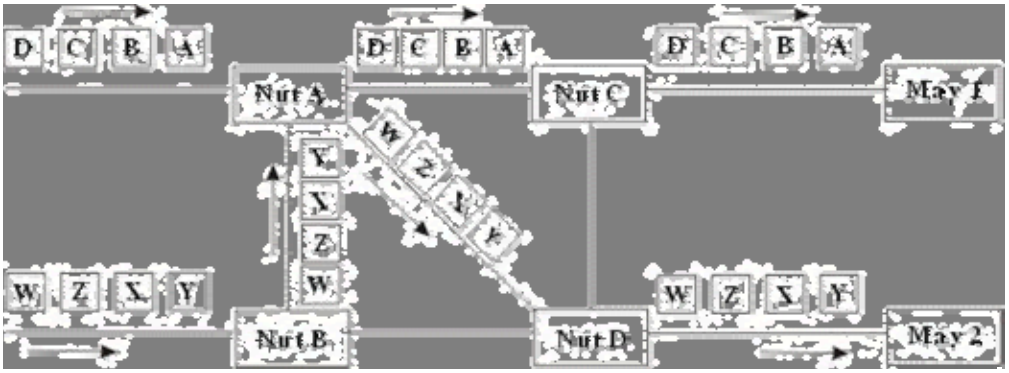
Với phương thức chuyển mạch gói theo sơ đồ rời rạc các gói tin được chuyển đi trên mạng một cách độc lập, mỗi gói tin đều có mang địa chỉ nơ i gửi và nơi nhận. Mỗi nút trong mạng khi tiếp nhận gói tin sẽ quyết định xenm đường đi của gói tin phụ thuộc vào thuật toán tìm đường tại nút và những thông tin về mạng mà nút đó có. Việc truyền theo phương thức này cho ta sự mềm dẻo nhất định do đường đi với mỗi gói tin trở nên mềm dẻo tuy nhiên điều này yêu cầu một số lượng tính toán rất lớn tại mỗi nút nên hiện nay phần lớn các mạng chuyển sang dùng phương chuyển mạch gói theo đường đi xác định.



Hình 8.5: Ví dụ phương thức sơ đồ rời rạc.

Phương thức chuyển mạch gói theo đường đi xác định:

Trước khi truyền dữ liệu một đường đi (hay còn gọi là đường đi ảo) được thiết lập giữa trạm gửi và trạm nhận thông qua các nút của mạng. Đường đi trên mang số hiệu phân biệt với các đường đi khác, sau đó các gói tin được gửi đi theo đường đã thiết lập để tới đích, các gói tin mang số hiệu củ đường ảo để có thể được nhận biết khi qua các nút. Điều này khiến cho việc tính toán đường đi cho phiên liên lạc chỉ cần thực hiện một lần.



Hình 8.6: Ví dụ phương thức đường đi xác định

1. Mạng X25

Được CCITT công bố lần đầu tiên vào 1970 lúc lĩnh vực viễn thông lần đầu tiên tham gia vào thế giới truyền dữ liệu với các đặc tính:

- X25 cung cấp quy trình kiểm soát luồng giữa các đầu cuối đem lại chất lương đường truyền cao cho dù chất lương đương dây truyền không cao.
- X25 được thiết kế cho cả truyền thông chuyển mạch lẫn truyền thông kiểu điểm nối điểm.
- Được quan tâm và tham gia nhanh chóng trên toàn cầu.

Trong X25 có chức năng dồn kênh (multiplexing) đối với liên kết logic (virtual circuits) chỉ làm nhiệm vụ kiểm soát lỗi cho các frame đi qua. Điều này làm tăng độ phức tạp trong việc phối hợp các thủ tục giữa hai tầng kề nhau, dẫn đến thông lượng bị hạn chế do tổng phí xử lý mỗi gói tin tăng lên. X25 kiểm tra lỗi tại mỗi nút trước khi truyền tiếp, điều này làm cho đường truyền chó chất lượng rất cao gần như phi lỗi. Tuy nhiên do vậy khối lượng tích toán tại mỗi nút khá lớn, đối với những đường truyền của những năm 1970 thì điều đó là cần thiết nhưng hiện nay khi kỹ thuật truyền dẫn đã đạt được những tiến bộ rất cao thì việc đó trở nên lãng phí

2. Mạng Frame Relay

Mỗi gói tin trong mạng gọi là Frame, do vậy mạng gọi là Frame relay. Đặc điểm khác biệt giữa mạng Frame Relay và mạng X25 mạng Frame Relay là chỉ kiểm tra lỗi tại hai trạm gửi và trạm nhận còn trong quá trình chuyển vận qua các nút trung gian gói tin sẽ không được kiểm lỗi nữa. Do vậy thời gian xử lý trên mỗi nút nhanh hơn, tuy nhiên khi có lỗi thì gói tin phải được phát lại từ trạm đầu. Với độ an toàn cao của đường truyền hiện nay thì chi phí việc phát lại đó chỉ chiếm một tỷ lệ nhỏ nếu so với khối lượng tính toán được giảm đi tại các nút nên mạng Frame Relay tiết kiệm được tài nguyên của mạng hơn so với mạng X25.

Frame relay không chỉ là một kỹ thuật mà còn là thể hiện một phương pháp tổ chức mới. Với nguyên lý là truyền mạch gói nhưng các thao tác kiểm soát giữa các đầu cuối giảm đáng kể Kỹ thuật Frame Relay cho phép thông lượng tối đa đạt tới 2Mbps và hiện nay nó đang cung cấp các giải pháp để tương nối các mạng cục bộ LAả trong một kiến trúc xương sống tạo nên môi trường cho ứng dụng multimedia.

3. Mạng ATM (Cell relay)

Hiện nay kỹ thuật Cell Relay dựa trên phương thức truyền thông không đồng bộ (ATM) có thể cho phép thông lương hàng trăm Mbps. Đơn vị dữ liệu dùng trong ATM được gọi là tế bào (cell). các tế bào trong ATM có độ dài cố định là 53 bytes, trong đó 5 bytes dành cho phần chứa thông tin điều khiển (cell header) và 48 bytes chứa dữ liệu của tầng trên.

Trong kỹ thuật ATM, các tế bào chứa các kiểu dữ liệu khác nhau được ghép kênh tới một đường dẫn chung được gọi là đường dẫn ảo (virtual path). Trong đường dẫn ảo đó có thể gồm nhiều kênh ảo (virtual chanell) khác nhau, mỗi kênh ảo được sử dụng bởi một ứng dung nào đó tại một thời điểm.

ATM đã kết hợp những đặc tính tốt nhất của dạng chuyển mạch liên tục và dạng chuyển mạch gói, nó có thể kết hợp dải thông linh hoạt và khả năng chuyển tiếp cao tốc và có khả năng quản lý đồng thời dữ liệu số, tiếng nói, hình ành và multimedia tương tác.

Mục tiêu của kỹ thuật ATM là nhằm cung cấp một mạng dồn kênh, và chuyển mạch tốc độ cao, độ trễ nhỏ dáp ứng cho các dạng truyền thông đa phương tiện (multimecdia)

Chuyển mạch cell cần thiết cho việc cung cấp các kết nối đòi hỏi băng thông cao, tình trạng tắt nghẽn thấp, hổ trợ cho lớp dịch vụ tích hợp lưu thông dữ liệu âm thanh hình ảnh. Đặc tính tốc độ cao là đặc tính nổi bật nhất của ATM.

ATM sử dụng cơ cấu chuyển mạch đặc biệt: ma trận nhị phân các thành tố chuyển mạch (a matrix of binary switching elements) để vận hành lưu thông. Khả năng vô hướng (scalability) là một đặc tính của cơ cấu chuyển mạch ATM. Đặc tính này tương phản trực tiếp với những gì diễn ra khi các trạm cuối được thêm vào một thiết bị liên mạng như router. Các router có năng suất tổng cố định được chia cho các trạm cuối có kết nối với chúng. Khi số lượng trạm cuối gia tăng, năng suất của router tương thích cho trạm cuối thu nhỏ lại. Khi cơ cấu ATM mở rộng, mỗi thiết bị thu trạm cuối, bằng con đường của chính nó đi qua bộ chuyển mạch bằng cách cho mỗi trạm cuối băng thông chỉ định. Băng thông rộng được chỉ định của ATM với đặc tính có thể xác nhận khiến nó trở thành một kỹ thuật tuyệt hảo dùng cho bất kỳ nơi nào trong mạng cục bộ của doanh nghiệp.

ả hư tên gọi của nó chỉ rõ, kỹ thuật ATM sử dụng phương pháp truyền không đồng bộ (asynchronouns) các tề bào từ nguồn tới đích của chúng. Trong khi đó, ở tầng vật lý người ta có thể sử dụng các kỹ thuật truyền thông đồng bộ như SDH (hoặc SOả ET).

ả hận thức được vị trí chưa thể thay thế được (ít nhất cho đến những năm đầu của thế kỷ 21) của kỹ thuật ATM, hầu hết các hãng khổng lồ về máy tính và truyền thông như IBM, ATT, Digital, Hewlett - Packard, Cisco Systems, Cabletron, Bay ả etwork,... đều đang quan tâm đặc biệt đến dòng sản phẩm hướng đến ATM của mình để tung ra thị trường. Có thể kể ra đây một số sản phẩm đó như DEC 900 Multiwitch, IBM 8250 hub, Cisco 7000 rounter, Cablectron, ATM module for MMAC hub.

ả hìn chung thị trường ATM sôi động do nhu cầu thực sự của các ứng dụng đa phương tiện. Sự nhập cuộc ngày một đông của các hãng sản xuất đã làm giảm đáng kể giá bán của các sản phẩm loại này, từ đó càng mở rộng thêm thị trường. ả gay ở Việt ả am, các dự án lớn về mạng tin học đều đã được thiết kế với hạ tầng chấp nhận được với công nghệ ATM trong tương lai.

www.viet-ebook.co.cc

Chương 9

Ví dụ một số mạng LAN và WAN

Hiện nay trên thế giới có rất nhiều mạng máy tính, chúng được sử dụng để phục vụ cho nhiều lĩnh vực khác nhau như nghiên cứu khoa học, truyền dữ liệu, kinh doanh. Vì vậy nên các mạng này cũng rất đa dạng về chủng loại. Trong phần này ta xem xét một số mạng LAả và WAả thông dụng.

I. Mạng Novell NetWare

Được đưa ra bởi hãng ả ovell từ những năm 80 và đã được sử dụng nhiều trong các mạng cục bộ với số lượng ước tính hiện nay vào khoảng 50 -60%. Hệ điều hành mạng ả ovell ả etWare là một hệ điều hành có độ an toàn cao đặc biệt là với các mạng có nhiều người sử dụng. Hệ điều hành mạng ả etware khá phức tạp để lắp đặt và quản lý nhưng nó là một hệ điều hành mạng đang được dùng phổ biến nhất hiện nay. Hệ điều hành mạng ả ovell ả etWare được thiết kế như một hệ thống mạng *client-server* trong đó các máy tính được chia thành hai loại:

- ả hững máy têêính cung cấp tài nguyên cho mạng gọi là *server* hay còn gọi là máy chủ mạng.
- Máy sử dụng tài nguyên mạng gọi là *clients* hay còn gọi là trạm làm việc.

Các server (File server) của ả etware không chạy DOS mà bản thân ả etware là một hệ điều hành cho server điều đó đã giải phóng ả etware ra khỏi những hạn chế của DOS. Server của ả etware dùng một cấu trúc hiệu quả hơn DOS để tổ chức các tập tin và thư mục, với ả etware, chúng ta có thể chia mỗi ổ đĩa thành một hoặc nhiều tập đĩa (volumes), tương tự như các ổ đĩa logic của DOS. Các tập đĩa của ả ovell có tên chứ không phải là chữ cái. Tuy nhiên, để truy cập một tập đĩa của ả etware từ một trạm làm việc chạy DOS, một chữ cái được gán cho tập đĩa.

Với các hệ điều hành ả etware 3.x và 4.x các server phải được dành riêng, trong đó chúng ta không thể dùng một file server làm thêm việc cùa Workstation, tuy điều đó tốn kém hơn vì phải mua một máy tính để làm server nhưng nó có hiệu quả hơn vì máy tính server có thể tập trung để phục vụ mạng. Còn với ả etware 2.x thì có thể lưa chọn trong đó một file server có thể làm việc như một Workstation như hai tiến trình Server và Workstation tách tời nhau hoàn toàn.

Các trạm làm việc trên một mạng ả etware có thể là các máy tính DOS, chạy OS/2 hoặc các máy Macintosh. ả ếu mạng vừa có máy PC và Macintosh thì ả etware có thể là sự lựa chọn tốt.

Tất cả các phiên bản của ả etware đều có đặc trưng được gọi là tính chịu đựng sai hỏng của hệ (System Fault Tolerance SFT) được thiết kế để giữ cho mạng vẫn chạy ngay cả khi phần cứng có sai hỏng.

ả etWare là một hệ điều hành nhưng không phải là một hệ điều hành đa năng mà tập trung chủ yếu cho các ứng dụng truy xuất tài nguyên trên mạng, nó có một tập hợp xác định saün

các dịch vụ dành cho người sử dụng. Tại đây ả ovell ả etWare có một hệ thống các yêu cầu và trả lời mà Client và Server đều hiểu, nó bao gồm:

- ả hóm chương trình trên máy người dùng: Hệ điều hành trạm, các giao diện cho phép nhười sử dụng chi xuất các tài nguyên của mạng như là các tài nguyên của máy cục bộ, chương trình truyền số liệu qua mạng.

- Hệ điều hành trên máy máy chủ: Chương trình thực hiên từ DOS, Lưu các thông số của DOS, chuyển CPU của server qua chế độ protectied mode, quản lý việc sử dụng tài nguyên của mạng cho người sử dụng.

- Các tiện ích trên mạng: dành cho người sử dụng và người quản trị mạng.

- ả ovell ả etWare hỗ trợ các giao thức cơ bản sau:

- Giao thức truy xuất (Access Protocol) (Ethernet, Token Ring, ARCnet, Proả ET-10, FDDI)

- Giao thức trao đổi gói tin trên mạng (Internet Packet Exchange -IPX)

- Giao thức thông tin tìm đường (Routing Information Protocol - RIP)

- Giao thức thông báo dịch vụ (Sevice Advertising Protocol - SAP)

- Giao thức nhân ả etWare (ả etWare Core Protocol - ả CP) cho phép người dùng truy xuất vào file server

Do nhu cầu cần thích nghi với nhiều kiểu mạng và để dễ dàng nâng cấp và quản lý, ả ovell ả etWare cũng được chia thành nhiều tầng giao thức tương tự cấu trúc 7 tầng cuả hệ thống mở OSI.

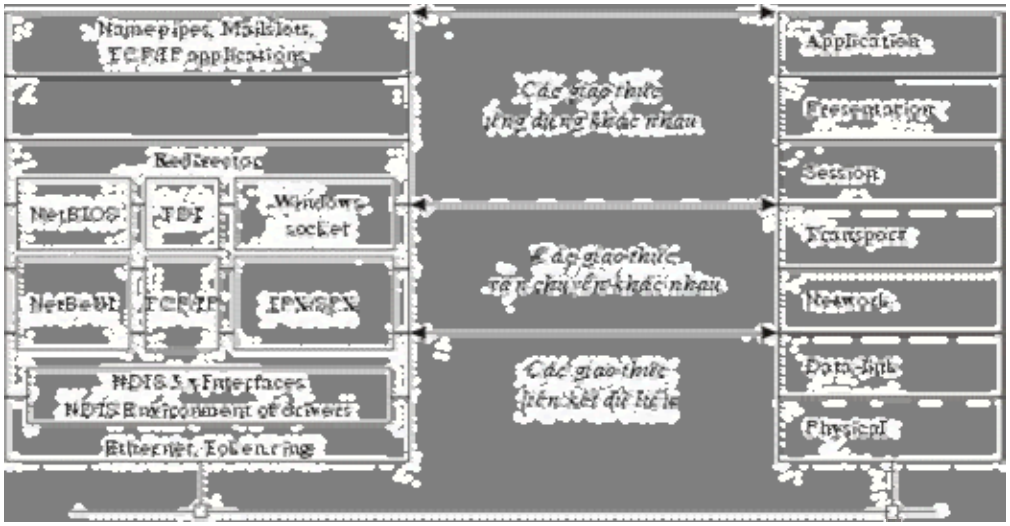
<table>
<tr><td>Tầng ứng dụng
(Application)</td><td rowspan="2">Giao thức
thông báo dịch vụ
(Service advertising
protocol - SAP)</td><td rowspan="2">Giao thức thông
tin tìm đường
(Routing Information
Protocol - RIP)</td><td rowspan="2">Giao thức
nhân NetWare
(NetWare core
protocol - NCP)</td></tr>
<tr><td>Tầng trình bày
(Presentation)</td></tr>
<tr><td>Tầng giao dịch
(Session)</td><td colspan="3">Hệ thống nhập xuất cơ bản trên mạng
(NetBIOS)</td></tr>
<tr><td>Tầng vận chuyển
(Transport)</td><td colspan="3">Trao đổi gói tin tuần tự
(Sequence Packet eXchange - SPX)</td></tr>
<tr><td>Tầng mạng
(Network)</td><td colspan="3">Trao đổi gói tin liên mạng
(Internet Packet Exchange - IPX)</td></tr>
<tr><td>Tầng liên kết dữ liệu
(Data link)</td><td colspan="3">Giao thức truy xuất và kỹ thuật mạng lưới
(Access protocol and wiring techniques)
(Chuẩn giao tiếp liên kết dữ liệu mở ODI)</td></tr>
<tr><td>Tầng vật lý
(Physical)</td><td colspan="3">Ethernet, Token Ring, ARCnet
cáp đồng trục, cáp trần xoắn cặp
(IEEE 802.X hoặc FDDI)</td></tr>
</table>

Hình 9.1: Cấu trúc của Hệ điều hành Novell NetWare

II. Mạng Windows NT

Mạng dùng hệ điều hành **Windows NT** được đưa ra bởi hãng Microsoft với phiên bản mới nhất hiện nay là Windows ả T 5.0, cụm từ windows ả T được hiểu là công nghệ mạng trong môi trường Windows (Windows ả etwork Technology). Hiện mạng Windows ả T đang được đánh giá cao và được đua vào sử dụng ngày một nhiều. Windows ả T là một hệ điều hành đa nhiệm, đa xử lý với địa chỉ 32 bit bộ nhớ. ả goài việc yểm trơ các ứng dụng DOS, Windows 3.x, Win32 GUI và các ứng dụng dựa trên ký tự, Windows ả T còn bao gồm các thành phần mạng, cơ chế an toàn, các công cụ quản trị có khả năng mạng diện rộng, các phần mềm truy cập từ xa. Windows ả T cho phép kết nối với máy tính lớn, mini và máy Mac.

Hệ điều hành mạng Windows ả T có thể chay trên máy có một CPU cũng như nhiều CPU. Hệ điều hành mạng còn có đưa vào kỹ thuật gương đĩa qua đó sử dụng tốt hệ thống nhiều đĩa nâng cao năng lực hoạt động. Hệ điều hành mạng Windows ả T đảm bảo tránh được những người không được phép vào trong hệ thống hoặc thâm nhập vào các file và chương trình trên đĩa cứng. Hệ điều hành mạng Windows ả T cung cấp các công cụ để thiết lập các lớp quyền dành cho nhiều nhiệm vụ khác nhau làm cho phép xây dựng hệ thống an toàn một cách mềm dẻo. Windows ả T được thiết kế dành cho giải pháp nhóm (Workgroup) khi bạn muốn có kiểm soát nhiều hơn đối với mạng ngang hàng (như Windows For Workgroup, LAả tastic hay ả ovell lite). ả goài ra chức năng mới của Windows ả T server là mô hình vùng (Domain) được thiết lập cho các mạng lớn với khả năng kết nối các mạng toàn xí nghiệp hay liên kết các kết nối mạng với các mạng khác và những công cụ cần thiết để điều hành.



Hình 9.2: Cấu trúc của Hệ điều hành Windows NT

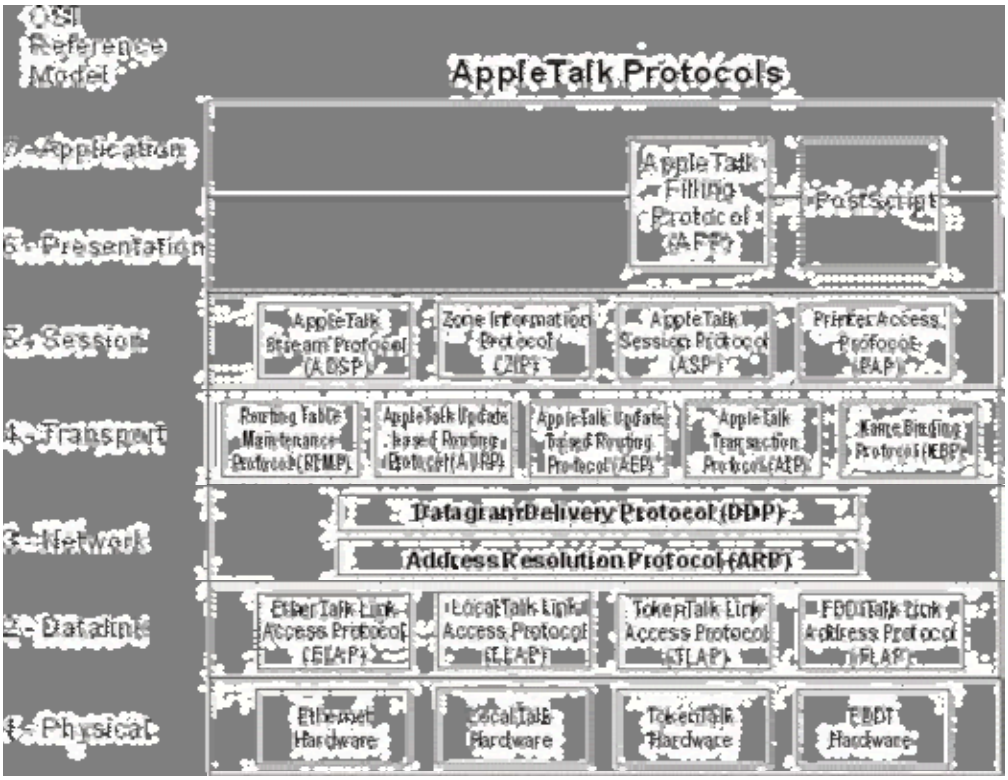
III. Mạng Apple talk

Vào đầu những năm 1980, khi công ty máy tính Apple chuẩn bị giới thiệu máy tính Macintosh, các kỹ sư Apple đã thấy rằng mạng sẽ trở nên rất cần thiết. Họ muốn rằng mạng MAC cũng là một bước tiến mơi trong cuộc cách mạng về giao diện thân thiện người dùng do Apple khởi xướng. Với ý định như vậy, Apple xây dựng một giao thức mạng cho

họ máy Macintosh, và tích hợp giao thức trên vào máy tính để bàn. Cấu trúc mạng mới do Apple xây dựng được gọi là Apple Talk.

Mặc dù Apple Talk là giao thức mạng độc quyền của Apple, nhưng Apple cũng đã ấn hành nhiều tài liệu về Apple Talk trong cố gắng khuyến khích các nhà sản xuất phần mềm khác phát triển trên Apple Talk. ả gày nay đã có nhiều sản phẩm thương mại trên nền Apple Talk như của ả ovell, Microsoft.

Ban đầu **AppleTalk** chỉ cài đặt trên hệ thống cáp riêng của hãng là LocalTalk và có phạm vi ứng dụng rất hạn chế. Phiên bản đầu của Apple Talk được thiết kế cho nhóm người dùng cục bộ hay được gọi là *Apple Talk phase 1*. Sau khi tung ra thị trường 5 năm, số người dùng đã vượt quá 1,5 triệu người cài đặt, Apple nhận thấy những nhóm người dùng lớn đã vượt quá giới hạn của *Apple Talk phase 1*, nên họ đã nâng cấp giao thức. Giao thức đã được cải tiến được biết dưới cái tên *Apple Talk phase 2,* cải tiến khả năng tìm đường của Apple Talk và cho phép Apple Talk chạy trên những mạng lớn hơn.



Hình 9.3: Cấu trúc của Hệ điều hành Appletalk

Hãng Apple thiết kế Apple Talk độc lập với tầng liên kết dữ liệu. Apple hỗ trợ nhiều loại cài đặt của tầng liên kết dữ liệu, bao gồm *Ethernet, Token Ring, Fiber Distributed Data Interface* (FDDI), và *Local Talk*. Trên Apple Talk, Apple xem Ethernet như *ethertalk*, Token Ring như *tokentalk*, và FDDI như *fdditalk*.

Các giao thức chính của mạng AppleTalk:

- ***LLAP*** *(Local Talk Link Access)* là giao thức do Apple phát triển để hoạt động với cáp riêng của hãng (cũng được gọi là LocalTalk**)** dựa trên cáp xoắn đôi bọc kim

(STP), thích hợp với các mạng nhỏ, hiệu năng thấp. Tốc độ tối đa là 230,4 Kb/s và khoảng cách các đoạn cáp có độ dài giới hạn là 300m, số lượng trạm tối đa là 32.

- ***ELAP*** *(Ethertalk Link Access)* và *TLAP (tokentalk Link Access)* là các giao thức cho phép sử dụng các mạng vật lý tương ứng là Ethernet và Token Ring.

- ***AARP*** *(AppleTalk Addresss Resolution Protocol)* là các giao thức cho phép ánh xạ giữa các địa chỉ vật lý của Ethernet và Token Ring, là giao diện giữa các tầng cao của AppleTalk với các tầng vật lý của Ethernet và Token Ring.

- ***DDP*** *(Datagram Delivery Protocol)* là giao thức tầng Mạng cung cấp dịch vụ theo phương thức không liên kết giữa 2 sockets (để chỉ 1 địa chỉ dịch vụ; một tổ hợp của địa chỉ thiết bị, địa chỉ mạng và socket sẽ định danh 1 cách duy nhất cho môãi tiến trình). DDP thực hiện chức năng chọn đường (routing) dựa trên các bảng chọn đường cho RTMP bảo trì.

- ***RTMP*** *(Routing Table Maintenance protocol)* cung cấp cho DDP thông tin chọn đường trên phương pháp vector khoảng cách tương tự như RIP (Routing Information Protocol) dùng trong ả etware IPX/SPX.

- ***NBP*** *(Naming Binding Protocol)*: cho phép định danh các thiết bị bởi các tên lôgic (ngoài địa chỉ của chúng). Các tên này ẩn dấu địa chỉ tầng thấp đối với người sử dụng và đối với các tầng cao hơn.

- ***ATP*** *(AppleTalk Transaction Protocol)* là giao thức thức tầng vận chuyển hoạt động với phương thức không liên kết. Dich vụ vận chuyển này được cung cấp thông qua một hệ thống các thông báo nhận và truyền lại. Độ tin cậy cũa ATP dựa trên các thao tác (transaction) (một thao tác bao gồm một cặp các thao tác hỏi-đáp).

- ***ASP*** *(AppleTalk Section Protocol)* là giao thức tầng giao dịch của AppleTalk, cho phép thiết lập, duy trì và hủy bỏ các phiên liên lạc giữa người yêu cầu dịch vụ và người cung cấp dịch vụ.

- ***ADSP*** *(AppleTalk Data Stream Protocol)* là một giao thức phủ cả tầng vận chuyển và tầng giao dịch, có thể thay cho nhóm giao thức dùng với ATP.

- ***ZIP*** *(Zone Information Protocol)* là giao thức có chức năng tổ chức các thiết bị thành các vùng (zone) để làm giảm độ phức tạp của 1 mạng bằng cách giới hạn sự tương tác của người sử dụng vào đúng các thiết bị mà anh ta cần.

- ***PAP*** *(Printer Access protocol)* cũng là 1 giao thức của tầng giao dịch tương tự như ASP. ả ó không chỉ cung cấp các dịch vụ in như tên gọi mà còn yểm trợ các kiểu liên kết giữa người yêu cầu và người cung cấp dịch vụ.

- ***AFP*** *(AppleTalk Filling Protocol)* là giao thức cung cấp dịch vụ File và đảm nhận việc chuyển đổi cú pháp dữ liệu, bảo vệ an toàn dữ liệu (tương tự tầng trình bày trong mô hình OSI).

IV. Mạng Arpanet

Đây là mạng được thiết lập tại Mỹ vào giữa những năm 60 khi bộ quốc phòng Mỹ muốn có một mạng dùng để ra lệnh và kiểm soát mà có khả năng sống còn cao trong trường hợp có chiến tranh hạt nhân. ấ hững mạng sử dụng đường điện thoại thông thường vào lúc đó tỏ ra không đủ an toàn khi mà một đường dây hay một tổng đài bị phá hủy cũng có thể dẫn đến mọi cuộc nói chuyện hay liên lạc thông qua nó bị gián đoạn, việc đó còn đôi khi dẫn đến cắt rời liên lạc.

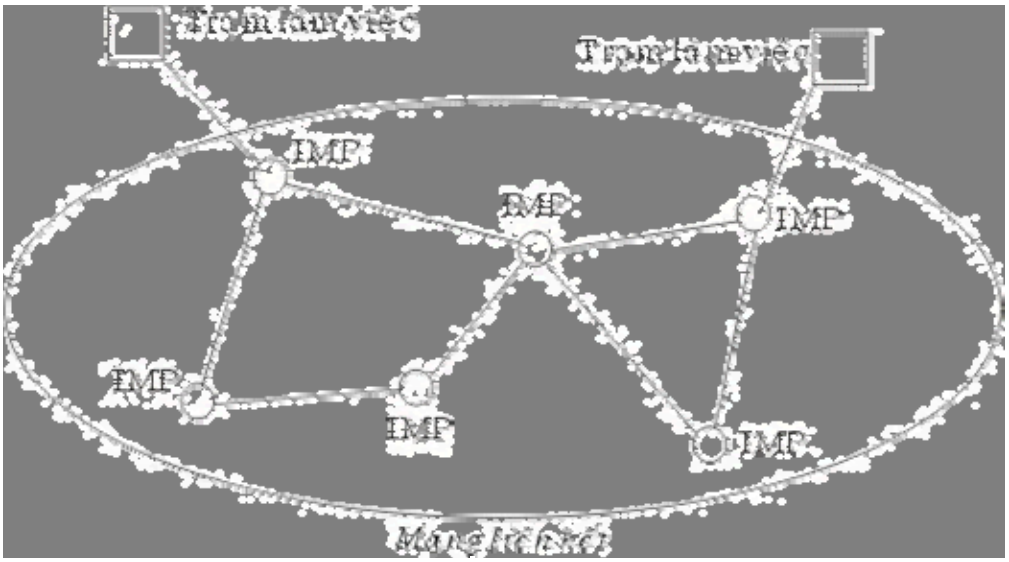
Để làm được điều này khi bộ quốc phòng Mỹ đưa ra chương trình ARPA (Advanced Research Projects Agency) với sự tham gia của nhiều trường đại học và công ty dưới sự quản lý của khi bộ quốc phòng Mỹ.

Vào đầu những năm 1960 những ý tưởng chủ yếu của chuyển mạch gói đã được Paul Baran công bố và sau khi tham khảo nhiều chuyên gia thì chương trình ARPA quyết định mạng tương lai của khi bộ quốc phòng Mỹ sẽ là mạng chuyển mạch gói và nó bao gồm một mạng liên kết và các trạm (host). Mạng liên kết bao gồm các máy tính dùng để liên kết các đường truyền dữ liệu được gọi là các điểm trung chuyển thôâng tin (IMP - Interface Message Processor).

Một IMP sẽ được liên kết với ít nhất là hai IMP khác với độ an toàn cao, các thông tin được chuyển trên mạng liên kết dưới dạng các gói dữ liệu tách rời, có nghĩa là khi có một số đường và nút bị phá hủy thì các gói tin tự động được chuyển theo những đường khác. Mỗi nút một máy tính của hệ thống bao gồm một trạm có được kết nối với một IMP trên mạng, nó gửi thông tin của mình đến IMP để rồi sau đó IMP sẽ phân gói, rồi lần lượt gửi các gói tin theo những đường mà nó lựa chọn để đến đích.

Tháng 10 năm 1968 ARPA quyết định lựa chọn hãng BBẩ một hãng tư vấn tại Cambridge, Massachsetts làm tổng thầu. Lúc đó BBẩ đã lựa chon máy DDP-316 làm IMP, các IMP được nối với đường thuê bao 56 Kbps từ các công ty điện thoại. Phần mềm được chia làm hai phần: phần liên kết mạng và phần cho nút, với phần mềm cho liên kết mạng bao gồm phần mềm tại các IMP đầu cuối và các IMP trung gian, các giao thức liên kết IMP với khả năng đảm bảo an toàn cao.

Phần mềm tại nút bao gồm phần mềm danh cho việc liên kết giữa nút với IMP, các giao thức giữa các nút với nhau trong quá trình truyền dữ liệu.



Hình 9.4: Cấu trúc ban đầu của mạng ARPANET

Vào tháng 10 năm 1969 mạng ARPAả ET bắt đầu được đưa vào hoạt động thử nghiệm với 4 nút là những trường đại học và trung tâm nghiên cứu tham gia chính vào dự án, mạng phát triển rất nhanh đến tháng 3 năm 1971 đã có 15 nút và tháng 9 năm 1972 đã có tới 35 nút. Các cải tiến tiếp theo cho phép nhiều trạm có thể liên kết với một IMP do vậy sẽ tiết kiệm tài nguyên và một trạm có thể liên kết với nhiều IMP nhằm tránh việc IMP hư hỏng làm gián đoạn liên lạc.

Cùng với việc phát triển các nút ARPA cũng dành ngân khoản cho phát triển các mạng truyền dữ liệu dùng kỹ thuật vệ tinh và dùng kỹ thuật radio. Điều đó cho phép thiết lập các nút tại những điểm các khoảng cách rất xa. Về các giao thức truyền thông thì sau khi thấy rằng các giao thức của mình không chạy được trên nhiều liên kết mạng vào năm 1974 ARPA đã đầu tư nghiên cứu hệ giao thức TCP/IP và dựa trên hợp đồng giữa BBả và Trường đại học tổng hợp Berkeley - California các nhà nghiên cứu của trường đại học đã viết rất nhiều phần mềm, chương trình quản trị trên cơ ở hệ điều hành Uả IX. Dựa trên các phần mềm mới về truyền thông trên cơ sở TCP/IP đã cho phép dễ dành liên kết các mạng LAả vào mạng ARPAả ET. Vào năm 1983 khi mạng đã hoạt động ổn định thì phần quốc phòng của mạng (gồm khoảng 160 IMP với 110 IMP tại nước Mỹ và 50 IMP ở nước ngoài, hàng trăm nút) được tách ra thành mạng MILả ETvà phần còn lại vẫn tiếp tục hoạt động như là một mạng nghiên cứu.

Trong những năm 1980 khi có nhiều mạng LAả được nối vào ARPAả ET để giảm việc tìm kiếm địa chỉ trên mạng người ta chia vùng các máy tính đưa tên các máy vào địa chỉ IP và xây dựng hệ quản trị cơ sở phân tán các tên các trạm của mạng Hệ cơ sở dữ liệu đó gọi là Dả S (Domain ả aming System) trong đó có chức mọi thông tin liên quan đến tên các trạm.

Vào năm 1990 với sự phát triển của nhiều mạng khác mà ARPAả ET là khởi xướng thì ARPAả ET đã kết thúc hoạt động của mình, tuy nhiên MILả ET vẫn hoạt động cho đến ngày nay.

V. Mạng NFSNET

Vào cuối những năm 1970 khi Quỹ khoa học quốc gia Hoa kỳ (ả FS - The U.S. ả ational Science Foundation) thấy được sự thu hút của ARPAả ET trong nghiên cứu khoa học mà qua đó các nhà khoa học có thể chia sẻ thông tin hay cùng nhau nghiên cứu các đề án. Tuy nhiên việc sử dụng ARPAả ET cần thông qua bộ quốc phòng Mỹ với nhiều hạn chế và nhiều cơ sở nghiên cứu khoa học không có khả năng đó. Điều đó khiến ả FS thiết lập một mạng ảo có tên là CSả ET trong đó sử dụng các máy tính tại công ty BBả cho phép các nhà nghiên cứu có thể kết nối vào để tiếp tục nối với mạng ARPAả ET hay gửi thư điện tử cho nhau. Vào năm 1984 ả FS bắt đầu nghiên cứu tới việc thiết lập một mạng tốc độ cao dành cho các nhóm nghiên cứu khoa học nhằm thay thế mạng ARPAả ET, bước đầu ả FS quyết định xây dựng được đường trục truyền số liệu nối 6 máy tính lớn (Supercomputer) tại 6 trung tâm máy tính. Tại mỗi trung tâm máy tính lớn tại đây được nối với một máy mini loại LSI-11 và các máy mini được nối với nhau bằng đường thuê bao 56 Kbps tương tự như kỹ thuật đã sử dụng ở mạng ARPAả ET. Đồng thời ả FS cũng cung cấp ngân khoản cho khoảng 20 mạng vùng để liên kết với các máy tính lớn trên và qua đó tới các máy tính lớn khác. Toàn bộ mạng bao gồm mạng trục và các mạng vùng được gọi là ả FSả ET, mạng ả FS có được kết nối với mạng ARPAả ET.

Mạng ả FS được phát triển rất nhanh, sau một thời gian hoạt động đường trục chính được thay thế bằng đường cáp quang 448 Kbps và các máy IBM RS6000 được sử dụng làm công việc kết nối. Đến năm 1990 đường trục đã được nâng lên đến 1.5 Mbps.

Với việc phát triển rất nhanh và ả FS thấy rằng chính quyền không có khả năng tiếp tục tài trợ nhưng do các công ty kinh doanh không thể sử dụng mạng ả FSả ET (do bin cấm theo luật) nên ả FS yểm trợ các công ty MERIT, MCI, IBM thành lập một công ty không sinh lợi (nonprofit corporation) có tên là Aả S (Advanced ả etworks and Services) nhằm phát triển việc kinh doanh hóa mạng. ASả tiếp nhận mạng ả FSả ET và bắt đầu nâng cấp đường trục lên từ 1.5 Mbps lên 45 Mbps để thành lập mạng Aả Sả ET.

Vào năm 1995 khi các công ty cung cấp dịch vụ liên kết phát triển khắp nơi thì mạng trục Aả Sả ET không còn cần thiết nữa và Aả Sả ET được bán cho công ty America Online. Hiện nay các mạng vùng của ả FS mua các dịch vụ truyền dữ liệu để liên kết với nhau, mạng ả FS đang sử dụng dịch vụ của 4 mạng truyền dữ liệu là PacBell, Ameritech, MFS, Sprint mà qua đó các mạng vùng ả FS có thể lựa chọn để kết nối với nhau.

VI. Mạng Internet

Cùng với sự phát triển của ả FSả ET và ARPAả ET nhất là khi giao thức TCP/IP đã trở thành giao thức chính thước duy nhất trên các mạng trên thì số lượng các mạng, nút muốn tham gia kết nối vào hai mạng trên đã tăng lên rất nhanh. Rất nhiều các mạng vùng được kết nối với nhau và còn liên kết với các mạng ở Canada, châu Aâu.

Vào khoảng giữa những năm 1980 người ta bắt đầu thấy được sự hình thành của một hệ thống liên mạng lớn mà sau này được gọi là Internet. Sự phát triển của Internet được tính theo cấp số nhân, nếu như năm 1990 có khoảng 200.000 máy tính với 3.000 mạng con thì năm 1992 đã có khoảng 1.000.000 máy tính được kết nối, đến năm 1995 đã có hàng trăm mạng cấp vùng, chục ngàn mạng con và nhiều triệu máy tính. Rất nhiều mạng lớn đang hoạt động cũng đã được kết nối vào Internet như các mạng SPAả , ả ASA network, HEPả ET, BITả ET, IBM network, EARả . Việc liên kết các mạng được thực hiện thông qua rất nhiều đường nối có tốc độ rất cao.

Hiện nay một máy tính được gọi là thành viên của Internet nếu máy tính đó có giao thức truyền dữ liệu TCP/IP, có một địa chỉ IP trên mạng và nó có thể gửi các gói tin IP đến tất cả các máy tính khác trên mạng Internet.

Tuy nhiên trong nhiều trường hợp thông qua một nhà cung cấp dịch vụ Internet người sử dụng kết nối máy của mình với máy chủ của nhà phục vụ và được cung cấp một địa chỉ tạm thời trước khi khai thác các tài nguyên của Internet. Máy tính của người đó có thể gửi các gói tin cho các máy khác bằng địa chỉ tạm thời đó và địa chỉ đó sẽ trả lại cho nhà cung cấp khi kết thúc liên lạc. Vì máy tính của người đó sử dụng trong thời gian liên kết với Internet cũng có một địa chỉ IP nên người ta vẫn coi máy tính đó là thành viên của Internet.

Vào năm 1992 cộng đồng Internet đã ra đời nhằm thúc đẩy sự phát triển của Internet và điều hành nó. Hiện nay Internet có 5 dịch vụ chính:

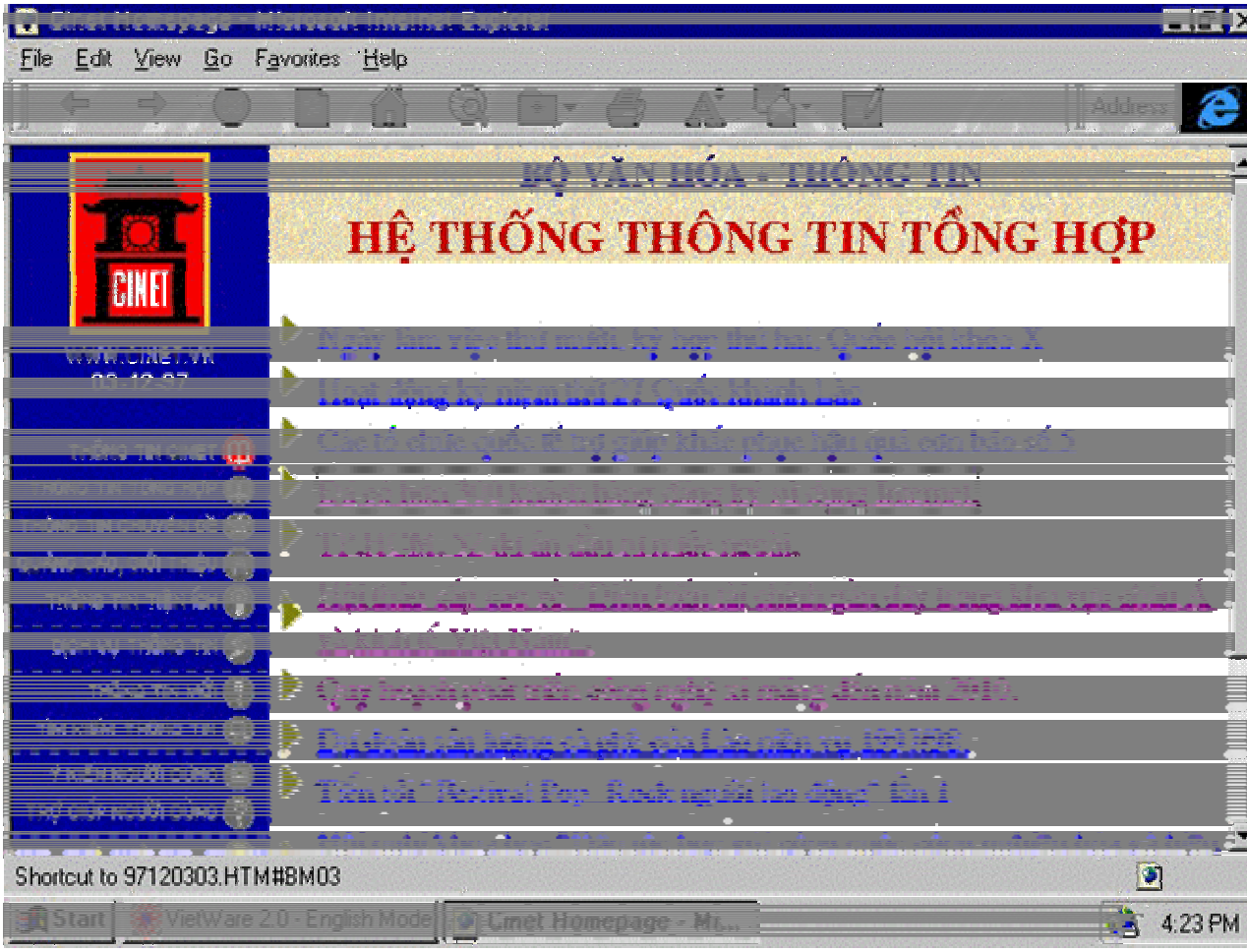
- **Thư điện tử** (Email): đây là dịch vụ đã có từ khi mạng ARPAả ET mới được thiết lập, nó cho phép gửi và nhận thư điến tử cho mọi thành viên khác trong mạng.

- **Thông tin mới** (ả ews): Các vân đề thời sự được chuyển thành các diễn đàn cho phép mọi người quan tâm có thể trao đổi các thông tin cho nhau, hiện nay hiện nay có hàng nghìn diễ đàn về mọi mặt trên Internet.

- **Đăng nhập từ xa** (Remote Login): Bằng các chương trình như Telnet, Rlogin người sử dụng có thể từ một trạm của Internet đăng nhập (logon) vào một trạm khác nếu như người đó được đăng ký trên máy tính kia.

- **Chuyển file** (File transfer): Bằng chương trình FTP người sử dụng có thể chép các file từ một máy tính trên mạng Internet tới một máy tính khác. ả gười ta có thể chép nhiều phần mềm, cơ sở dữ liệu, bài báo bằng cách trên.

- **Dịch vụ WWW** (World Wide Web): WWW là một dịch vụ đặc biệt cung cấp thông tin từ xa trên mạng Internet. Các tập tin siêu văn bản được lưu trữ trên máy chủ sẽ cung cấp các thông tin và dẫn đường trên mạng cho phép người sử dụng dễ dàng Truy cập các tập tin văn bản, đồ họa, âm thanh.



Hình 9.5: Ví dụ một trang Web cho phép dễ dàng khai thác các trang Web khác

ả gười sử dụng nhận được thông tin dưới dạng các trang văn bản, một trang là một đơn thể nằm trong máy chu. Đây là dịch vụ đang mang lại sức thu hút to lớn cho mạng Internet, chúng ta có thể xây dựng các trang Web bằng ngôn ngữ HTML (Hypertext Markup Language) với nhiều dạng phong phú như văn bản, hình vẽ, video, tiếng nói và có thể có

các kết nối với các trang Web khác. Khi các trang đó được đặt trên các máy chủ Web thì thông qua Internet người ta có thể xem được sự thể hiện của các trang Web trên và có thể xem các trang web khác mà nó chỉ đến.

Các phần mềm thông dụng được sử dụng hiện nay để xây dựng và duyệt các trang Web là Mosaic, ả avigator của ả etscape, Internet Explorer của Microsoft, Web Access của ả ovell.

www.viet-ebook.co.cc

Chương 10 :

Giới thiệu về hệ điều hành mạng Windows NT

I. Thế nào là một hệ điều hành mạng

Với việc ghép nối các máy tính thành mạng thì cần thiết phải có một hệ thống phần mềm có chức năng quản lý tài nguyên, tính toán và xử lý truy nhập một cách thống nhất trên mạng, hệ như vậy được gọi là hệ điều hành mạng. Mỗi tài nguyên của mạng như tệp, đĩa, thiết bị ngoại vi được quản lý bởi một tiến trình nhất định và hệ điều hành mạng điều khiển sự tương tác giữa các tiến trình và truy cập tới các tiến trình đó.

Căn cứ vào việc truy nhập tài nguyên trên mạng người ta chia các thực thể trong mạng thành hai loại chủ và khách, trong đó máy khách (Client) truy nhập được vào tài nguyên của mạng nhưng không chia sẻ tài nguyên của nó với mạng, còn máy chủ (Server) là máy tính nằm trên mạng và chia sẻ tài nguyên của nó với các người dùng mạng.

Hiện nay các hệ điều hành mạng thường được chia làm hai loại là hệ điều hành mạng ngang hàng (Peer-to-peer) và hệ điều hành mạng phân biệt (client/server).

Với hệ điều hành mạng ngang hàng mỗi máy tính trên mạng có thể vừa đóng vai trò chủ lẫn khách tức là chúng vừa có thể sử dụng tài nguyên của mạng lẫn chia sẻ tài nguyên của nó cho mạng, ví dụ: LAå tastic của Artisoft, å etWare lite của å ovell, Windows (for Workgroup, 95, å T Client) của Microsoft.

Với hệ điều hành mạng phân biệt các máy tính được phân biệt chủ và khách, trong đó máy chủ mạng (Server) giữ vai trò chủ và các máy cho người sử dụng giữ vai trò khách (các trạm). Khi có nhu cầu truy nhập tài nguyên trên mạng các trạm tạo ra các yêu cầu và gửi chúng tới máy chủ sau đó máy chủ thực hiện và gửi trả lời. Ví dụ các hệ điều hành mạng phân biệt: å ovell å etware, LAå Manager của Microsoft, Windows å T Server của Microsoft, LAå Server của IBM, Vines của Banyan System với server dùng hệ điều hành Unix.

II. Hệ điều hành mạng Windows NT

Windows å T là hệ điều hành mạng cao cấp của hãng Microsoft. Phiên bản đầu có tên là Windows å T 3.1 phát hành năm 1993, và phiên bản server là Windows å T Advanced Server (trước đó là LAå Manager for å T). å ăm 1994 phiên bản Windows å T Server và Windows å T Workstation version 3.5 được phát hành. Tiếp theo đó ra đời các bản version 3.51. Các phiên bản workstation có sử dụng để thành lập mạng ngang hàng; còn các bản server dành cho quản lý file tập trung, in ấn và chia sẻ các ứng dụng.

å ăm 1995, Windows å T Workstation và Windows å T Server version 4.0 ra đời đã kết hợp shell của người anh em Windows 95 nổi tiếng phát hành trước đó không lâu (trước đây shell của Windows å T giống shell của Windows 3.1) đã kết hợp được giao diện quen thuộc, dễ sử dụng của Windows 95 và sự mạnh mẽ, an toàn, bảo mật cao của Windows å T.

Windows å T có hai bản mà nó đi đôi với hai cách tiếp cận mạng khác nhau. Hai bản này gọi là Windows å T Workstation và Windows å T server. Với hệ điều hành chuẩn của å T ta có thể xây dựng mạng ngang hàng, máy chủ mạng và mọi công cụ quản trị cần thiết cho

một máy chủ mạng ngoài ra còn có thể có nhiều giải pháp về xây dựng mạng diện rộng. Cả hai bản Windows ả T station và Windows ả T server cùng được xây dựng trên cơ sở nhân ả T chung và các giao diện và cả hai cùng có những đặc trưng an toàn theo tiêu chuẩn C2. Windows ả T Wordstation được sử dụng để kết nối những nhóm người sử dụng nhỏ, thường cùng làm việc trong một văn phòng. Tuy nhiên với Windows ả T server ta có được một khả năng chống hỏng hóc cao, những khả năng cung cấp dịch vụ mạng lớn và những lựa chon kết nối khác nhau, Windows ả T Server không hạn chế về số người có thể thâm nhập vào mạng.

Với Windows ả T ta cũng có những công cụ quản trị từ xa vào mạng mà có thể thực hiện được việc quản trị từ những máy tính ở xa. ả ó thích hợp với tất cả các sơ đồ mạng BUS, STAR, RIả G và hỗn hợp.

Windows ả T là hệ điều hành có sức mạnh công nghiệp đầu tiên cho số lượng khổng lồ các máy tính IBM compatible. Windows ả T là một hệ điều hành thực sự dành cho người sử dụng, các cơ quan, các công ty xí nghiệp. Windows ả T là một hệ điều hành đa nhiệm, đa xử lý với địa chỉ 32 bit bộ nhớ. ả ó yểm trợ các ứng dụng DOS, Windows, Win32 GUI và các ứng dụng dựa trên ký tự. Windows ả T server là một hệ điều hành mạng hoàn chỉnh, nó nhanh chóng được thừa nhận là một trong những hệ điều hành tốt nhất hiện nay vì:

- Là hệ điều hành mạng đáp ứng tất cả các giao thức truyền thông phổ dụng nhất. ả goài ra nó vừa cho phép giao lưu giữa các máy trong mạng, vừa cho phép truy nhập từ xa, cho phép truyền file v.v... Windows ả T là hệ điều hành vừa đáp ứng cho mạng cục bộ (LAả) vừa đáp ứng cho mạng diện rộng (WAả) như Intranet, Internet.

- Windows ả T server hơn hẳn các hệ điều hành khác bởi tính mềm dẻo, đa dạng trong quản lý. ả ó vừa cho phép quản lý mạng theo mô hình mạng phân biệt (Clien/Server), vừa cho phép quản lý theo mô hình mạng ngang hàng (peer to peer).

- Windows ả T server đáp ứng tốt nhất các dịch vụ viễn thông, một dịch vụ được sử dụng rộng rãi trong tương lai.

- Windows ả T server cài đặt đơn giản, nhẹ nhàng và điều quan trọng nhất là nó tương thích với hầu như tất cả các hệ mạng, nó không đòi hỏi người ta phải thay đổi những gì đã có.

- Cho phép dùng các dịch vụ truy cập từ xa (Remote access service - RAS), có khả năng phụp vụ đến 64 cổng truy nhập từ xa (trong đó Lan manager 16 cổng).

- Đáp ứng cho cả các máy trạm Macintosh nối với Windows ả T server.

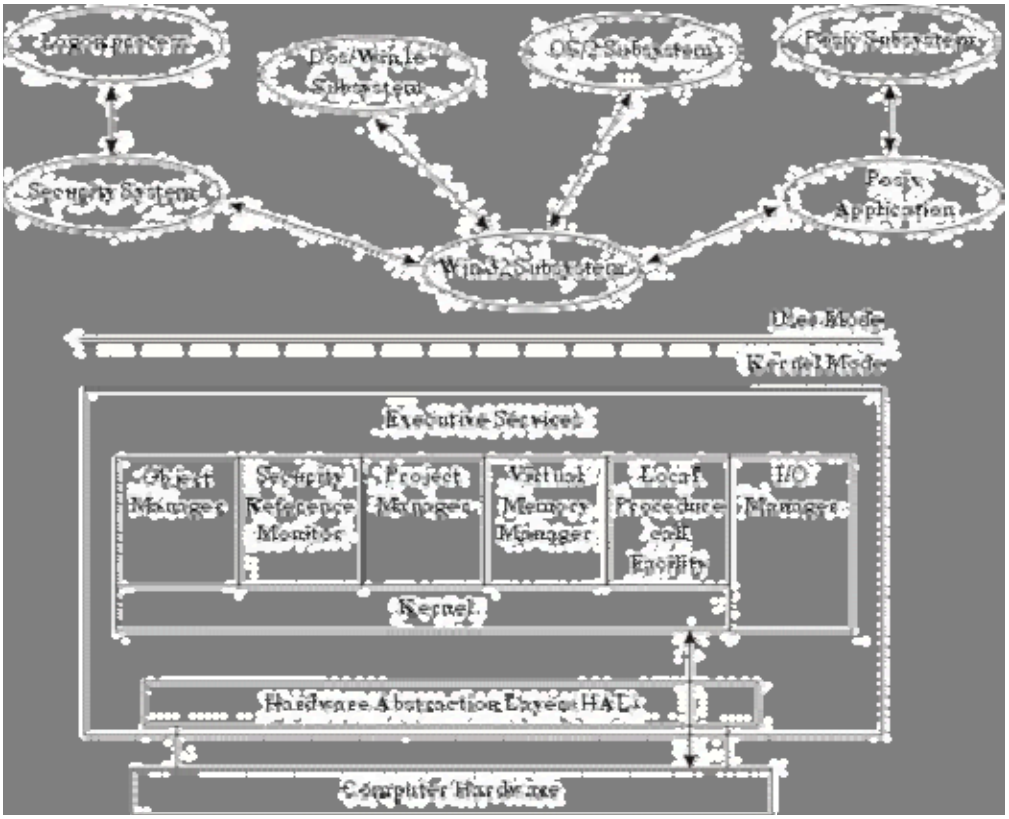
Windows ả T yểm trợ mọi nghi thức mạng chuẩn như ả etBUEI, IPX/SPX, TCP/IP và các nghi thức khác. Windows ả T cũng tương thích với những mạng thông dụng hiên nay như ả ovell ả etWare, Banyan VIả ES, và Microsoft LAả Manager. Đối với mạng lớn và khả năng thâm nhập từ xa sản phẩm Windows ả T Server cũng cũng cấp các chức năng bổ xung nhu khả năng kết nối với máy tính lớn và máy MAC.

III. Cấu trúc của hệ điều hành Windows NT

Windows ả T được thiết kế sử dụng cách tiếp cận theo đơn thể (modular). Các đơn thể khác nhau (còn được gọi là các bộ phận, thành phần) của Windows ả T được trình bày trong hình 1 Các bộ phận của Windows ả T có thể chạy dưới hai chế độ: User (người sử dụng) và Kernel (cốt lõi của hệ điều hành). Khi một thành phần của hệ điều hành chạy dưới cốt lõi của hệ điều hành (Kernel), nó truy cập đầy đủ các chỉ thị máy cho bộ xử lý đó và có thể truy cập tổng quát toàn bộ tài nguyên trên hệ thống máy tính.

Trong Windows ả T: Executive Services, Kernel và HAL chạy dưới chế độ cốt lõi của hệ điều hành.

Hệ thống con (Subsystem) Win 32 và các hệ thống con về môi trường, chẳng hạn như DOS/Win 16.0S/2 và hệ thống con POSIX chạy dưới chế độ user. Bằng cách đặt các hệ thống con này trong chế độ user, các nhà thiết kế Windows ả T có thể hiệu chỉnh chúng dễ dàng hơn mà không cần thay đổi các thành phần được thiết kế để chạy dưới chế độ Kernel.



Hình 10.1: Cấu trúc Windows NT

Các lớp chính của hệ điều hành WINDOWS NT SERVER gồm:

- ***Lớp phần cứng trừu tượng (Hardware Astraction Layer - HAL):*** Là phần cứng máy tính mà cốt lõi của hệ điều hành (Kernel) có thể đươc ghi vào giao diện phần cứng ảo, thay vì vào phần cứng máy tính thực sự. Phần lớn cốt lõi của hệ điều hành sử dụng HAL để truy cập các tài nguyên máy tính. Điều này có nghĩa là cốt lõi của hệ điều hành và tất cả các thành phần khác phụ thuộc vào cốt lõi có thể dễ dàng xuất (Ported) thông qua Microsoft đến các nền (Platform) phần cứng khác. Một thành phân nhỏ trong cốt lõi của hệ điều hành, cũng như bộ quản lý ả hập / Xuất truy cập phần cứng máy tính trực tiếp mà không cần bao gồm HAL.

- ***Lớp Kernel cốt lõi của hệ điều hành):*** Cung cấp các chức năng hệ điều hành cơ bản được sử dụng bởi các thành phần thực thi khác. Thành phần Kernel tương đối nhỏ và cung cấp các thành phần cốt yếu cho những chức năng của hệ điều hành. Kernel chủ yếu chịu trách nhiệm quản lý luồng, quản lý phần cứng và đồng bộ đa sử lý.

- **Các thành phần Executive:** Là các thành phần hệ điều hành ở chế độ Kernel thi hành các dịch vụ như :

    - Quản lý đối tượng (object manager)
    - Bảo mật (security reference monitor)
    - Quản lý tiến trình (process manager)
    - Quản lý bộ nhớ ảo (virtual memory manager)
    - Thủ tục cục bộ gọi tiện ích, và quản trị nhập/xuất (I/O Manager)

IV.Cơ chế quản lý của Windows NT

1. Quản lý đối tượng (Object Manager):

Tất cả tài nguyên của hệ điều hành được thực thi như các đối tượng. Một đối tượng là một đại diện trừu tượng của một tài nguyên. ả ó mô tả trạng thái bên trong và các tham số của tài nguyên và tập hợp các phương thức (method) có thể được sử dụng để truy cập và điều khiển đối tượng.

Ví dụ một đối tượng tập tin sẽ có một tên tập tin, thông tin trạng thái trên file và danh sách các phương thức, như tạo, mở,đóng và xóa, đối tượng mô tả các thao tác có thể được thực hiện trên đối tượng file.

Bằng cách xử lý toàn bộ tài nguyên như đối tượng Windows ả T có thể thực hiện các phương thức giống nhau như: tạo đối tượng, bảo vệ đối tượng, giám sát việc sử dụng đối tượng (Client object) giám sát những tài nguyên được sử dụng bởi một đối tượng.

Việc quản lý đối tượng (Object Manager) cung cấp một hệ thống đặt tên phân cấp cho tất cả các đối tượng trong hệ thống. Do đó, tên đối tượng tồn tại như một phần của không gian tên toàn cục và được sử dụng để theo dõi việc tạo và sử dụng đối tượng.

Sau đây là một số ví dụ của loại đối tượng Windows ả T :

- Đối tượng Directory (thư mục).
- Đối tượng File (tập tin).
- Đối tượng kiểu object.
- Đối tượng Process (tiến trình).

- Đối tượng thread (luồng).
- Đối tượng Section and segment (mô tả bộ nhớ).
- Đối tượng Port (cổng).
- Đối tượng Semaphore và biến cố.
- Đối tượng liên kết Symbolic (ký hiệu).

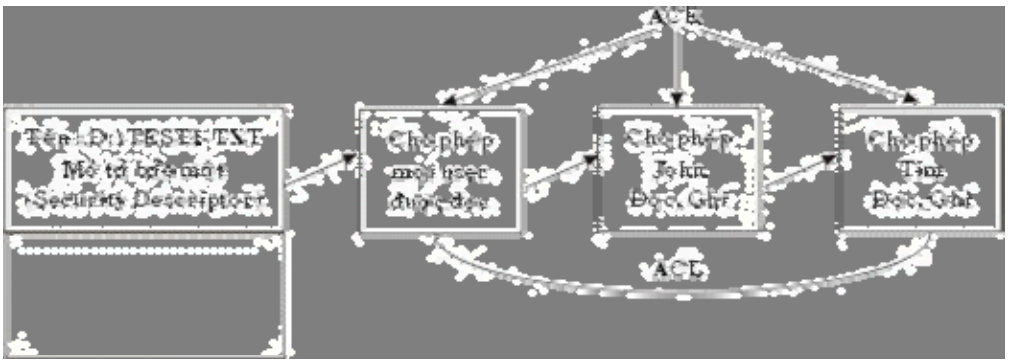
2. Cơ chế bảo mật (SRM - Security Reference Monitor):

Được sử dụng để thực hiện vấn đề an ninh trong hệ thống Windows ả T. Các yêu cầu tạo một đối tượng phải được chuyển qua SRM để quyết định việc truy cập tài nguyên được cho phép hay không. SRM làm việc với hệ thống con bảo mật trong chế độ user. Hệ thống con này được sử dụng để xác nhận user login vào hệ thống Windows ả T.

Để kiểm soát việc truy cập, mỗi đối tượng Windows ả T có một danh sách an toàn (Access Control List - ACL). Danh sách an toàn của mỗi đối tượng gồm những phần tử riêng biệt gọi là Access Control Entry (ACE). Mỗi ACE chứa một SecurityID (SID: số hiệu an toàn) của người sử dụng hoặc nhóm. Một SID là một số bên trong sử dụng với máy tính Windows ả T mô tả một người sử dụng hoặc một nhóm duy nhất giữa các máy tính Windows ả T.

ả goài SID, ACE chứa một danh sách các hành động (action) được cho phép hoặc bị từ chối của một user hoặc một nhóm. Khi người sử dụng đăng nhập vào mạng Windows ả T, sau khi việc nhận dạng thành công, một Security Access Token (SAT) được tạo cho người dùng đó. SAT chứa SID của người dùng và SID của tất cả các nhóm người dùng thuộc mạng Windows ả T. Sau đó SAT hoạt động như một "passcard" (thẻ chuyển) cho phiên làm việc của người dùng đó và được sử dụng để kiểm tra tất cả hoạt động của người dùng.

Khi người dùng tham gia mạng truy cập một đối tượng, Security Reference Monitor kiểm tra bộ mô tả bảo mật của đối tượng xem SID liệt kê trong SAT có phù hợp với giá trị trong ACE không. ả ếu phù hợp, các quyền về an ninh được liệt trong ACE áp dụng cho người dùng đó.



Hình 10.2: Ví dụ về danh sách an toàn (Access Control List).

3. Quản lý nhập / xuất (I/O Manager) :

Chịu trách nhiệm cho toàn bộ các chức năng nhập / xuất trong hệ điều hành Windows ả T. I/O Manager liên lạc với trình điều khiển của các thiết bị khác nhau.

4. I/O *Manager:*

Ssử dụng một kiến trúc lớp cho các trình điều khiển. Mỗi bộ phận điều khiển trong lớp này thực hiện một chức năng được xác định rõ. Phương pháp tiếp cận này cho phép một thành phần điều khiển được thay thế dễ dàng mà không ảnh hưởng phần còn lại của các bộ phận điều khiển.

Hình 10.3:Các trình điều khiển thiết bị theo lớp của I / O Manager

V. Các cơ chế bảo vệ dữ liệu trong Windows NT

Cơ chế bảo vệ dữ liệu của Windows ả T gọi là fault tolerance, nó cho phép hệ thống khả năng tiếp tục làm việc và bảo toàn dữ liệu của hệ thống trong trường hợp một phần của hệ thống có sự cố hỏng hóc sai lệch. Trong Windows ả T cơ chế fault tolerance bao gồm các biện pháp sau:

- Chống cúp điện bất thường.
- Cung cấp khả năng bảo vệ hệ thống đĩa (fault tolerance disk subsystem).
- Cung cấp khả năng sao chép dự phòng (backup) từ băng từ.

Khả năng bảo vệ hệ thống đĩa của Windows ả T là RAID ø (viết tắt của Redundant Array of Inexpensiredisk). Thực chất RAID là một loạt các biện pháp để bảo vệ hệ thống đĩa. Các biện pháp trong RIAD được chia thành 6 mức sau:

- ***Mức 0*:** Đây là mức ứng với biện pháp chia nhỏ đĩa (disk striping). Thực chất nội dung của biện pháp này là phân chia dữ liệu thành khối và sau đó sắp xếp các khối dữ liệu theo thứ tự trong tất cả các đĩa thành 1 mảng.
- ***Mức 1*:** Mức này ứng với biện pháp disk Mirroring, biện pháp này cho phép tạo ra 2 đĩa giống nhau. ả ếu trong quá trình vận hành mạng một đĩa có sự cố thì hệ thống sử dụng dữ liệu của đĩa kia.
- ***Mức 2*:** Mức này ứng với biện pháp phân chia nhỏ đĩa bằng cách phân chia các file thành các byte và sắp xếp các byte sang nhiều đĩa. Mức này sử dụng mã sửa sai

(error correcting code) trong quá trình phân chia đĩa. ả ói chung biện pháp dùng ở mức này tốt hơn biện pháp dùng trong mức 1.

- ***Mức 3*:** Mức này sử dụng biện pháp giống mức 2. Tuy nhiên mã sửa sai (error correction code) chỉ sử dụng cho một đĩa. Không áp dụng cho nhiều đĩa như ở mức 2. ả gười ta thường dùng mức này để truy nhập vào một số ít file có dung tích lớn.

- ***Mức 4*:** Mức này sử dụng biện pháp giống ở mức 2 và 3 nhưng bằng phương pháp phân chia đĩa thành các khối lớn. Giống như mức 3 tất cả các mã sửa sai (error correction code) được hgi vào một đĩa và tách khỏi khối dữ liệu.

- ***Mức 5*:** Trong mức này người ta sử dụng biện pháp phân chia đĩa thành từng phần gọi là Striping with party. Biện pháp sử dụng ở mức này tương tự như ức 4, số liệu được phân nhỏ thành các khối lớn và sau đó ghi vào tất cả các đĩa. Các thông tin (party Infomation) được coi như các dữ liệu dùng tạm thời (data redundancy).

Ngoài ra chúng ta còn có thể áp dụng các biện pháp bảo vệ dữ liệu trong Windows NT:

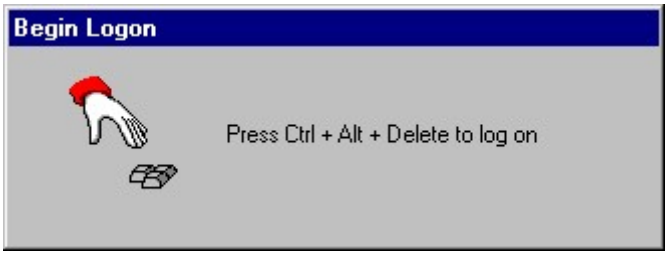
- ***Biện pháp Disk mirroring:*** Disk mirroring là cách sao tạm (redundant) lại đĩa hoặc partition. Biện pháp này bảo vệ dữ liệu tránh các sự cố bằng cách dưa ra chế độ thường xuyên backup đĩa hoặc partition. Hình dưới chỉ ra cách dùng biện pháp Mirroring:

- ***Disk Duplexing:*** Biện pháp dùng đĩa kép (Disk Duplexing) tường tự như disk mirroring chỉ khác là chúng dùng 2 disk controler. Diều này cho thên hả năng bảo vệ khi controler của một đĩa có sự cố. Trong khi đó biện pháp Mirror không thể khắc phục được tình huống này.

- ***Mirror Set:*** Các partition hoặc đĩa trong chế độ Mirror được tạo ra bằng cách lặp sao lại partition hoặc đĩa trên đĩa khác cùng một tên ổ đĩa được gán cho cả 2 partition. Ta có thể dùng establish Mirror trong menu Fault tolerance. ả ếu đĩa hoặc partition trong chế độ Mirror bị lỗi thì chế độ Mirror cần phải ngắt để thực hiện chế độ sao chép dự phòng vào một đĩa riệng. Sau đó sao backup trở lại.

VI. Giới thiệu về hoạt động của Windows NT Server

Khi chúng ta khởi động Windows ả T Server hộp Begin logon sẽ hiện ra, server chờ đợi để chúng ta bấm Ctrl+Alt +Del để có thể tiếp tục hoạt động. Ở đây có điểm khác với các hệ điều hành DOS, Windows 95 là tổ hợp Ctrl+Alt +Del không phải là khởi động lại máy. Trong trường hợp này Windows ả T loại bỏ mọi chương trình Virus hay không có phép đang hoạt động trước khi bước vào làm việc.



Hình 10.4: Thông báo gia nhập mạng

Lúc này chúng ta sẽ thấy hộp Logon Information xuất hiện và yếu cầu chúng ta phải đánh đúng tên và mật khẩu thì mới được đăng nhập vào Server. ả ếu là người dùng mới thì phải được người quản trị khai báo tên và mật khẩu trước khi đăng nhập..

Hình 10.5: Màn hình gia nhập mạng

Cũng giống như màn hình nền của hệ điều hành Windows 95 khi muốn thực hiện các trình, gọi các menu hệ thống chúng ta dùng nút Start ở cuối màn hình

Hình 10.6: Điểm khởi đầu của Windows

Trước muốn kết thúc chương trình và tắt máy chúng ta phải bấm phím Start rồi chọn ShutDown, màn hình kết thúc sẽ hiện ra cho chúng ta lựa chon công yêu cầu về tắt hay khởi động lại.

Hình 10.7: Màn hình thoát khỏi Windows

Chương 11

Hệ thống quản lý của mạng Windows NT

Các mạng máy tính hiện nay được thiết kế rất đa dạng và đang thực hiện những ứng dụng trên nhiều lĩnh vực của đời sống xã hội. Điều đó có nghĩa là các thông tin lưu trữ trên mạng và các thông tin truyền giao trên mạng ngày càng mang nhiều giá trị có ý nghĩa sống còn. Do vậy những người quản trị mạng ngày càng phải quan tâm đến việc bảo vệ các tài nguyên của mình.

Việc bảo vệ an toàn là quá trình bảo vệ mạng khỏi bị xâm nhập hoặc mất mát, khi thiết kế các hệ điều hành mạng người ta phải xây dựng một hệ thống quản lý nhiều tầng và linh hoạt giúp cho người quản trị mạng có thể thực hiện những phương án về quản lý từ đơn giản mức độ thấp cho đến phức tạp mức độ cao trong những mạng có nhiều người tham gia. Thông qua những công cụ quản trị đã được xây dựng saün người quản trị có thể xây dựng những cơ chế về an toàn phù hợp với cơ quan của mình.

Thông thường hệ thống mạng có những mức quản lý chính sau:

- **Mức quản lý việc thâm nhập mạng (Login/Password):** Mức quản lý việc thâm nhập mạng (Login/Password) xác định những ai và lúc nào có thể vào mạng. Đối với người quản trị và người sử dụng mạng, mức an toàn này dường như khá đơn giản mà theo đó mỗi người sử dụng (người sử dụng) có một tên login và mật khẩu duy nhất.

- **Mức quản lý trong việc quản lý sử dụng các tài nguyên của mạng:** Kiểm soát những tài nguyên nào mà người sử dụng được phép truy cập, sử dụng và sử dụng như thế nào.

- **Mức quản lý với thư mục và file:** Mức an toàn của file kiểm soát những file và thư mục nào người sử dụng được dùng trên mạng và được sử dụng ở mức độ nào

- **Mức quản lý việc điều khiển File Server:** Mức an toàn trên máy chủ kiểm soát ai có thể được thực hiện các thao tác trên máy chủ như bật, tắt, chạy các chương trình khác. ả gười ta cần có cơ chế như mật khẩu để bảo vệ.

I. Quản lý các tài nguyên trong mạng

ả hư chúng ta đã biết, mạng LAả cung cấp các dịch vụ theo hai cách: qua cách chia sẻ tài nguyên theo nguyên tắc ngang hàng và thông qua những máy chủ trung tâm. Dù bất cứ phương pháp nào được sử dụng, vấn đề cần phải giải quyết là là giúp người sử dụng xác định được các tài nguyên có saün ở đâu để có thể sử dụng.

Các kỹ thuật sau đây đã được sử dụng để tổ chức tài nguyên mạng máy tính:

- Quản lý đơn lẻ từng máy chủ (stand-alone services).

- Quản lý theo dịch vụ thư mục (directory services).

- Quản lý theo nhóm (workgroups).

- Quản lý theo domain (domains).

1. Quản lý đơn lẻ từng máy chủ (Stand-alone Services)

Với cách quản lý này trong mạng LAå thường chỉ có một vài máy chủ, mỗi máy chủ sẽ quản lý tài nguyên của mình, mỗi người sử dụng muốn thâm nhập những tài nguyên của máy chủ nào thì phải khai báo và chịu sự quản lý của máy chủ đó. Mô hình trên phù hợp với những mạng nhỏ với ít máy chủ và khi có trục trặc trên một máy chủ thì toàn mạng vẫn hoạt động. Cũng vì trong mạng LAå chỉ có ít máy chủ, do đó người sử dụng không mấy khó khăn để tìm các tập tin, máy in và các tài nguyên khác của mạng (plotter, CDRom, modem...).

Việc tổ chức như vậy không cần những dịch vụ quản lý tài nguyên phức tạp. Tuy nhiên khi trong mạng có từ hai máy chủ trở lên vấn đề trở nên phức tạp hơn vì mỗi máy chủ riêng lẻ giữ riêng bảng danh sách các người sử dụng và tài nguyên của mình. Khi đó mỗi người sử dụng phải tạo lập và bảo trì tài khoản của mình ở hai máy chủ khác nhau mới có thể đăng nhập (logon) và truy xuất đến các máy chủ này. å goài ra việc xác định vị trí của các tài nguyên trong mạng cũng rất khó khăn khi mạng có qui mô lớn.

2. Quản lý theo dịch vụ thư mục (Directory Services)

Hệ thống các dịch vụ thư mục cho phép làm việc với mạng như là một hệ thống thống nhất, tài nguyên mạng được nhóm lại một cách logic để dễ tìm hơn. Giải pháp này có thể được dùng cho những mạng lớn. Ở đây thay vì phải đăng nhập vào nhiều máy chủ, người sử dụng chỉ cần đăng nhập vào mạng và được các dịch vụ thư mục cấp quyền truy cập đến tài nguyên mạng, cho dù được cung cấp bởi bất kể máy chủ nào.

å gười quản trị mạng chỉ cần thực hiện công việc của mình tại một trạm trên mạng mặc dù các điểm nút của nó có thể nằm trên cả thế giới. Hệ điều hành å etware 4.x cung cấp dịch vụ nổi tiếngï và đầy ưu thế cạnh tranh này với tên gọi ***Netware Directory Services (NDS).***

Giải pháp này thích hợp với những mạng lớn. Các thông tin của å DS được đặt trong một hệ thống cơ sở dữ liệu đồng bộ, rộng khắp được gọi là DIB (Data Information Base). Cơ sở dữ liệu trên quản lý các dữ liệu dưới dạng các đối tượng phân biệt trên toàn mạng. Các định nghĩa đối tượng sẽ được đặt trên các tập tin riêng của một số máy chủ đặc biệt, mỗi đối tượng có các tính chất và giá trị của mỗi tính chất. Đối tượng bao hàm tất cả những gì có tên phân biệt như å gười sử dụng, File server, Print server, group. Mỗi loại đối tượng có những tính chất khác nhau ví dụ như đối tượng å gười sử dụng có tính chất về nhóm mà người sử dụng đó thuộc, còn nhóm có các tính chất về người sử dụng mà nhóm đó chứa.

Việc thiết lập các dịch vụ như vậy cần được lập kế hoạch, thiết kế rất cẩn thận, liên quan đến tất cả các đơn vị phòng ban có liên quan. Loại mạng này có khuyết điểm là việc thiết kế, thiết lập mạng rất phức tạp, mất nhiều thời gian nên không thích hợp cho các mạng nhỏ.

3. Quản lý theo nhóm (Workgroup)

Các nhóm làm việc làm việc theo ý tưởng ngược lại với các dịch vụ thư mục. å hóm làm việc dựa trên nguyên tắc mạng ngang hàng (peer-to-peer network), các người sử dụng chia sẻ tài nguyên trên máy tính của mình với những người khác, máy nào cũng vừa là chủ

(server) vừa là khách (client). ả gười sử dụngĩ có thể cho phép các người sử dụng khác sử dụng tập tin, máy in, modem... của mình, và đến lượt mình có thể sử dụng các tài nguyên được các người sử dụng khác chia sẻ trên mạng. Mỗi cá nhân người sử dụng quản lý việc chia sẻ tài nguyên trên máy của mình bằng cách xác định cái gì sẽ được chia sẻ và ai sẽ có quyền truy cập. Mạng này hoạt động đơn giản: sau khi logon vào, người sử dụng có thể duyệt (browse) để tìm các tài nguyên có saün trên mạng.

Workgroup là nhóm logic các máy tính và các tài nguyên của chúng nối với nhau trên mạng mà các máy tính trong cùng một nhóm có thể cung cấp tài nguyên cho nhau. Mỗi máy tính trong một workgroup duy trì chính sách bảo mật và CSDL quản lý tài khoản bảo mật SAM (Security Account Manager) riêng ở mỗi máy. Do đó quản trị workgroup bao gồm việc quản trị CSDL tài khoản bảo mật trên mỗi máy tính một cách riêng lẻ, mang tính cục bộ, phân tán. Điều này rõ ràng rất phiền phức và có thể không thể làm được đối với một mạng rất lớn.

ả hưng workgroup cũng có điểm là đơn giản, tiện lợi và chia sẽ tài nguyên hiệu quả, do đó thích hợp với các mạng nhỏ, gồm các nhóm người sử dụng tương tự nhau.

Tuy nhiên Workgroup dựa trên cơ sở mạng ngang hàng (peer-to-peer), nên có hai trở ngại đối với các mạng lớn như sau:

- Đối với mạng lớn, có quá nhiều tài nguyên có saün trên mạng làm cho các người sử dụng khó xác định chúng để khai thác.

- ả gười sử dụng muốn chia sẻ tài nguyên thường sử dụng một cách dễ hơn để chia sẻ tài nguyên chỉ với một số hạn chế người sử dụng khác.

Điển hình cho loại mạng này là Windws for Workgroups, LAả tastic, LAả Manager... Window 95, Windows ả T Workstation.

4. Quản lý theo vùng (Domain)

Domain mượn ý tưởng từ thư mục và nhóm làm việc. Giống như một workgroup, domain có thể được quản trị bằng hỗn hợp các biện pháp quản lý tập trung và địa phương. Domain là một tập hợp các máy tính dùng chung một nguyên tắc bảo mật và CSDL tài khoản người dùng (người sử dụng account). ả hững tài khoản người dùng và nguyên tắc an toàn có thể được nhìn thấy khi thuộc vào một CSDL chung và được tập trung.

Giống như một thư mục, một domain tổ chức tài nguyên của một vài máy chủ vào một cơ cấu quản trị. ả gười sử dụng được cấp quyền logon vào domain chứ không phải vào từng máy chủ riêng lẻ. ả goài ra, vì domain điều khiển tài nguyên của một số máy chủ, nên việc quản lý các tài khoản của người sử dụng được tập trung và do đó trở nên dễ dàng hơn là phải quản lý một mạng với nhiều máy chủ độc lập.

Các máy chủ trong một domain cung cấp dịch vụ cho các người sử dụng. Một người sử dụng khi logon vào domain thì có thể truy cập đến tất cả tài nguyên thuộc domain mà họ được cấp quyền truy cập. Họ có thể dò tìm (browse) các tài nguyên của domain giống như trong một workgroup, nhưng nó an toàn, bảo mật hơn.

Để xây dựng mạng dựa trên domain, ta phải có ít nhất một máy Windows ả T Server trên mạng. Một máy tính Windows ả T có thể thuộc vào một workgroup hoặc một domain, nhưng không thể đồng thời thuộc cả hai. Mô hình domain được thiết lập cho các mạng lớn với khả năng kết nối các mạng toàn xí nghiệp hay liên kết các kết nối mạng với các mạng khác và những công cụ cần thiết để điều hành.

Việc nhóm những người sử dụng mạng và tài nguyên trên mạng thành domain có lợi ích sau:

- Mã số của người sử dụng được quản lý tập trung ở một nơi trong một cơ sở dữ liệu của máy chủ, do vậy quản lý chặt chẽ hơn.
- Các nguồn tài nguyên cục bộ được nhóm vào trong một domain nên dễ khai thác hơn.

Quản lý theo Workgroup và domain là hai mô hình mà Windows NT lựa chọn. Sự khác nhau căn bản giữa Workgroup và domain là trong một domain phải có ít nhất một máy chủ (máy chủ) và tài nguyên người sử dụng phải được quản lý bởi máy chủ đó.

II. Hệ thống quản lý trên Hệ điều hành mạng Windows NT Server

Windows ả T cung cấp những chức năng tuân theo chuẩn C2 (chuẩn về an toàn quốc tế) trong đó Windows ả T đảm bảo tránh được những người không được phép vào trong hệ thống hoặc thâm nhập vào các file và chương trình trên đĩa cứng. ả gười ta không thể thâm nhập vào được nếu không có mật khẩu đúng. và qua đó đã bảo vệ được các file. Windows ả T cung cấp công cụ để xây dựng các lớp quyền dành cho nhiều nhiệm vụ khác nhau nhằm xây dựng hệ thống an toàn một cách mềm dẻo.

ả hiều người sử dụng có thể có quyền vào một máy chủ Windows ả T. Một tài khoản của người sử dụng trên máy bao gồm tên, mật khẩu và nhiều tính chất được cho bởi người quản trị mạng. ả gười sử dụng có thể che các thư mục hay file của mình từ những người khác và cài đặt các thông số của File manager, Programe Manager, Control Panel một cách phù hợp.

Khi người dùng thâm nhập vào hệ thống thì tự động khởi động mọi thông số đã được lưu trữ từ trước. ả ếu người sử dụng có quyền cao hơn thì họ có thể chia sẻ hoặc ngừng các tài nguyên đang dùng chung trên mạng như máy in hay file hoặc họ có thể thay đổi quyền của những người dùng mạng khác khi thâm nhập vào mạng.

1. Mô hình Workgroup (nhóm) của mạng Windows NT

Mỗi người truy cập vào mạng Windows ả T tổ chức theo mô hình Workgroup cần phải đăng ký:

- Tên vào mạng
- Mật khẩu vào mạng

Dựa vào tên và mật khẩu đã cho, Windows ả T cung cấp cho người một số gọi là mã số của người sử dụng (user account). Mã số này được lưu dữ trong cơ sở dữ liệu là hệ thống

quản trị tài nguyên (SAM - Security Account Manager database). Hệ thống quản trị tài nguyên dùng để đảm bảo an toàn về tài nguyên trên mạng. ả gười vào mạng muốn truy nhập vào tài nguyên phải qua sự kiểm duyệt của hệ thống quản trị tài nguyên. Trong mô hình Workgroup mỗi máy trạm có một nguồn tài nguyên tương ứng với một hệ thống quản trị tài nguyên bảo vệ nó.

Chú ý: Mỗi người khai thác mạng phải nhớ nhiều mã số, vì ứng với mỗi máy trạm có một hệ thống quản trị tài nguyên riêng của nó.

2. Mô hình vùng (Domain)

Domain là một khái niệm rất cơ bản trong Windows ả T server, nó là hạt nhân để tổ chức các mạng có quy mô lớn.

Mỗi người tham gia trong Domain cần phải đăng ký thông tin sau:

- Tên Domain
- Tên người sử dụng
- Mật khẩu

Các thông tin này được lưu ở máy chủ dưới dạng một mã số, gọi là tài khoản người sử dụng (user account) và các mã số cũa người sử dụng trong một domain được tổ chức thành một cơ sở dữ liệu trên máy chủ. Khi người sử dụng muốn truy nhập vào một Domain người đó phải chọn tên Domain trong hộp thoại trên máy trạm. Máy trạm sẽ chuyển các thông tin về hệ thống quản trị tài nguyên (SAM - Security Account Manager database) của Domain để kiểm tra. Khi đó hệ thống quản trị tài nguyên trên máy chủ sẽ kiểm tra các thông tin này, nếu kết quả kiểm tra là đúng, người khai thác mới được quyền truy nhập vào tài nguyên của Domain.

Một máy Windows ả T mà không tham gia vào một Domain có nhược điểm sau:

- Máy trạm chỉ có thể cung cấp các mã số được tạo ra trên nó. ả ếu máy này bị hư hỏng thì những người khai thác mạng không thể truy nhập bằng mã số của họ. ả ếu máy này nằm trong một Domain nào đó thì các mã số này còn được lưu trong SAM của một Domain trên máy Máy chủ.
- Qua máy trạm không tham gia vào Domain, người khai thác mạng không thể truy nhập vào tài nguyên của Domain, mặc dù mã số của của người này có trong SAM của Domain

Trong một Domain thường có các loại máy thực hiện những công việc sau:

- Primary domain Controller (PDC), bao giờ cũng phải có để quản trị hệ thống các người sử dụng và các tài khoản trong Domain (hệ thống này gọi là cơ sở dữ liệu SAM - Security Account Manager của Domain). SAM trên máy chủ được thiết kế như hệ thống kiểm soát Domain. Trong một Domain chỉ có duy nhất một PDC.

- ả goài ra hệ thống còn có một hay nhiều máy làm Backup Domain Controller (BDC). Các BDC có thể dùng thay thế cho máy PDC trong trường hợp cần thiết, chẳng hạn máy PDC bị hư

ả gười quản trị Domain chỉ cần tạo tài khoản người sử dụng (user account) chỉ một lần trên máy Primary Domain Controller, thông tin được tự động copy đến các máy Backup Domain Controller.

3. Mô hình quan hệ giữa các Domain trong mạng Windows NT

Trong một mạng có thể có nhiều Domain nhưng một máy tính Windows ả T là thành viên của chỉ một domain tại mỗi thời điểm. Tuy nhiên, có một vài trường hợp đôi khi chúng ta cần truy cập tài nguyên trong những domain khác, để là được điều này hệ điều hành Windows ả T server cho phép giữa các Domain có thể tồn tại một quan hệ gọi là quan hệ tin cậy (trust relationship). Chúng ta có thể sử dụng quan hệ tin cậy giữa các Domain cho phép người dùng trên một Domain truy cập tài nguyên trong Domain khác.

Hai Domain A, B gọi là quan hệ tin cậy (trust relationship) mà trong đó Domain A tin cậy Domain B nếu giữa chúng có một mối liên kết sao cho người khai thác mạng của Domain B có thể truy nhập vào Domain A từ một máy trạm trong Domain B.

Từ góc độ của người quản trị mạng mục đích của việc thiết lập quan hệ tin cậy giữa các Domain là làm cho việc quản lý mạng trở lên đơn giản hơn bằng cách kết hợp các Domain vào một đơn vị quản lý. Trong quan hệ tin cậy các Domain được chia ra như sau:

- Domain được tin cậy (trusted domain)
- Domain tin cậy (trusting domain)

Một Domain là loại này hoặc loại kia *thông thường* phụ thuộc vào nó chứa mã số của người sử dụng (người sử dụng account) hay chỉ chứa tài nguyên (resource)

- Domain tin cậy (trusting domain) là Domain chứa tài nguyên.
- Domain được tin cậy (trusted domain) là Domain chứa mã số người sử dụng.

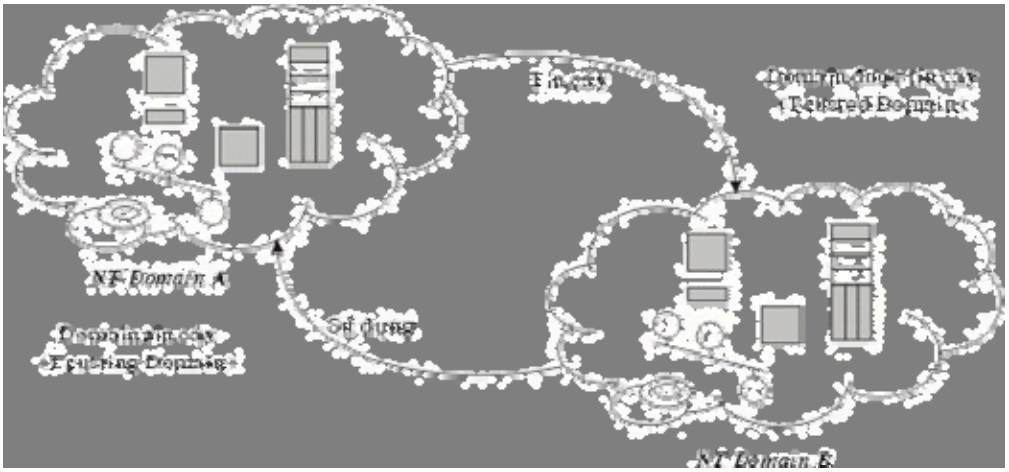
Khi người sử dụng truy nhập từ một máy trạm trong Domain tin cậy (trusting domain) vào Domain được tin cậy (trusted domain) thì quá trình kiểm soát diễn ra như sau:

- ả gười sử dụng phải cho mã số (mã số này ứng với tên, mật khẩu, tên domain cần truy nhập)
- Mã số được chuyển về máy chủ của Domain tin cậy.
- Máy chủ của Domain tin cậy chuyển mã số này sang Domain được tin cậy.
- Kết quả kiểm tra của máy chủ trong Domain được tin cậy diễn ra theo quá trình ngược lại.

Ở đây chúng ta chú ý:

- Việc liên kết giữa các Domain không có tính bắc cầu.

- Thông qua việc thiết lập mối quan hệ tin tưởng, chúng ta có thể sử dụng một tài khoản để truy xuất đến nhiều tài nguyên của nhiều Domain. Có thể quản trị nhiều Domain từ một vị trí tập trung.



Hình 11.1: Mô hình tin cậy của các Domain trong mạng Windows NT

4. Nhóm (group) trong Windows NT

Trong mạng Windows ả T khái niệm nhóm (group) là một trong những khái niệm quan trọng đối với công việc quản lý, điều hành mạng Windows ả T. ả hóm làm cho việc khai thác tài nguyên được dễ dàng thuận lợi và đơn giản hóa việc quản trị. Mỗi nhóm được đăng ký bởi một tài khoản (group account) và có các thành viên của nó. Các quyền đã được gán cho nhóm sẽ tự động gán cho các người sử dụng là thành viên của nhóm. Các tiện lợi của nhóm như sau:

- Quyền có thể được gán cho, hoặc hủy đi trên mọi thành viên của nhóm.

- Khi một người sử dụng bị loại ra khỏi nhóm, thì tự động bị mất các quyền đã được cấp khi còn trong nhóm.

Trong mạng Windows ả T người ta phân biệt phân biệt hai loại nhóm là nhóm toàn cục (global group) và nhóm cục bộ (local group).

5. Nhóm toàn cục (global group)

ả hóm toàn cục còn được gọi là nhóm vùng (domain group). Thành viên của nhóm là các người dùng cấp vùng (domain user). Họ ngược lại với người dùng cục bộ (local user) là người có phạm vi giới hạn trong máy tính mà họ được xác định. Thành viên của nhóm toàn cục được phép chuyển ra ngoài (export) một Domain khác. Phạm vi của nhóm toàn cục là toàn bộ vùng trên đó user dược xác định, và thấy được từ bất kỳ máy tính ả T nào trong vùng đó. Quyền có thể được gán cho nhóm toàn cục cho các tài nguyên trên một máy ả T Server hay ả T Workstation trong vùng.

Các tài khoản nhóm toàn cục được lưu ở PDC (Primary DomainController) của Domain, và được sao lưu đến các BDC (Backup Domain Controller) trong Domain đó.

ả hóm toàn cục có những đặc trưng sau:

- Thành viên của nhóm phải là các người sử dụng của domain (domain user account).
- ả hóm toàn cục có thể được gán quyền cho tài nguyên bất kỳ trong vùng mà chúng được xác định.
- ả hóm toàn cục có thể được gán quyền đến các tài nguyên trong vùng khác với vùng chúng được xác định khi quan hệ tin cậy (trust relationship) giữa các vùng có hiệu lực.
- Các thành viên của nhóm toàn cục có thể sử dụng nguồn tài nguyên trong vùng bất kỳ mà nhóm toàn cục có quyền.
- ả hóm toàn cục chỉ chứa mã số của người sử dụng trong Domain của nó. ả ó không thể chứa các nhóm cục bộ và nhóm toàn cục khác.

6. Nhóm cục bộ (local group)

ả hóm cục bộ, trái lại, được gán quyền cho nguồn tài nguyên trên máy ả T mà nó được xác định. ả ếu máy ả T là một phần của vùng, thì để tiện cho việc gán quyền, một nhóm cục bộ có thể chứa các tài khoản người dùng cấp vùng (domain user account) và các nhóm toàn cục trong Domain đó, nơi máy tính ả T là thành viên, hoặc những người dùng từ Domain được tin cậy. Các người dùng cấp vùng (domain user) có thể được gán quyền truy cập đến tài nguyên bất kỳ trong Domain đó.

ả ếu Windows ả T computer không nối với mạng thì các thành viên trong local group có thể được gán quyền để truy xuất đến tài nguyên trên máy tính mà trong đó các thành viên được tạo ra còn nếu Windows ả T computer nối vào mạng thì để tiện lợi cho việc phân quyền thì người quản trị mạng có thể đưa global group và domain user vào trong local group .

Có hai loại nhóm cục bộ: ***nhóm cục bộ trạm làm việc (workstation local group)*** và ***nhóm cục bộ vùng (domain local group).*** Một mạng làm việc theo cơ chế vùng bao gồm cả Windows ả T Server và Windows ả T Workstation việc hiểu rõ sự khác nhau giữa hai loại nhóm cục bộ là rất quan trọng.

a. Nhóm cục bộ trạm làm việc (Workstation local group):

ả hóm cục bộ trạm làm việc hiện diện trên Windows ả T Workstation trên đó chúng được tạo ra. Chúng được chứa trong dữ liệu SAM lưu trữ trên Windows ả T Workstation. Một người dùng cục bộ được tạo ra bằng công cụ *User Manager* của Windows ả T Workstation (khác với công cụ *User Manager for Domains* trên Windows ả T Server) có thể có quan hệ thành viên chỉ trong nhóm cục bộ của trạm làm việc đó. Một nhóm cục bộ trong một trạm làm việc chỉ có thể được dùng trên máy tính trên đó nhóm được tạo ra, và không thể làm việc rên bất kỳ máy Windows ả T nào khác.

ả hóm cục bộ trạm làm việc có thể chứa:

- Các tài khoản người dùng cục bộ từ trạm làm việc trên đó nó được xác định.
- Các tái khoản người dùng cấp vùng (domain user account) và các nhóm toàn cục từ vùng trong đó họ được xác định.
- Các tái khoản người dùng cấp vùng (domain user account) và các nhóm toàn cục từ các vùngï được ủy quyền.

b. Nhóm cục bộ vùng (Domain local group):

Nhóm cục bộ vùng hoạt động trên Windows ả T Server ở mức vùng, và được tạo ra bằng *User Manager for Domains* (trên Windows ả T Server). Các nhóm cục bộ vùng chỉ có thể hiện hữu trên máy Windows ả T Server tạo ra nó. Do đó, các nhóm cục bộ vùng có thể dùng để truy cập nguồn tài nguyên trên máy tính Windows ả T Server trong vùng đó, mà không dùng để truy cập nguồn tài nguyên trên máy tính Windows ả T Workstation trong vùng này. ả hóm cục bộ vùng không thể được gán quyền trên bộ điều khiển không có cấp vùng, thậm chí cả các máy chủ.

III. Các mô hình Domain trong mạng Windows NT

Windows ả T máy chủ cung cấp 4 kiểu tổ chức domain gọi tắt là các mô hình domain (domain models). Dưới đây là 4 mô hình tổ chức của nó:

- Mô hình domain đơn (single domain)
- Mô hình domain chính (master domain)
- Mô hình multiple master domain
- Mô hình complete truts

1. Mô hình Domain đơn (single domain)

Mô hình domain đơn là mô hình trong mạng chỉ có một domain. Mô hình này thích hợp cho mạng ít người khai thác, cần quản lý tập trung. Mô hình đơn nói chung tương tự như mô hình workgroup, trong mô hình này người sử dụng có thể xem xét, khai thác tài nguyên theo cả mô hình workgroup và mô hình domain.

Loại mô hình này không có các quan hệ ủy quyền vì chỉ có một domain duy nhất, domain này cũng chứa CSDL SAM cho toàn bộ mạng và việc quản trị mạng có thể thực hiện từ một vị trí trung tâm.

Các tài khoản người dùng trong vùng (domain user account) và tài khoản nhóm trong vùng (domain group acconunt) có thể được xây dựng và có các quyền truy cập tài nguyên được gán trên các nhóm và người dùng riêng rẽ và có một phạm vi bao gồm tất cả các máy vi tính trong vùng.

Trong mô hình Domain đơn vấn đề an toàn dữ liệu, quản lý hệ thống được xem xét một cách tốt hơn so với Workgroup.

2. Mô hình Domain chính (Master domain)

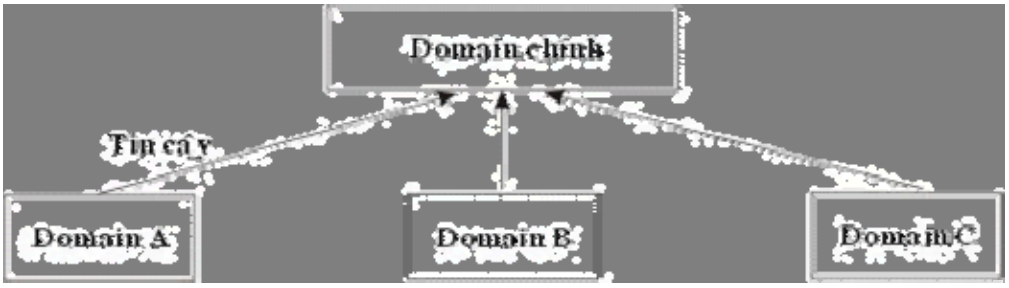
Mô hình Domain chính có thể được sử dụng cho các cơ quan khi họ muốn tổ chức mạng thành nhiều Domain tài nguyên (Resource domain) nhưng vẫn có những tiện lợi trong việc quản lý tập trung. Bằng cách phân chia tài nguyên mạng vào nhiều Domain, chúng ta sẽ tiện tổ chức và quản lý một lượng tài nguyên lớn. Một Domain chủ (master domain) được sử dụng để hổ trợ việc quản trị tập trung mà trong đó tất cả mã số của người sử dụng và mã số các nhóm toàn cục (global group) trên mạng được lưu giữ.

Đặc điểm của mô hình domain chính :

- Mô hình Master Domain là mô hình có nhiều Domain, trong đó có 1 Domain là Domain chính (premery domain). Mô hình này thích hợp cho mạng có số người dùng không quá lớn, nhưng cần phải phân chia thành các đơn vị nhỏ hơn nhưng việc quản lý được tiến hành tập trung.

- Trong mô hình này tất cả mã số của người khai thác mạng và mã số của các nhóm toàn cục (global group) đều chứa trên server trên Domain chính.

- Trong mô hình này tất cả các khác Domain đều tin cậy với Domain chính.



Hình 11.2: Mô hình Domain chính

Trong mô hình này mã số của người sử dụng quản lý tập trung và các nhóm toàn cục chỉ cần xác định một lần trong Domain chính. Tài nguyên được nhóm logic thành các đơn vị nhỏ hơn để có thể quản lý bởi từng Domain.

Mô hình Domain chính là mô hình quản lý tập trung vì vậy chiến lược phát triển mạng cần dựa vào các nhóm cục bộ và các nhóm toàn cục.

Mô hình này không những quản lý tập trung các mã số của người sử dụng mà còn cung cấp các dịch vụ như cài đặt phần mềm, sao chép backup cho tất cả các máy chủ trên mạng.

Tuy nhiên mô hình này có nhược điểm có thể gây ùn tắc nếu có quá nhiều nhóm và nhiều người dùng và các nhóm cục bộ cần phải xác định trong mỗi Domain mà chúng được sử dụng.

3. Mô hình nhiều Domain chính (muliple master domain)

Mô hình **nhiều Domain chính** (muliple master domain) có thể được sử dụng cho các tổ chức có nhiều khu vực và mỗi khu vực có nhiều bộ phận. Trong nhiều mạng kiểu như vậy, bộ phận điều hành riêng biệt cho mỗi khu vực muốn quản lý tập trung các tài nguyên mạng trong khu vực. Chúng ta xây dựng một Domain chủ (master domain) cho mỗi khu vực và chia các tài nguyên trong mỗi khu vực thành nhiều Domain tài nguyên (resoure domain) riêng biệt.

Trên mô hình này tồn tại các quan hệ sau:

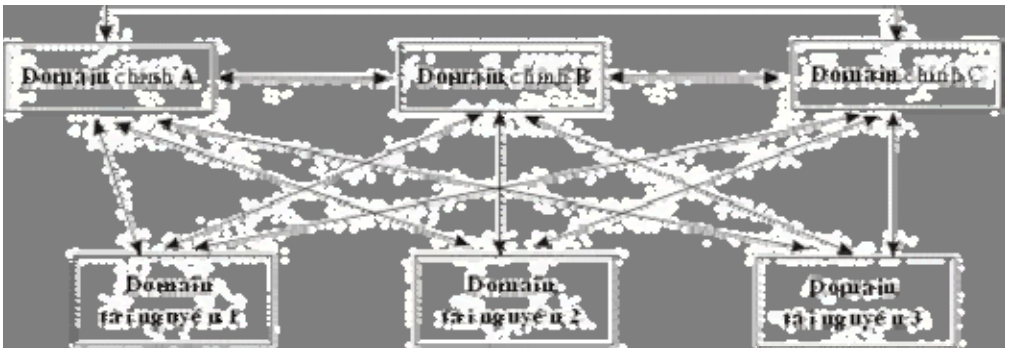
- Mỗi Domain chính quan hệ tin cậy hai chiều với các domain chính khác. Điều này cho phép mỗi Domain chính có thể quản lý các domain chính khác.

- Các Domain không phải là chính không có mã số của người sử dụng mà chỉ cung cấp tài nguyên trên mạng.

- Các Domain không phải là chính tin cậy đối với tất cả các Domain chính. ả hờ điều này mỗi mã số của người sử dụng sẽ được sử dụng trên tất cả các Domain chính và có được quyền truy nhập vào tài nguyên trong các tài nguyên trên các Domain khác của mạng.

Bằng cách phân chia tài nguyên mạng thành nhiều Domain, chúng ta có nhiều thuận lợi trong việc tổ chức quản lý một số lượng lớn các tài nguyên trong các đơn vị phù hợp.

Mô hình nhiều Domain chính có ưu điểm đối với mạng nhiều người dùng trong đó các tài nguyên được nhóm một cách logic theo công việc. Tuy nhiên các nhóm cục bộ và toàn cục phải xác định nhiều lần và mã số của người sử dụng phải chứa ở nhiều Domain chính .



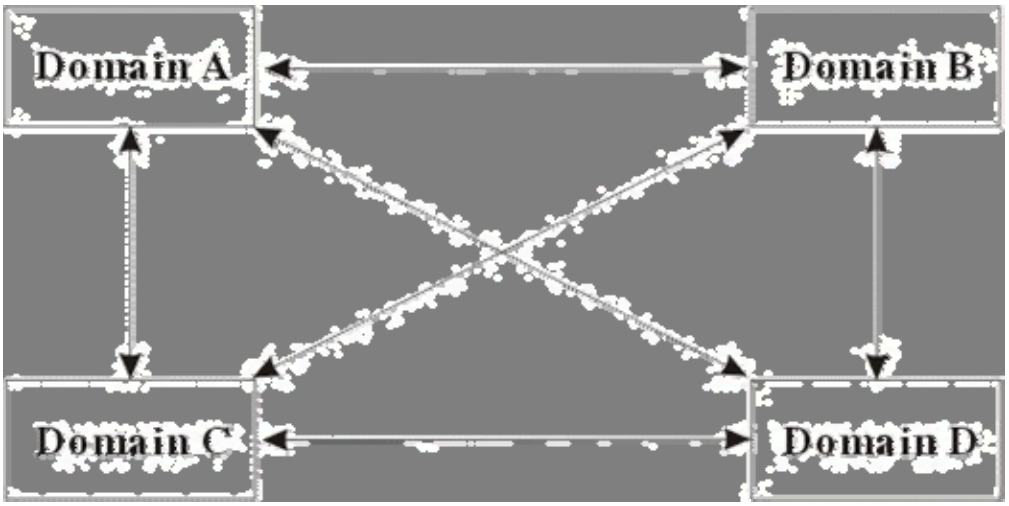
Hình 11.3: Mô hình nhiều Domain chính

4. Mô hình tin cậy hoàn toàn (complete trust)

Mô hình tin cậy hoàn toàn là mô hình mà trong đó mỗi Domain là quan hệ tin cậy 2 chiều với các Domain khác. Với mô hình này, người sử dụng có thể truy nhập vào bất kỳ Domain nào trên mạng từ một máy trạm nào đó.

Mô hình này có thể áp dụng với qui mô mạng tùy ý và phù hợp cho các cơ quan không có nhóm quản trị tập trung, nó cho phép không hạn chế số người khai thác mạng và số nhóm. Mỗi bộ phận trong đơn vị có thể kiểm soát được mã số của người sử dụng cũng như tài nguyên của bộ phận mình trong đó tài nguyên và mã số người sử dụng được nhóm thành một Domain.

www.viet-ebook.co.cc



Hình 11.4: Mô hình nhiều Mô hình tin cậy hoàn toàn

IV. Các mặt hạn chế của những mô hình Domain

Mô hình vùng có một số kẽ hở về cấu trúc. ả những hạn chế về domain được thảo luận ở đây nhằm mục đích giúp bạn thiết kế mạng chính xác và hoàn hảo.

- Domain ả T đơn điệu theo nghĩa là không có cách nào diễn tả quan hệ phân cấp hoặc nhóm tài nguyên trong một vùng đơn. ả gười dùng có thể sử dụng những quyền được ủy thác thể hiện các quan hệ giữa những vùng, nhưng đây là quan hệ sử dụng và không thích hợp cho việc tổ chức mạng dựa trên phạm vi địa lý, tài nguyên sở hữu, logic hoặc nền tảng sơ đồ tổ chức.

- Mô hình vùng Domain chính duy nhất theo Microsoft thích hợp cho các mạng ít hơn 40.000 người dùng và nhóm. Khi số người dùng và nhóm tăng lên, số quan hệ ủy quyền và chi phí quản lý quan hệ cũng tăng. ả ói cách khác chi phí quản lý mạng có thể tăng bất thình lình khi kích thước mạng tăng.

- ả gười dùng phải cẩn trọng về kẻ hở của quan hệ ủy quyền - đặc biệt quan hệ ủy quyền hai chiều. ả ếu không cẩn thận trong việc gán các quan hệ ủy quyền và không có kế hoạch đúng đắn, người sử dụng có thể kết thúc bằng một mô hình ủy quyền trọn vẹn, với tất cả những hạn chế của mô hình đi kèm.

- ả goài ra có một nguy cơ thực sự sẽ xảy ra là người cài đặt mạng có thể cài đặt một mạng hoạt động tốt trong thời gian ngắn còn khi mạng hoạt động dài hạn này sẽ ù nảy sinh vấn đề về mặt chính sách là ai ủy quyền cho ai.

www.viet-ebook.co.cc

Chương 12

Cài đặt, quản trị, sử dụng mạng Windows NT

I. Cài đặt hệ điều hành mạng Windows NT server

Trước khi cài đặt mạng Windows ả T thì cũng giống như cài các hệ điều hành khác chúng ta phải cắm card mạng vào máy, thiết lập mạng và đảm bảo nó được hoạt động tốt. Khi cài chúng ta có thể sử dụng phần mềm trên đĩa CD ROM (nếu máy của chúng ta là PC thì chúng ta sử dụng thư mục I386) hoặc chúng ta chép thư mục I386 lên đĩa cứng trước khi cài đặt. Để cài đặt Windows ả T ta và trong thư mục I386 và chạy lệnh "WIả ả T"

Chú ý trong trường hợp này chương trình sẽ yêu cầu chuẩn bị 3 đĩa mềm loại 1.44Mb để cài các chương trình khởi động cần thiết và trong quá trình cài đặt các đĩa mềm trên sẽ được sử dụng. ả ếu ta không muốn thì thực hiện lệnh "WIả ả T /B" và phải chỉ đường dẫn của chương trình nguồn như d:\I386.

Yêu cầu về phần cứng cho việc cài đặt windows NT

Thiết bị phần cứng	Yêu cầu
Processor	Intel 486, Pentium, Pentium Pro, những hệ thống chạy trên RISC (Ex: MIPS R4x00, DEC?s Alpha AXP). Windows ả T hỗ trợ lên đến 4 CPU ở Mode Symmetriccal Multi-Processing
Display device	VGA hay những thiêt bị có độ phân giải cao hơn
Hard disk	Tối thiểu phải có 110 MB Hard Disk còn trống trong suốt quá trình cài đặt
Floppy disk	31/2 inch hay 51/4 inch
CD-ROM	CD-ROM drive hay đĩa CD-ROM mà ta có thể truy xuất được thông qua đường mạng
ả etwork adapter	Một hay nhiều card mạng, card mạng không có cũng được nhưng chức năng mạng sẽ không có
Memory	ả T khuyến cáo ít nhất phải có 16 MB Ram cho cả hai hệ thống chạy trên Intel và RISC

Chương 13 :

Quản lý và khai thác File, thư mục trong mạng Windows NT

Trong số các tài nguyên của mạng chia sẻ cho người sử dụng thông tin lưu trữ trên đĩa cứng của các máy chủ là tài nguyên quan trọng nhất. Không phải ngẫu nhiên mà cái tên "File server" trở nên rất quen thuộc với những người dùng mạng giống như "ả etwork server". Tuy nhiên để làm sao có thể sử dụng, quản lý các tài nguyên đó một cách tốt nhất Windows ả T cung cấp cho chúng ta một cơ chế quản lý và phương thức khai thác. Thông thường chúng ta phải khai báo các tài nguyên trước khi chúng được người sử dụng khai thác. ả goài ra người sử dụng cũng được cung cấp quyền sử dụng một cách phù hợp.

I. Cơ chế an toàn của File và thư mục trong Windows NT

Quá trình truy cập tập tin (File hoặc thư mục) **trong Windows NT:** Khi một người sử dụng muốn truy cập một tập tin thì tất cả các thông tin về phương thức phục hồi giao dịch và phục hồi giao dịch khi bị lỗi sẽ được đăng ký bởi Log File Server. ả ếu giao dịch thành công, tập tin đó sẽ truy xuất được, ngược lại giao dịch sẽđược phục hồi. ả ếu có lỗi trong quá trình giao dịch, tiến trình giao dịch sẽ kết thúc.

Việc truy xuất tập tin (File hoặc thư mục) được quản lý thông qua các quyền truy cập (right), quyền đó sẽ quyết định ai có thể truy xuất và truy xuất đến tập tin đó với mức độ giới hạn nào. ả hững Quyền đó là Read, Execute, Delete, Write, Set Permission, Take Ownership.

Trong đó:

- **Read (R):** Được đọc dữ liệu, các thuộc tính, chủ quyền của tập tin.
- **Execute (X):** Được chạy tập tin.
- **Write (W):** Được phép ghi hay thay đổi thuộc tính.
- **Delete (D):** Được phép xóa tập tin.
- **Set Permission (P):** Được phép thay đổi quyền hạn của tập tin.
- **Take Ownership (O):** Được đặt quyền chủ sở hữu của tập tin.

Bảng tóm tắt các mức cho phép

Permission	R	X	W	D	P	O
ả o Access						
Read	X	X				

Change	X	X	X	X		
Full Control	X	X	X	X	X	X
Special Access	?	?	?	?	?	?

Để đảm bảo an toàn khi truy xuất đến tập tin (File và thư mục)ă, chúng ta có thể gán nhiều mức truy cập (permission) khác nhau đến các tập tin thông qua các quyền được gán trên tập tin. Có 5 mức truy cập được định nghĩa trước liên quan đến việc truy xuất tập tin (File và thư mục) là: ả o Access, Read, Change, FullControl, Special Access. Special Access được tạo bởi người quản trị cho bất cứ việc chọn đặt sự kết hợp của R, X, W, D, P, O. ả hững người có quyền hạn Full Control, P, O thì họ có quyền thay đổi việc gán các quyền hạn cho Special Access.

- Khi một người quản trị mạng định dạng một partition trong Windows ả T, hệ thống sẽ mặc định có cấp cho quyền Full Control tới partition đó cho nhóm Everyone. Điều này có nghĩa không hạn chế truy xuất của tất cả người dùng.

- Tùy thuộc trên yêu cầu bảo mật cho các tập người quản lý sẽ cân nhắc việc xóa bỏ nhóm Everyone trong danh sách các quyền hạn sau khi định dạng hay hạn chế nhóm Everyone với quyền Read. ả ếu sự hạn chế này là cần thiết, người quản trị nên cấp quyền hạn Full Control cho nhóm Administrators tới partition gốc.

Ở đây quyền truy cập được gán cho người sử dụng và nhóm người sử dụng do vậy quyền truy cập của một người sử dụng được tính bởi quyền hạn người đó và các nhóm mà người đó là thành viên. Khi người dùng đó truy xuất tài nguyên, các quyền hạn của người dùng được tính theo lối sau:

- ả hững quyền hạn của ngươi dùng và các nhóm trùng nhau.

- ả ếu một trong những quyền là ả o Access thì quyền hạn chung là ả o Access.

- ả ếu những quyền hạn đã yêu cầu được liệt kê không rõ ràng trong danh sách các quyền hạn, yêu cầu truy xuất này là không chấp nhận.

Một người sử dụng thuộc hai nhóm, nếu một nhóm quyền hạn của người dùng là ả o Access, nó luôn được liệt kê đầu tiên trong danh sách Access Control List.

Quyền sở hữu của các tập tin: ả gười tạo ra tập tin đó có thể cho các nhóm khác hay người dùng khác khả năng làm quyền sở hữu. Administrator luôn có khả năng làm quyền sở hữu của các tập tin.

ả ếu thành viên của nhóm Administrator có quyền sở hữu một tập tin thì nhóm những Aministrator trở thành chủ nhân. ả ếu người dùng không phải là thành viên của nhóm Administrator có quyền sở hữu thì chỉ người dùng đó là chủ nhân.

ả hững chủ nhân của tập tin có quyền điều khiển của tập tin đó và có thể luôn luôn thay đổi các quyền hạn. Trong File Manager, dưới Security Menu, sau khi xuất hiện hộp thoại

Owner, chúng ta lựa chọn tập tin, chủ nhân hiện thời và nhấn nút Take Ownership, cho phép lập quyền sở hữu nếu được cấp quyền đó.

Để có quyền sở hữu một tập tin chúng ta cần một trong những điều kiện sau:

- Có quyền Full Control.
- Có những quyền Special Access bao gồm Take Ownership.
- Là thành viên của nhóm Administrator.

II. Các thuộc tính của File và thư mục

- **Archive:** Thuộc tính này được gán bởi hệ điều hành chỉ định rằng một File đã được sửa đổi từ khi nó được Backup. Các phần mềm Backup thường xóa thuộc tính lưu trữ đó. Thuộc tính lưu trữ này có thể chỉ định các File đã được thay đổi khi thực thi việc Backup.
- **Compress:** Chỉ định rằng các File hay các thư mục đã được nén hay nên được nén. Thông số này chỉ được sử dụng trên các partition loại ả TFS.
- **Hidden:** Các File và các thư mục có thuộc tính này thường không xuất hiện trong các danh sách thư mục.
- **Read Only:** Các File và các thư mục có thuộc tính này sẽ không thể bị xóa hay sửa đổi.
- **System:** Các File thường được cho thuộc tính này bởi hệ điều hành hay bởi chương trình OS setup. Thuộc tính này ít khi được sửa đổi bởi người quản trị mạng hay bởi các User.
- ả goài ra các File hệ thống và các thư mục còn có cả hai thuộc tính chỉ đọc và ẩn.

Lưu ý: Việc gán thuộc tính nén cho các File hay thư mục mà ta muốn Windows ả T nén sẽ xảy ra trong chế độ ngầm (background). Việc nén này làm giảm vùng không gian đĩa mà File chiếm chỗ. Có một vài thao tác chịu việc xử lý chậm vì các File nén phải được giải nén trước khi sử dụng. Tuy nhiên việc nén File thường xảy ra thường xuyên như là các File dữ liệu quá lớn mà có nhiều người dùng chia sẻ.

III. Chia sẻ Thư mục trên mạng

Không có một người sử dụng nào có thể truy xuất các File hay thư mục trên mạng bằng cách đăng nhập vào mạng khi không có một thư mục nào được chia se.

Việc chia sẻ này sẽ làm việc với bảng FAT và ả TFS file system. Để nâng cao khả năng an toàn cho việc chia sẻ, chúng ta cần phải gán các mức truy cập cho File và Thư mục.

Khi chúng ta chia sẻ một thư mục, thì chúng ta sẽ chia sẻ tất cả các File và các Thư mục con. ả ếu cần thiết phải hạn chế việc truy xuất tới một phần của cây thư mục, chúng ta phải

sử dụng việc cấp các quyền cho một user hay một nhóm đối với các Thư mục và các File đó.

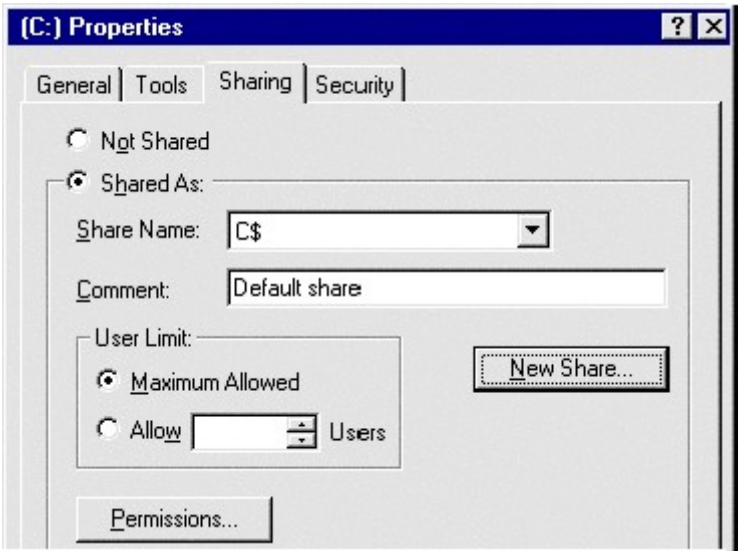
Để chia sẻ một Thư mục, ta phải Login như một thành viên của nhóm quản trị mạng hay nhóm điều hành server.

Tất cả các thủ tục chia sẻ thư mục được thực thi trong Windows ả T Explorer.

Để chia sẻ một thư mục ta phải thực hiện các bước sau:

- **Right-click** lên Thư mục đó trong Windows ả T Explorer. Hiện ra menu

- Click **Properties** trong Menu. Hiện ra hộp đối thoại sau:



- Chọn **Sharing tab** hiện ra hộp đối thoại sau:

- Chọn **Shared As** để kích hoạt việc chia sẻ.

- Đưa một tên cần chia sẻ vào hộp **Share name**. Mặc nhiên tên Thư mục được chọn sẽ hiện ra. Đưa dòng ghi chú liên quan đến việc chia sẻ thư mục đó vào hộp **Comment**

- Thiết lập giới hạn số lượng các user bằng cách gỏ một con số vào hộp **Allow**

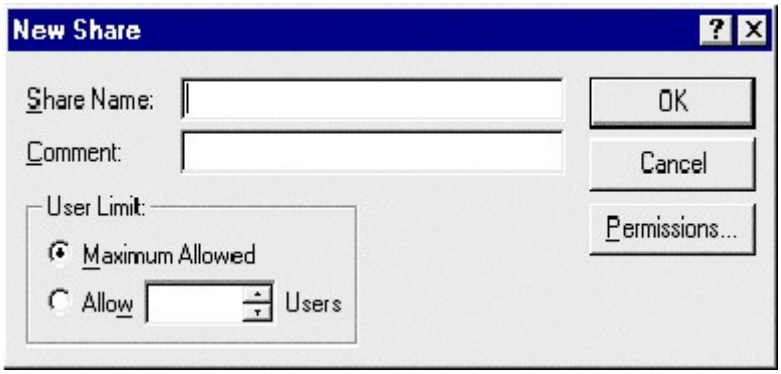
- ả ếu muốn hạn chế việc truy xuất thì click **Permissions button.**

- **Click OK.**

Sau khi một thư mục được chia sẻ Icon cho thư mục đó có 1 bàn tay chỉ định rằng thư mục đó đã được chia sẻ.

ả ếu chúng ta muốn thêm một chia sẻ mới với cùng một thư mục đã được chia sẻ (có thể với hai chia sẻ có hai quyền truy cập khác nhau), ta thực hiện các bước sau:

- **Right-click** vào thư mục đã được chia sẻ trong Windows ả T Explorer.
- Click **Properties** trong **Menu** rút gọn, hiện ra hộp đối thoại **Properties**
- Click **Sharing tab.**
- Click button **New Share** để tạo một sự chia sẻ mới, hiện ra hộp đối thoại sau

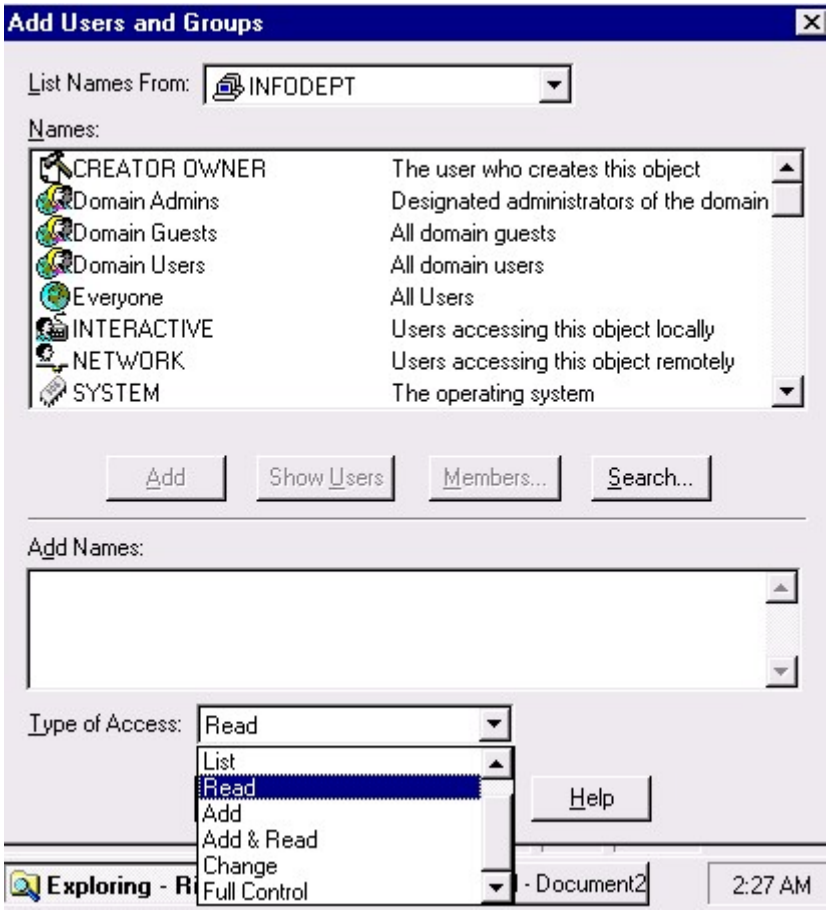


- Mỗi lần tạo một sự chia sẻ chúng ta phải đưa một tên mới cũng như những lời chú thích việc chia sẻ đó sẽ cho ai sử dụng.

IV. Thiết lập quyền truy cập cho một người sử dụng hay một nhóm

Để thiết lập các quyền truy cập đối với một thư mục đã được chia sẻ cho một người sử dụng hay một nhóm ta thực hiện:

- **Right-click** lên thư mục đó trong Windows ả T Explorer.
- Click **Properties** trong menu rút gọn**.**
- Chọn **Sharing** tab để hiện các tính chất của thư mục đó
- Click button **Permissions** trong **sharing tab** . Hiện ra Cửa sổ ***The Access Through Share Permissions.***
- Chọn button **Add**, hiện ra cửa sổ ***Add User and group.***



- Chọn một tên trong hộp ***Names*** và click button **Add.** Kết quả là tên đó được đưa vào hộp ***Add Names.***

- Chọn quyền truy xuất trong hộp ***Type of Access*** cho các tên đã chọn .

- Click button **OK.**

Khi chúng ta tạo một sự chia sẻ mới, quyền truy cập mặc nhiên cho nhóm **Everyone** là đầy đủ (**Full Control**). Giả sử rằng chúng ta sẽ gán giá trị mặc nhiên này cho quyền truy cập của thư mục và File. Khi cần thiết sẽ hạn chế việc truy xuất tới thư mục đó.

Ở đây có một vài chú ý:

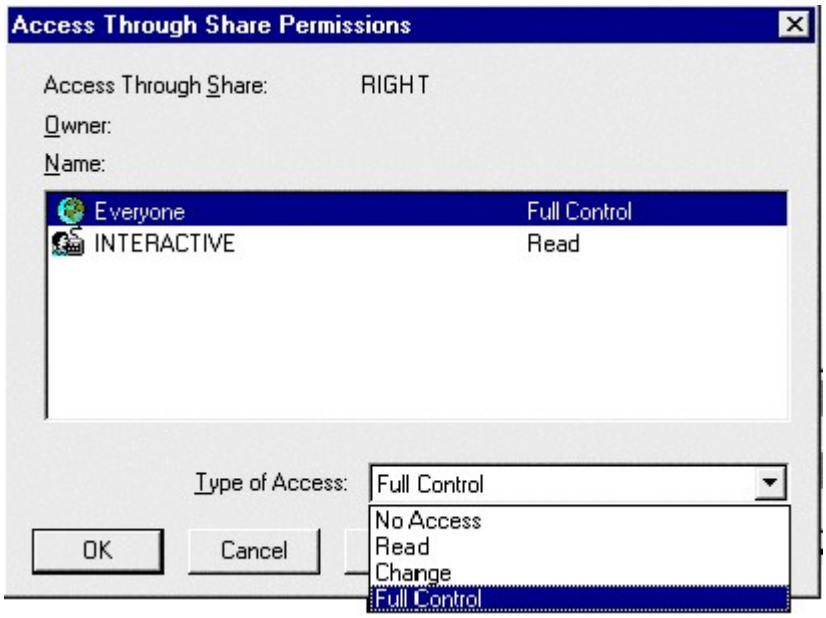
- Các người sử dụng thường chỉ cóù quyền đọc trong các thư mục chứa các chương trình ứng dụng vì họ không cần phải sửa đổi các File.

- Trong một vài trường hợp, các chương trình ứng dụng đòi hỏi các user chia sẻ một thư mục cho các File tạm thời. ấ ếu thư mục đó nằm trong cùng thư mục chứa trình ứng dụng, chúng ta có thể cho phép user tạo hay xóa các File trong thư mục đó bằng việc gán quyền **Change**.

- Thông thường các người sử dụng cần quyền **Change** trong bất kỳ thư mục nào chứa các Files dữ liệu và chỉ trong các thư mục cá nhân của ho là có đầy đủ các quyền truy cậpï.

Để sửa đổi các quyền truy cập đối với một thư mục đã được chia sẻ ta thực hiện:

- **Right-click** lên thư mục được chia sẻ trong Windows ả T Explorer.
- Click **Properties**
- Click **Sharing** tab.
- Click button **Permissions** hiện ra cửa sổ ***Access Through Share Permissions*** sau:



- Chọn 1 tên trong hộp ***Name***
- Chọn một quyền khác trong hộp ***Type of Access*** mà ta muốn gán.
- Click **OK**.

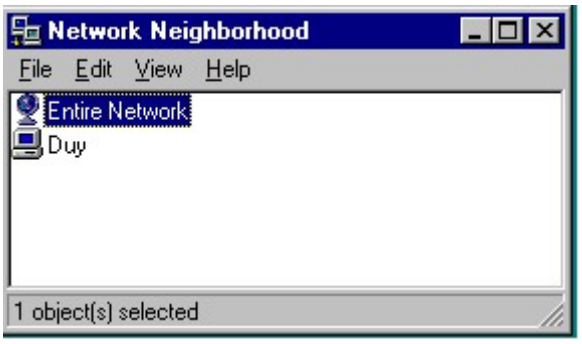
Thông qua việc chia sẻ một thư mục cho một user hay một nhóm cũng góp phần vào việc bảo đảm an toàn cho một thư mục không cho user khác hay nhóm khác truy xuất thư mục đó.

V. Sử dụng các thư mục mạng

Muốn sử dụng các thư mục mạng thì trước hết thư mục đó được cho phép chia sẻ, chúng ta phải liên kết thư mục mạng đó với tên một chữ cái tương ứng như một tên đĩa mạng (E,F ,G ,H I,...). Sau khi thư mục được chia sẻ đã kết nối với ký tự ổ đĩa mạng người dùng có thể truy cập thư mục được chia sẻ, các thư mục và file con của nó như là nó đang ở trên máy tính của mình .

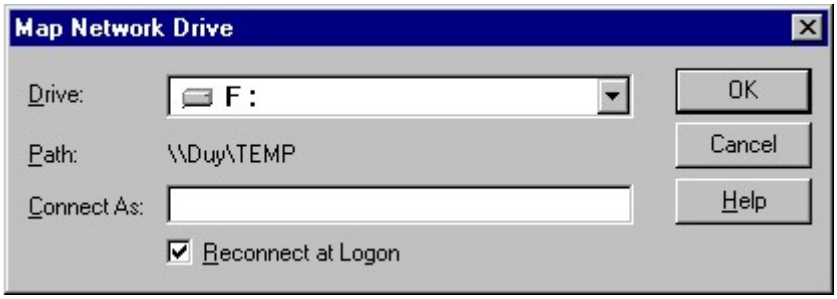
Có thể dùng Network Neighborhood để thực hiện công việc trên như sau :

- Click đúp trên **Network Neighborhood** để mở trình duyệt mạng.



- Duyệt qua **Network Neighborhood** để tìm nơi muốn liên kết.

- Click phải vào thư mục đã được chia sẻ mà chúng ta muốn truy cập và chọn **Map Network Drive** trong thực đơn **Options** ta thấy hộp **Map Network Drive** hiện ra



- Trong trường **Drive** của hộp thoại **Map Network Drive**, chọn ổ đĩa mạng chúng muốn liên kết với thư mục chia sẻ.

- ẩ ếu thấy cần, chọn Path và gõ vào tên theo tổng quát Uẩ C (Universal ẩ aming Convention - xem cấu trúc ở phần dưới) để sửa lại đường dẫn tới tài nguyên được chia sẻ. (Việc này chỉ thực hiện khi sử dụng ẩ etwork ẩ eighborhood.)

- ẩ ếu chúng ta không được quyền để truy cập vào tài nguyên chia sẻ trên nhưng trong cương vị người dùng khác thì chúng ta được quyền truy cập, trong trường hợp đó hãy gõ tên người dùng đó vào trường **Connect As**.

- Kích hoạt hộp kiểm tra **Reconnect at Logon** nếu muốn liên kết lâu dài, đó là loại kết nối được phục hồi mỗi lần chú ta đăng nhập vào mạng.

- Chọn **OK** để lưu các thông tin trên.

ẩ goài ra ta có thể dùng lệnh **NET USE** để thực hiện các công việc trên.

Lệnh ẩ ET USE dùng Universal ẩ aming Convention (Uẩ C) để truy cập các tài nguyên dùng chung. Tên Uẩ C bắt đầu bằng một dấu phân cách đặt biệt \\, dấu này chỉ sự bắt đầu của tên Uẩ C (tên Uẩ C có dạng "\\computer_name\share_name[\sub_directory]". ẩ ET USE được dùng đểû truy cập một nguồn tài nguyên dùng chung. Lệnh ẩ ET USE dùng bộ hướng dẫn mạng (ẩ etwork Redirector) trên máy tính ẩ T để thiết lập sự nối kết dùng nguồn tài nguyên chung.

Chúng ta có thể xem ai dùng các file dùng chung khi ta đang xem trạng thái của một file dùng chung, File Manager sẽ cung cấp cho ta các thông tin bằng dùng chọn Properties trong thực đơn File

Đề mục	Nội dung
Total Opens	Tổng số các user đang làm việc với file đó
Total Locks	Tổng số các khóa trên file
Open By	Tên của người dùng đã mở file
For	Loại truy xuất mà người dùng đã mở file
Locks	Một số khóa mà người dùng đặt trên file
File ID	Con số nhận diện của file

Khi chúng ta dùng Windows Explorer để xem các tài nguyên chúng ta có thì các ổ đĩa mạng xuất hiện và cho chúng ta khai thác.

www.viet-ebook.co.cc

Chương 14 :

Sử dụng máy in trong mạng Windows NT

Hiện nay máy in trên mạng cũng là một tài nguyên việc chia sẻ của mạng cho người sử dụng. Tuy các máy in đang ngày càng rẻ đi nhưng với nhu cầu về chất lượng đang ngày một cao thì việc chia sẻ các máy in đắt tiền trên mạng vẫn đang cần thiết. Windows ả T là một hệ điều hành mạng mà bất kỳ máy tính Windows ả T nào cũng có thể cung cấp các dịch vụ in ấn cho người sử dụng trong mạng.

Khi chia sẻ một máy in trên mạng (cho nhiều người có thể cùng sử dụng) chúng ta cần phải giải quyết những vấn đề sau :

- Máy in không làm được 2 việc một lúc, nếu phải nhận cùng một lúc thì sẽ có va chạm, do vậy mạng phải có cơ chế sắp xếp công việc sao cho máy in có thể thực hiện một cách lần lượt các công việc in.

- Các công việc in được thực hiện bởi những người sử dụng khác nhau có thể cần những mức độ ưu tiên khác nhau và hệ thống quan lý in cần có khả năng thực hiện điều này.

I. Cơ chế in trong mạng Windows NT

Thông thường máy in mạng được quản lý thông qua một máy chủ mà trên đó thực hiện nhiệm vụ quản lý các công việc in, máy chủ đó thường được gọi là máy chủ in (Print server) và chạy chương trình quản lý in. Windows ả T cho phép cài đặt máy in tại bất cứ đâu trên mạng, mỗi một máy có cài đặt Windows ả T đều có thể thực hiện nhiệm vụ máy chủ in. ả ó có thể quản lý máy in gắn trực tiếp vào nó hay một máy in gắn vào máy khác trên mạng.

Để giải quyết những vấn đề đặt ra với công việc in trên mạng Windows ả T sử dụng kỹ thuật gọi là Spooling mà chủ yếu như sau:

- Khi người sử dụng quyết định thực hiện một công việc in thì công việc in đó không trực tiếp gửi ra máy in mà nó được đặt trong một file tại máy chủ in. Ở đây việc thực hiện giống như hàng đợi rạp hát, nó là một vùng lưu trữ các công việc in và có nhiệm vụ ngăn chặn xung đột khi các user chi xuất đồng thời ra máy in.

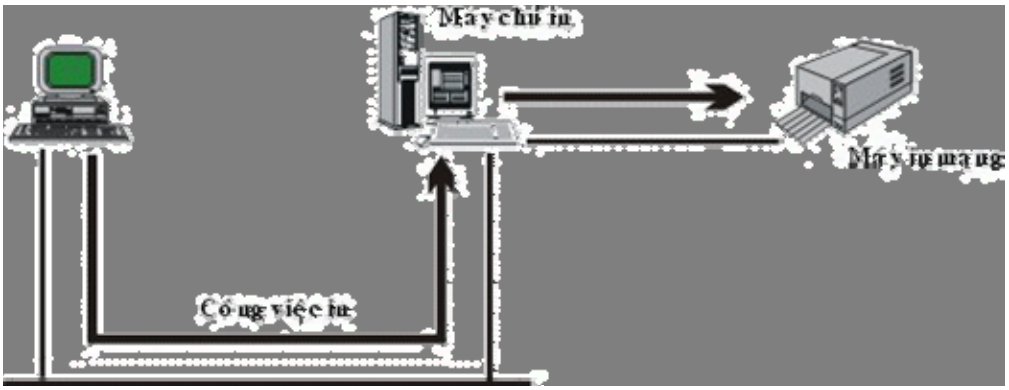
- Máy chủ in duy trì các hàng đợi để cất giữ các công việc in và đưa chúng tới máy in ngay khi có thể. Trong khi đó người sử dụng có thể làm tiếp công việc ngay khi công việc in được cất vào hàng đợi.

- Khi máy in rảnh máy chủ in sẽ chuyển lần lượt các công việc in đang đứng đợi trong hàng tới máy in. Tại đây máy chủ in phải có một khả năng lưu trữ dữ liệu lớn để có thể lưu trữ nhiều công việc in một lúc và cần phải có khả năng đáp ứng những yêu cầu đa dạng của các công việc in.

Để giải quyết vấn đề nẩy sinh với máy in trong mạng Windows ả T tiến hành phân biệt giữa máy in vật lý gọi là Printing device và một thực thể logic của máy in gọi là logic printer. Máy in logic được sử dụng để kiểm soát các tác vụ sau đây :

- Công việc in được gởi đi đâu.
- Công việc in ấn gởi đi khi nào.
- Thứ tự ưu tiên của các tác vụ in.

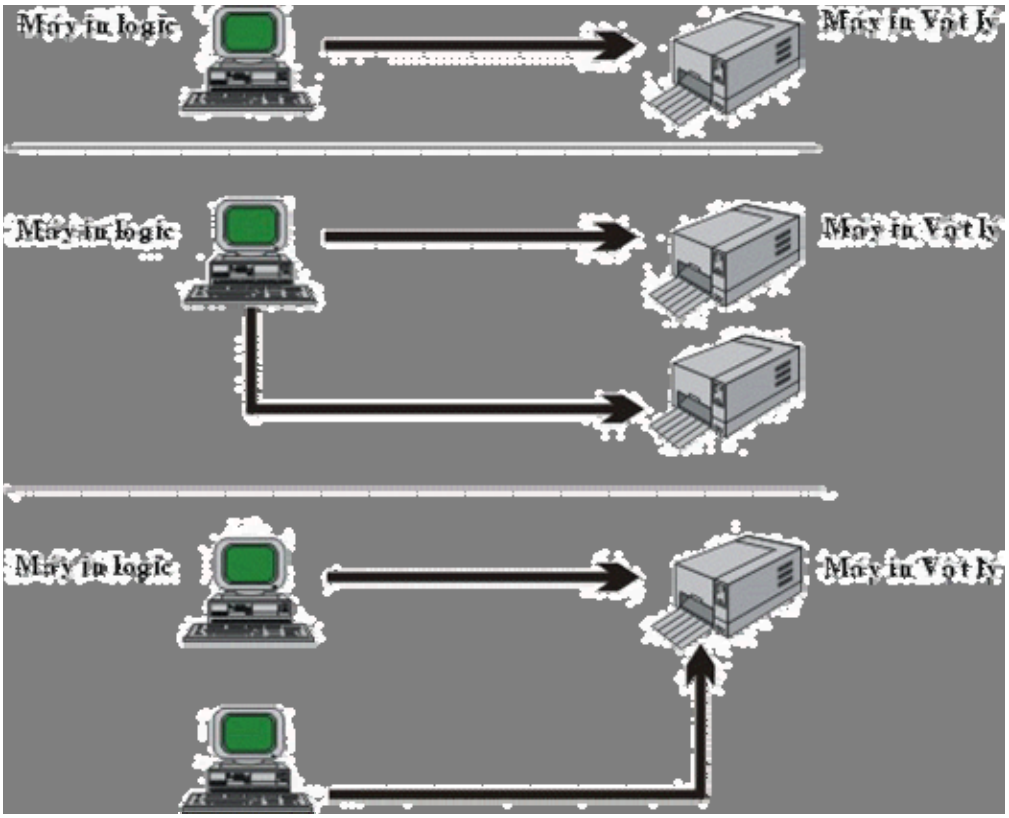
ả gười sử dụng in ra spool thông qua việc in ra máy in logic, họ sử dụng máy in logic như là máy in đang được gắn là máy của họ nhưng thực sự các dữ liệu được in ra máy in logic được chuyển cho mạng và qua đó đến máy chủ in trước khi được đưa ra máy in mạng.



Hình 14.1: Máy chủ in và spool

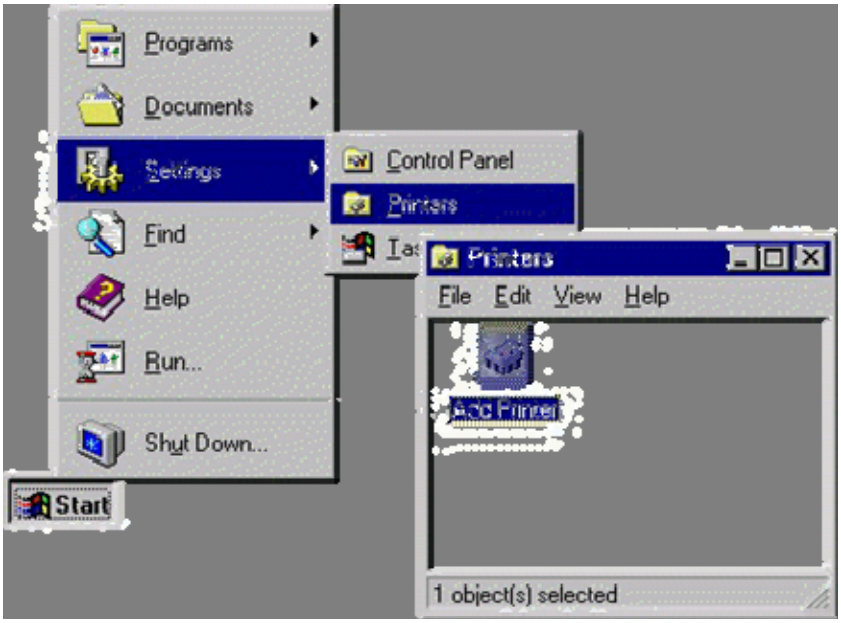
Máy chủ in sẽ liên kết các máy in logic với máy in vật lý, nó phải đảm bảo các công việc in phải được đưa đúng đến máy in vật lý. Tại đây có 3 trường hợp có thể đối với mối quan hệ giữa máy in logic và máy in vật lý

- Một máy in logic liên kết với một máy in vật lý.
- ả hiều máy in logic liên kết với một máy in vật lý.
- Một máy in logic liên kết với nhiều máy in vật lý.

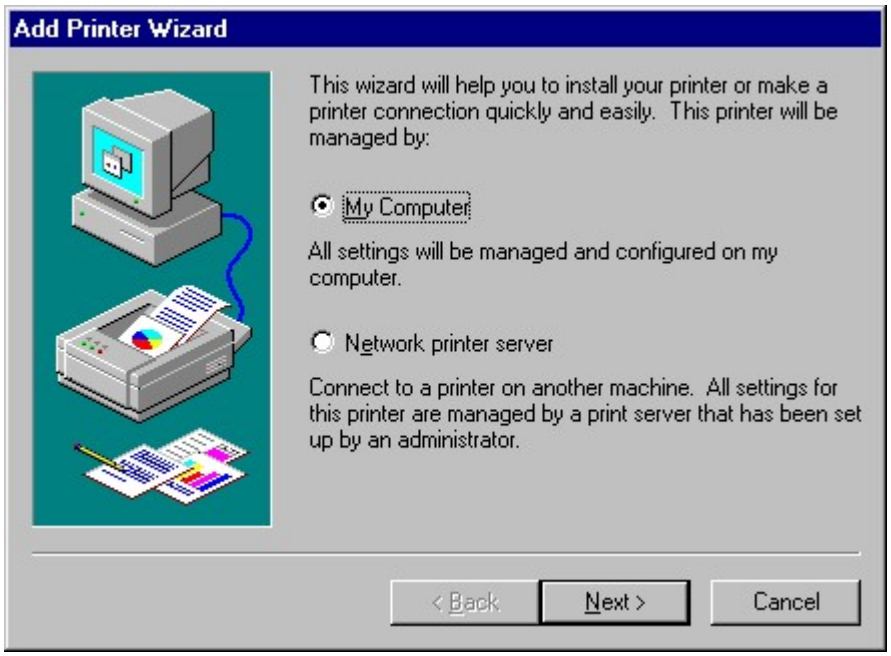


Hình 14.2: Liên kết giữa máy in Logic và máy in vật lý

ả ếu Server chưa cài đặt máy in logic, ta phải cài đặt máy in logic tương ứng với một máy in thực tế cho Server. Vào menu **Start**, chọn **Settings**, chọn **Printers**, chọn **Add Printer** như:



Hộp sau đó hộp hội thoại **Add printer** winzar hiện ra

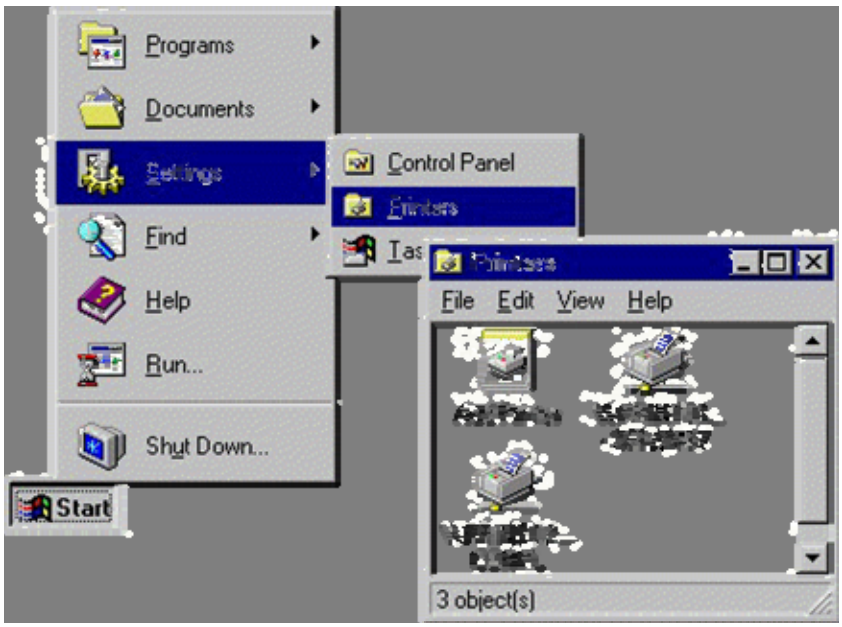


- Chọn My Computer nếu máy in của chúng ta không có card mạng và được nối trực tiếp vào Server.

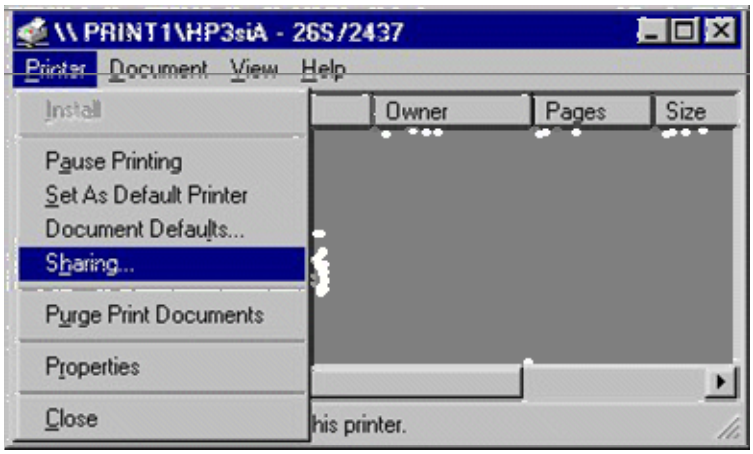
- Chọn ả etwork printer server nếu máy in của chúng ta nối trực tiếp vào mạng.

- Chọn ả ext, chọn cổng nối với máy in (thường là LPT1). Chọn tên hãng sản xuất và loại máy in ta đang dùng, chọn ả ext, ta phải trả lời thêm vài câu hỏi phụ như ta có muốn in trang test không? Có muốn đặt máy in này là ngầm định không?

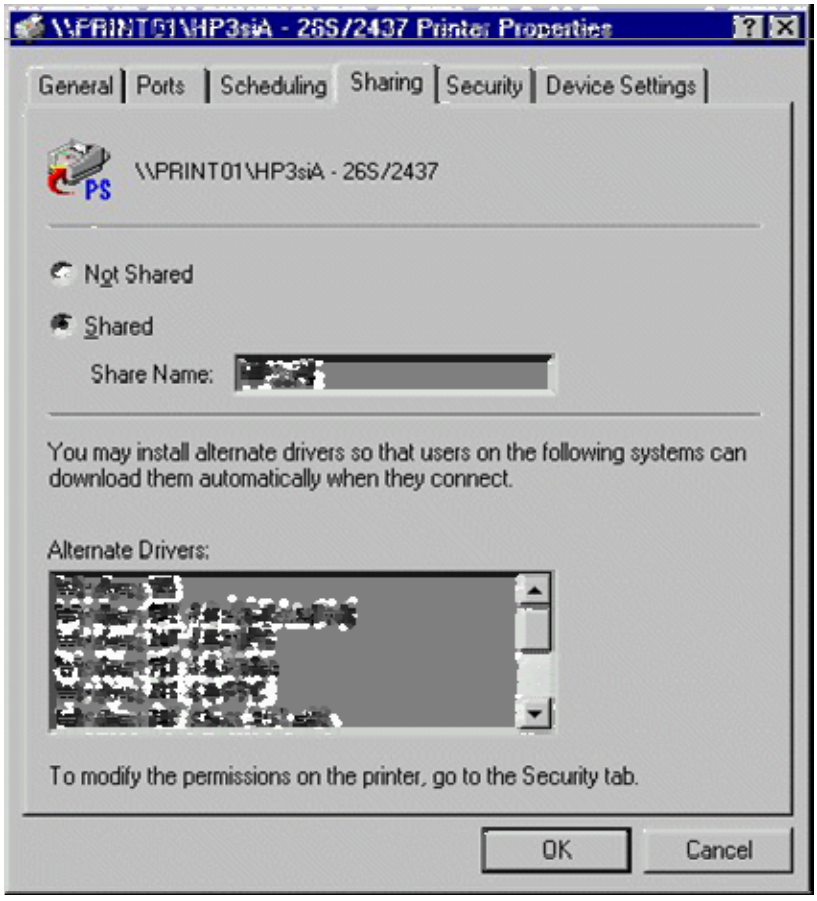
Sau khi cài đặt, chúng ta sẽ thấy xuất hiện thêm biểu tượng máy in mà vừa được cài đặt trong khung máy in. Chúng ta phải cho phép dùng chung máy in nàybằng cách lựa chọn máy in đó Trong khung Printers



- Ta nhắp chuột phải vào tên máy in đó, chọn **Sharing** như hình sau:



Khung **Printer properties** hiện ra cho chúng ta nhập các thông số như: tên máy in logic (Share namem), các tính chất khác như về an toàn. mà chúng ta muốn khi phục vụ mạng.



- Cuối cùng chọn **OK**, lúc này, ta sẽ thấy ở dưới biểu tượng máy in có bàn tay đỡ chứng tỏ máy in này đã được phép dùng chung. ấ ếu trên Server cài đặt nhiều loại máy in với nhiều chế độ khác nhau, ta có thể chọn máy in ngầm định bằng cách đánh dấu vào mục **Set As Default**.

- Để máy trạm có thể in được qua Server, nếu chưa cài đặt chúng ta phải cài máy in như sau: nhắp đúp vào tên Server có nối với máy in, khung **Shared Printers** sẽ hiện ra danh sách các máy in đã cài trên Server, chúng ta chọn tên máy in cần nối rồi bấm **OK**.

Quay trở lại khung màn hình **Print Manager** chúng ta nhìn thấy thông báo máy in này đã được phép sử dụng. Thoát ra khỏi **Print Manager** và chúng ta có thể in qua máy in mạng trên bất cứ một phần mềm nào trên Windows như Winword, Excel, v.v...

Bất kỳ máy tính Windows ả T có thể được cấu hình như là một **print server**. Tuy nhiên chỉ có những người là thành viên của những nhóm sau đây mới có quyền tạo ra các máy in:

- Administrator (ả T Worstation and Server).
- Server Operator (ả T Server).
- Print Operator (ả T Server).
- Power Users (ả T Worstation).

II. Bảo mật của máy in

Windows NT có các mức độ bảo mật trong in ấn như sau:

- **Quyền sở hửu máy in (Ownership) :** người sử dụng tạo ra một máy in chính là người chủ sở hửu máy in đó và có toàn quyền trên tất cả các thuộc tính của máy in logic. ả gười chủ sở hửu máy in có thể gán quyền cho những người dùng khác quản lý tài liệu hay toàn quyền điều khiển việc in ấn. Một người sử dụng có toàn quyền thì họ toàn quyền sở hửu máy in logic đó.

- **Quản lý thuộc tính máy in (Permissions):** quyền quản lý máy in bao gồm 4 quyền sau:

  - **No access:** không được phép truy cập.
  - **Print:** in
  - **Manage document:** quản lý văn bản, có khả năng thực hiện các thao tác: Điều khiển khởi đặt tài liệu, ả gừng, phục hồi, khởi động lại,và xóa các tài liệu.
  - **Full control:** toàn quyền điều khiển, thực hiện các quyền quản lý tài liệu và các quyền sau đây:
    - Thay đổi trật tự in ấn tài liệu.
    - ả gừng, tổng hợp lại,che dấu các máy in logic.
    - Thay đổi thuộc tính của máy in logic.
    - Hủy các máy in logic.
    - Thay đổi quyền của máy in logic

Có thể xem tài liệu ở máy in logic và quản lý chúng theo nhiều cách. ả gười sử dụng luôn quản lý được tất cả các tài liệu mà họ tạo ra. Để quản lý được các tài liệu của các người sử dụng khác, phải là người chủ sở hửu của máy in logic hay là thành viên của các nhóm:

- Administrator.
- Server Operator
- Print operator.

Bất kỳ một máy in nào cũng có thể làm việc trong môi trường mạng nhưng điều quan trọng là xem xét **chu kỳ làm việc (duty cycle)** của máy in. ả ghĩa là phải xem xét số lượng trang in tối đa mà máy in có thể in ra trong một khoảng thời gian nhất định.

Các máy in được thiết kế cho mạng thường có **chu kỳ làm việc (duty cycle)** cao. Các máy in có thể gắn vào bất cứ nơi đâu trên mạng. Công việc in không phù thuộc vào các thiết bị phần cứng hay các thiết bị kết nối mà do được quản lý bởi một **print server** và dữ liệu được chuyển vận trên mạng.

Chương 15 :

Các dịch vụ mạng của Windows NT Server

Cũng như các hệ điều hành khác Windows ả T cũng có những ưu, khuyết điểm của nó, tuy nhiên Windows ả T hiện nay chinh phục được nhiều người dùng với những ưu điểm không thể chối cãi. Là hệ điều hành mạng cho phép tổ chức quản lý một cách chủ động theo nhiều mô hình khác nhau: peer-to-peer, clien/server. ả ó thích hợp với tất cả các kiến trúc mạng hiện nay như: hình sao (start), đường thẳng (bus), vòng (ring) và phức hợp. ả ó có một số đặc tính ưu việt bảo đảm thực hiện cùng lúc nhiều chương trình mà không bị lỗi. Bản thân Windows ả T đáp ứng được hầu hết các giao thức phổ biến nhất trên mạng và cũng hỗ trợ được rất nhiều những dịch vụ truyền thông trên mạng. ả ó vừa đáp ứng được cho mạng cục bộ (LAả) và cho cả mạng diện rộng (WAả).

Windows ả T cho phép dùng giao thức Windows ả T TCP/IP, vốn là một giao thức được sử dụng rất phổ biến trên hầu hết các mạng diện rộng và trên Internet. Giao thức TCP/IP dùng tốt cho nhiều dịch vụ mạng trên môi trường Windows ả T.

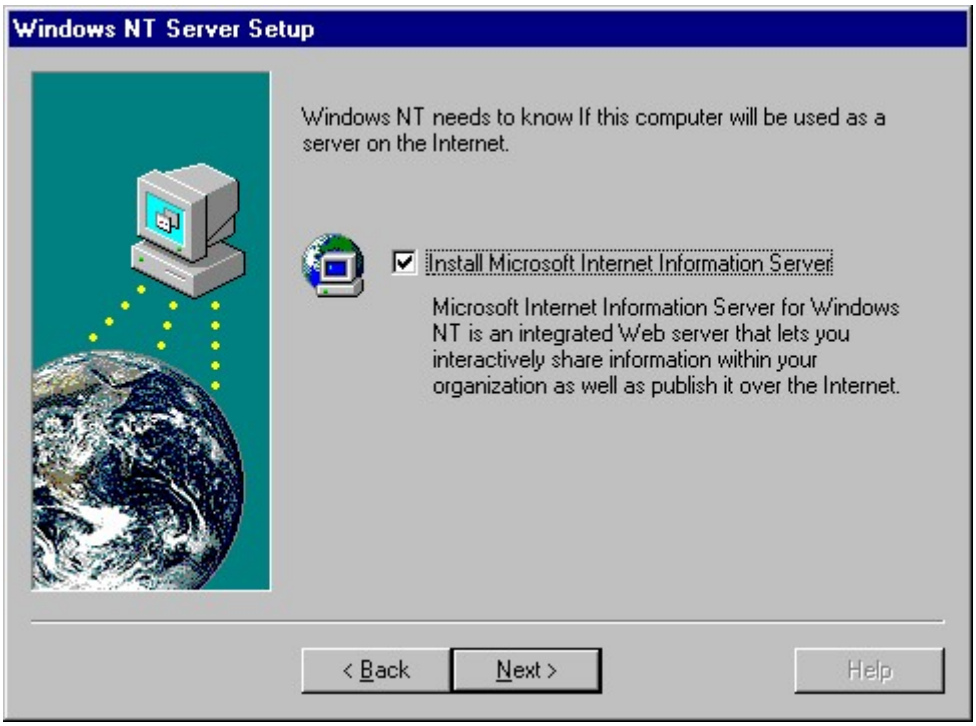
I. Internet Information Server (IIS)

Internet Information Server là một ứng dụng chạy trên Windows ả T, tích hợp chặt với Windows ả T, khi cài đặt IIS, IIS có đưa thêm vào tiện ích màn hình kiểm soát (Performance monitor) một số mục như thống kê số lượng truy cập, số trang truy cập. Việc kiểm tra người dùng truy cập cũng dựa trên cơ chế quản lý người sử dụng của Windows ả T. Sau khi cài đặt IIS, trong thư mục InetSrv sẽ có các thư mục gốc tương ứng cho từng dịch vụ chọn cài đặt.

IIS bao gồm 3 dịch vụ: World Wide Web (WWW), chuyển file (FTP - File Transfer Protocol) và Gopher. Cả 3 dịch vụ này đều sử dụng kết nối theo giao thức TCP/IP.

1. Cài đặt dịch vụ Internet Information Server

Khi cài đặt hệ điều hành Windows ả T đến phần mạng Windows ả T sẽ hỏi chúng ta xem có cài đặt dịch vụ Internet Information Server hay không với hộp hội thoại



Hình 15.1: Màn hình cài đặt của IIS

Để thực hiện việc cài đặt chúng ta Click vào phím ả ext và Hệ thống sẽ bắt đầu cài đặt các dịch vụ Internet Information Server.

2. Các dịch vụ trong IIS

a. WWW (World Wide Web) :

Là một trong những dịch vụ chính trên Internet cho phép người sử dụng xem thông tin một cách dễ dàng, sinh động. Dữ liệu chuyển giữa Web Server và Web Client thông qua nghi thức HTTP (Hypertext Transfer Protocol).

ả gười quản trị có thể xem các thông tin như các người dùng đã truy cập, các trang được truy cập, các yêu cầu được chấp nhận, các yêu cầu bị từ chối. thông qua các file có thể được lưu dưới dạng cơ sở dữ liệu.

b. FTP (File Transfer Protocol)

Sử dụng giao thức TCP để chuyển file giữa 2 máy và cũng hoạt động theo mô hình Client/Server, khi nhận được yêu cầu từ client, đầu tiên FTP Server sẽ kiểm tra tính hợp lệ của người dùng thông qua tên và mật mã. ả ếu hợp lệ, FTP Server sẽ kiểm tra quyền người dùng trên tập tin hay thư mục được xác định trên FTP Server. ả ếu hợp lệ và hệ thống file là ả TFS thì sẽ có thêm kiểm tra ở mức thư mục, tập tin theo ả TFS. Sau khi tất cả hợp lệ, người dùng sẽ được quyền tương ứng trên tập tin, thư mục đó.

Để sử dụng FTP có nhiều cách:

- Sử dụng Web Browser.

- Sử dụng Command line.
- Sử dụng từ <Run> command trong Windows.

c. Gopher

Là một dịch vụ sử dụng giao diện menu để Gopher Client tìm và chuyển bất kỳ thông tin nào mà Gopher Server đã được cấu hình. Gopher cũng sử dụng kết nối theo giao thức TCP/IP.

II. Dynamic Host Configuration Protocol (DHCP) :

Trong một mạng máy tính, việc cấp các địa chỉ IP tĩnh cố định cho các host sẽ dẫn đến tình trạng lãng phí địa chỉ IP, vì trong cùng một lúc không phải các host hoạt động đồng thời với nhau, do vậy sẽ có một số địa chỉ IP bị thừa. Để khắc phục tình trạng đó, dịch vụ DHCP đưa ra để cấp phát các địa chỉ IP động trong mạng.

Trong mạng máy tính ả T khi một máy phát ra yêu cầu về các thông tin của TCPIP thì gọi là DHCP client, còn các máy cung cấp thông tin của TCPIP gọi là DHCP server. Các máy DHCP server bắt buộc phải là Windows ả T server.

Cách cấp phát địa chỉ IP trong DHCP: Một user khi log on vào mạng, nó cần xin cấp 1 địa chỉ IP, theo 4 bước sau :

- Gởi thông báo đến tất cả các DHCP server để yêu cầu được cấp địa chỉ.
- Tất cả các DHCP server gởi trả lời địa chỉ sẽ cấp đến cho user đó.
- User chọn 1 địa chỉ trong số các địa chỉ, gởi thông báo đến server có địa chỉ được chọn.
- Server được chọn gởi thông báo khẳng định đến user mà nó cấp địa chỉ.

Quản trị các địa chỉ IP của DHCP server: Server quản trị địa chỉ thông qua thời gian thuê bao địa chỉ (lease duration). Có ba phương pháp gán địa chỉ IP cho các Worstation :

- Gán thủ công.
- Gán tự động.
- Gán động .

Trong phương pháp gán địa chỉ IP thủ công thì địa chỉ IP của DHCP client được gán thủ công bởi người quản lý mạng tại DHCP server và DHCP được sử dụng để chuyển tới DHCP client giá trị địa chỉ IP mà được định bởi người quản trị mạng

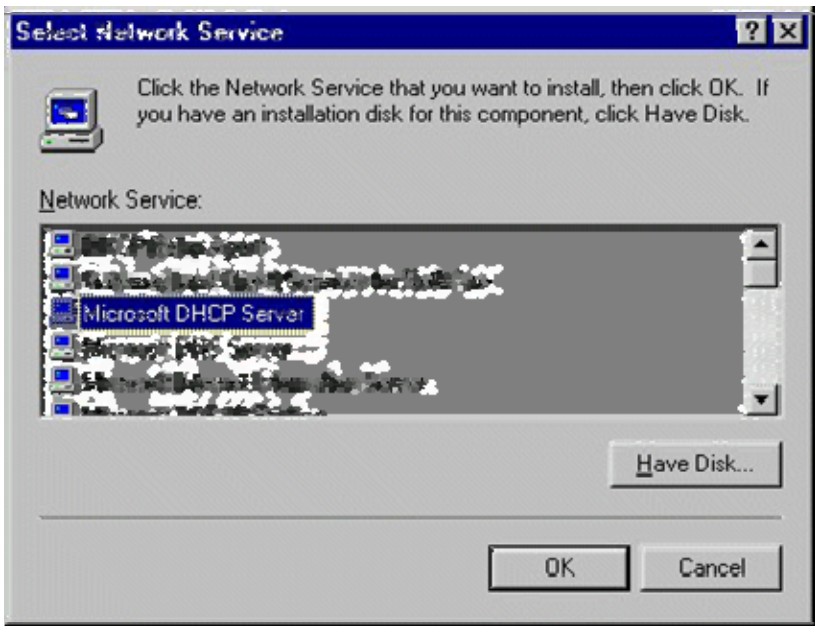
Trong phương pháp gán địa chỉ IP tự động DHCP client được gán địa chỉ IP khi lần đầu tiên nó nối vào mạng. Địa chỉ IP được gán bằng phương pháp này sẽ được gán vĩnh viễn cho DHCP client và địa chỉ này sẽ không bao giờ được sử dụng bởi một DHCP client khác

Trong phương pháp gán địa chỉ IP động thì DHCP server gán địa chỉ IP cho DHCP client tạm thời. Sau đó địa chỉ IP này sẽ được DHCP client sử dụng trong một thời gian đặc biệt. Đến khi thời gian này hết hạn thì địa chỉ IP này sẽ bị xóa mất. Sau đó nếu DHCP client cần nối kết vào mạng thì nó sẽ được cấp một địa chủ IP khác

Phương pháp gán địa chỉ IP động này đặc biệt hữu hiệu đối với những DHCP client chỉ cần địa chỉ IP tạm thời để kết nối vào mạng. Ví dụ một tình huống trên mạng có 300 users và sử dụng subnet là lớp C. Điều này cho phép trên mạng có 253 nodes trên mạng. Bởi vì mỗi computer nối kết vào mạng sử dụng TCP/IP cần có một địa chỉ IP duy nhất do đó tất cả 300 computer không thể đồng thời nối kết vào mạng. Vì vậy nếu ta sử dụng phương pháp này ta có thể sử dụng lại những IP mà đã được giải phóng từ các DHCP client khác.

Cài đặt DHCP chỉ có thể cài trên Windows ả T server mà không thể cài trên Client. Các bước thực hiện như sau:

- Login vào Server với tên Administrator .
- Click hai lần vào icon **Network** . Ta sẽ thấy hộp hội thoại **Network dialog box**



Hình 15.2: Màn hình cài đặt của DHCP

- Chọn tab **service** và click vào nút **Add** .
- Ta sẽ thấy một loạt các service của Windows ả T server nằm trong hộp hội thoại **Select Network Service**. Chọn **Microsoft DHCP server** từ danh sách các service được liệt kê ở phía dưới và nhấn **OK** và thực hiện các yêu cầu tiếp theo của Windows ả T.

Để cập nhật và khai thác DHCP server chúng ta chọn mục DHCP manager trong ả etwrok Administrator Tools.

III. Dịch vụ Domain Name Service (DNS)

Hiện nay trong mạng Internet số lượng các nút (host) lên tới hàng triệu nên chúng ta không thể nhớ hết địa chỉ IP được, Mỗi host ngoài địa chỉ IP còn có một cái tên phân biệt, Dẩ S là 1 cơ sở dữ liệu phân tán cung cấp ánh xạ từ tên host đếùn địa chỉ IP. Khi đưa ra 1 tên host, Dẩ S server sẽ trả về địa chỉ IP hay 1 số thông tin của host đó. Điều này cho phép người quản lý mạng dễ dàng trong việc chọn tên cho host của mình

DNS server được dùng trong các trường hợp sau :

- Chúng ta muốn có 1 tên domain riêng trên Interner để có thể tạo, tách rời các domain con bên trong nó.

- Chúng ta cần 1 dịch vụ Dẩ S để điều khiển cục bộ nhằm tăng tính linh hoạt cho domain cục bộ của bạn.

- Chúng ta cần một bức tường lửa để bảo vệ không cho người ngoài thâm nhập vào hệ thống mạng nội bộ của mình

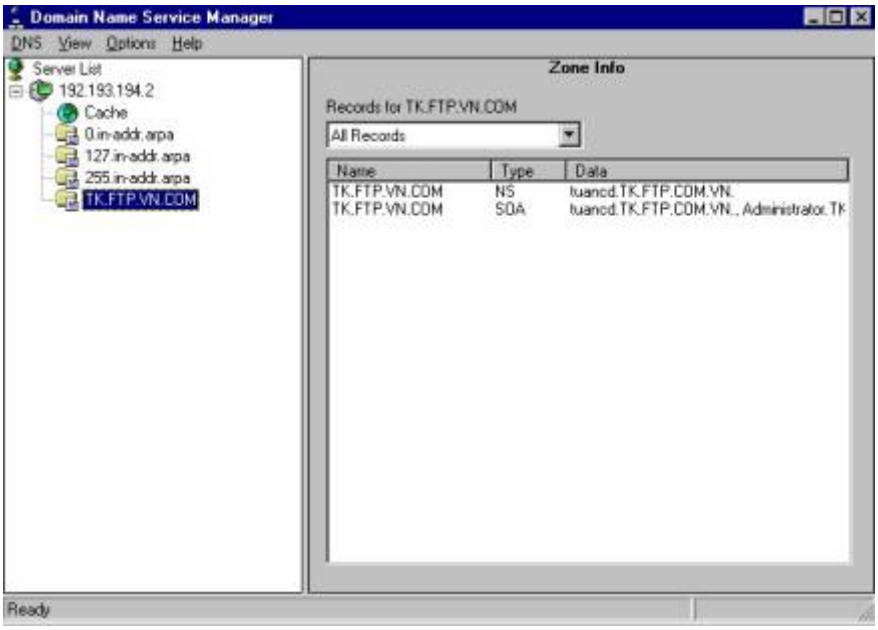
Có thể quản lý trực tiếp bằng các trình soạn thảo text để tạo và sửa đổi các file hoặc dùng Dẩ S manager để tạo và quản lý các đối tượng của Dẩ S như: Servers, Zone, Các mẫu tin, các Domains, Tích hợp với Win, .

Cài đặt Dẩ S chỉ có thể cài trên Windows ẩ T server mà không thể cài trên Client. Các bước thực hiện như sau:

- Login vào Server với tên **Administrator**.

- Click hai lần vào icon **Network**. Ta sẽ thấy hộp hội thoại **Network dialog box** tương tụ như trên và lựa chọn **Microsoft DNS Server**.

Để cập nhật và khai thác Dẩ S server chúng ta chọn mục **DNS manager trong Netwrok Administrator Tools**. Hộp hội thoại sau đây sẽ hiện ra



Hình 15.3: Màn hình DNS Manager

Mỗi một tập hợp thông tin chứa trong **DNS database** được coi như là **Resourse record**. ả những **Resourse record** cần thiết sẽ được liệt kê dươi đây:

Tên Record	Mô tả
A (Address)	Dẫn đường một tên host computer hay tên của một thiết bị mạng khác trên mạng tới một địa chỉ IP trong Dả S zone
Cả AME ()	Tạo một tên Alias cho tên một host computer trên mạng
MX ()	Định nghĩa một sự trao đổi mail cho host computer đó
ả S (name server)	Định nghĩa tên server Dả S cho Dả S domain
PTR (Pointer)	Dẫn đường một địa chỉ IP đến tên host trong Dả S server zone
SOA (Start of authority)	Hiển thị rằng tên server Dả S này thì chứa những thông tin tốt nhất

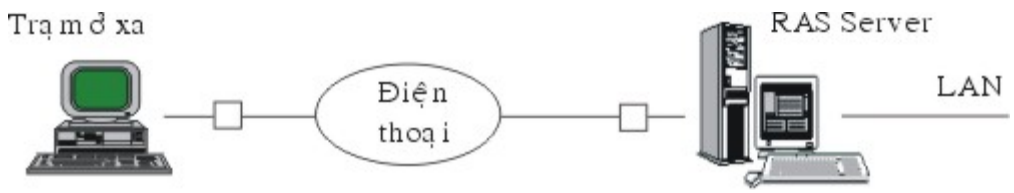
IV. Remote Access Service (RAS)

ả goài những liên kết tại chỗ với mạng cục bộ (LAả) các nối kết từ xa vào mạng LAả hiện đang là những yêu cầu cần thiết của người sử dụng. Việc liên kết đó cho phép một máy từ xa như của một người sử dụng tại nhà có thể qua đường dây điện thoại thâm nhập vào một mạng LAả và sử dụng tài nguyên của nó. Cách thông dụng nhất hiện nay là dùng modem để có thể truyền trên đường dây điện thoại.

Windows ả T cung cấp Dịch vụ Remote access Service cho phép các máy trạm có thể nối với tài nguyên của Windows ả T server thông qua đường dây điện thoại. RAS cho phép truyền nối với các server, điều hành các user và các server, thực hiện các chương trình khai thác số liệu, thiết lập sự an toàn trên mạng. .

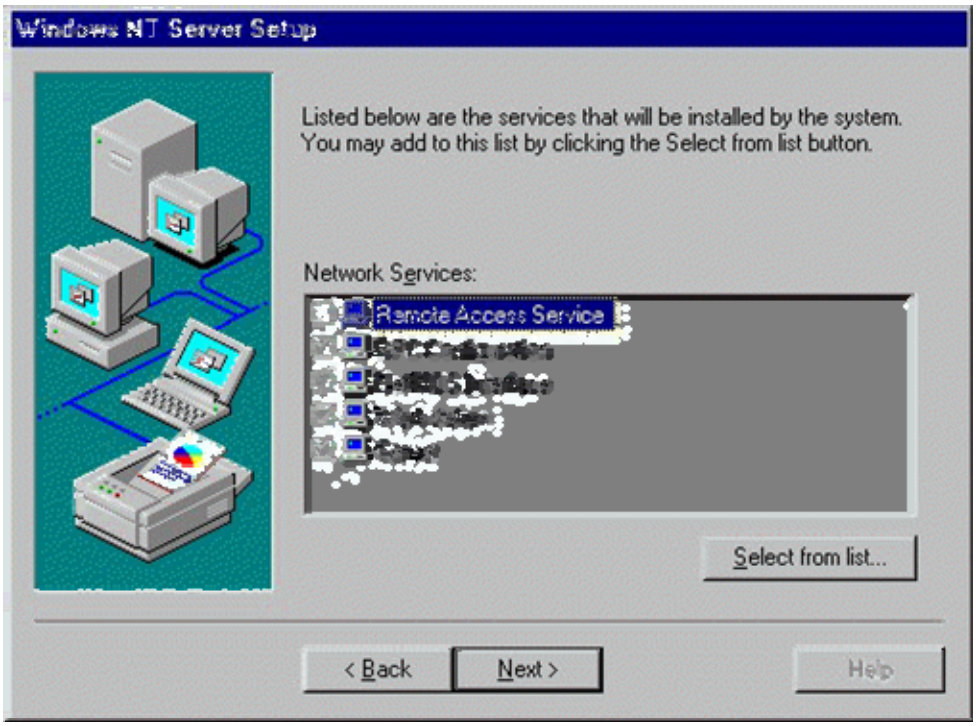
Máy trạm có thể được nối với server có dịch vụ RAS thông qua modem hoạc pull modem, cable null modem (RS232) hoặc X.25 network.

Khi đã cài đặt dịch vụ RAS, cần phải đảm bảo quyền truy nhập từ xa cho người sử dụng bằng tiện ích remote access amind để gán quyền hoặc có thể đăng ký người sử dụng ở remote access server. RAS cũng có cơ chế đảm bảo an toàn cho tài nguyên bằng cách kiểm soát các yếu tố sau: quyền sử dụng, kiểm tra mã số, xác nhận người sử dụng, đăng ký sử dụng tài nguyên và xác nhận quyền gọi lại.



Hình 15.4: Mô hình truy cập từ xa bằng dịch vụ RAS

Để cài đặt RAS chúng ta lựa chọn yêu cầu hộp Windows ả T server setup hiện ra lúc cài đặt hệ điều hành Windows ả T.



Với RAS tất cả các ứng dụng đều thực hiện trên máy từ xa, thay vì kết nối với mạng thông qua card mạng và đường dây mạng thì máy ở xa sẽ liên kết qua modem tới một RAS Server. Tất cả dữ liệu cần thiết được truyền qua đường điện thoại, mặc dù tốc độ truyền qua modem chậm hơn so với qua card mạng nhưng với những tác vụ của LAả không phải bao giờ dữ liệu cũng truyền nhiều.

Với những khả năng to lớn của mình trong các dịch vụ mạng, hệ điều hành Windows ả T là một trong những hệ điều hành mạng tốt nhất hiện nay. Hệ điều hành Windows ả T vừa cho phép giao lưu giữa các máy trong mạng, vừa cho phép truy nhập từ xa, cho phép truyền file, vừa đáp ứng cho mạng cục bộ (LAả) vừa đáp ứng cho mạng diện rộng (WAả) như Intranet, Internet. Với những khả năng như vậy hiện nay hệ điều hành Windows ả T đã có những vị trí vững chắc trong việc cung cấp các giải pháp mạng trên thế giới.

Mục lục

CHƯƠNG 1 - GIỚI THIỆU CHUNG ---- 7

1.1 MẠNG TRUYỀN THÔNG VÀ CÔNG NGHỆ MẠNG ---- 7
1.1.1 Giới thiệu chung ---- 7
1.1.2 Mạng máy tính ---- 9
1.1.3. Phân loại mạng máy tính ---- 13
1.1.4 Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng ---- 23
1.1.5 Chuẩn mạng ---- 25
1.2 MÔ HÌNH OSI ---- 26
1.2.1 Mô hình ---- 27
1.2.2 Chức năng các tầng ---- 32
1.2.3 Bộ giao thức TCP/IP – Mô hình Internet ---- 43
1.3 PHƯƠNG PHÁP TIẾP CẬN ---- 44

CHƯƠNG 2 - TẦNG ỨNG DỤNG ---- 45

2.1 GIAO THỨC TẦNG ỨNG DỤNG ---- 45
2.1.1 Giao thức tầng ứng dụng ---- 46
2.1.2 Các yêu cầu của ứng dụng ---- 51
2.1.3 Dịch vụ của các giao thức giao vận Internet ---- 53
2.1.4 Một số ứng dụng phổ biến ---- 56
2. 2 WORLD WIDE WEB: HTTP ---- 57
2.2.1 Tổng quan về HTTP ---- 58
2.2.2 Kết nối liên tục và không liên tục (persistent / nonpersistent) ---- 61
2.2.3 Khuôn dạng thông điệp HTTP ---- 63
2.2.4 Tương tác giữa người dùng Hrver-server ---- 67
2.2.5 GET có điều kiện (Conditional GET) ---- 69
2.2.6 Web cache ---- 71
2.2.7 Web động ---- 73
2.3 TRUYỀN FILE (FILE TRANSFER) FTP ---- 82
2.3.1 Các lệnh FTP (FTP Commands) ---- 84
2.4 - THƯ TÍN ĐIỆN TỬ (E-mail) TRÊN INTERNET ---- 85
2.4.1 SMTP ---- 87

2.4.2 So sánh SMTP với HTTP ---- 90
2.4.3 Khuôn dạng thư và chuẩn MIME ---- 91
2.4.4 Giao thức truy nhập mail. ---- 97
2. 5 DỊCH VỤ TÊN MIỀN - DNS. ---- 102
2.5.1 Các dịch vụ của DNS ---- 103
2.5.2 Cơ chế hoạt động của DNS ---- 105
2.5.3 Bản ghi DNS ---- 112
2.5.4 Thông điệp DNS ---- 113
2.6 CÁC ỨNG DỤNG THEO KIẾN TRÚC NGANG HÀNG ---- 115
2.6.1 Nhắn tin tức thì ---- 117
2.6.6. Kiến trúc Hệ thống MSN. ---- 121
2.6.3 Kiến trúc chia sẻ file ngang hàng Gnutella ---- 128
2.7 LẬP TRÌNH SOCKET ---- 134
2.7.1 Các hàm thao tác trên Socket ---- 135
2.7.2 Ví dụ một chương trình client/server đơn giản ---- 138
2.7.3 Web server đơn giản ---- 140

CHƯƠNG 3 - TẦNG GIAO VẬN ---- 148

3.1 DỊCH VỤ VÀ NGUYÊN TẮC CỦA TẦNG GIAO VẬN ---- 148
3.1.1 Quan hệ giữa tầng giao vận và tầng mạng ---- 150
3.1.2 Tổng quan về tầng giao vận trong Internet. ---- 152
3.2 DỊCH VỤ DỒN KÊNH, PHÂN KÊNH ---- 153
3.3 UDP – GIAO THỨC KHÔNG HƯỚNG NỐI ---- 158
3.3.1 Cấu trúc UDP segment ---- 162
3.3.2 UDP checksum ---- 162
3.4 CÁC NGUYÊN TẮC TRUYỀN DỮ LIỆU TIN CẬY ---- 164
3.4.1 Xây dựng giao thức truyền dữ liệu tin cậy ---- 165
3.4.2 Giao thức truyền dữ liệu tin cậy liên tục (Pipeline) ---- 175
3.4.3 Go-back-N (GBN) ---- 178
3.4.4 Giao thức lặp lại có lựa chọn (Selective Repeat) ---- 184
3.5 TCP – GIAO THỨC GIAO VẬN HƯỚNG NỐI ---- 189
3.5.1 Kết nối TCP ---- 190
3.5.2 Cấu trúc TCP Segment ---- 193
3.5.3 Số thứ tự và Số biên nhận ---- 195
3.5.4 Telnet: Một ví dụ về số thứ tự và số biên nhận ---- 196
3.5.5 Truyền dữ liệu tin cậy ---- 199

3.5.6 Kiểm soát lưu lượng ---- 205
3.5.7 Quản lý kết nối TCP ---- 207
3.6 KIỂM SOÁT TẮC NGHẼN TRONG TCP ---- 211

CHƯƠNG 4 - TẦNG MẠNG ---- 216

4.1 CÁC MÔ HÌNH DỊCH VỤ CỦA TẦNG MẠNG ---- 216
4.1.1 Mô hình dịch vụ mạng ---- 218
4.1.2 Nguồn gốc của dịch vụ chuyển mạch gói và chuyển mạch ảo. ---- 223
4.2 CÁC NGUYÊN LÝ ĐỊNH TUYẾN ---- 224
4.2.1. Thuật toán định tuyến link state. ---- 227
4.2.2. Thuật toán Distance vector. ---- 231
4.3 ĐỊNH TUYẾN PHÂN CẤP ---- 236
4.4 INTERNET PTOTOCOL ---- 238
4.4.1 Địa chỉ IPv4 ---- 240
4.4.2 Chuyển datagram từ nguồn tới đích: vấn đề địa chỉ và định tuyến ---- 248
4.4.3 Khuôn dạng gói dữ liệu IP ---- 251
4.4.4 Phân mảnh (Fragmentation) và Hợp nhất (Reassembly) gói tin IP ---- 254
4.4.5 Giao thức kiểm soát lỗi ICMP (Internet Control Message Protocol) ---- 258
4.5 ĐỊNH TUYẾN TRÊN INTERNET ---- 259
4.5.1 Định tuyến trong một miền (Intra-AS routing) (Định tuyến nội miền) ---- 260
4.5.2 Định tuyến giữa các miền (Inter-AS routing) (Định tuyến liên miền) ---- 264
4.6 CẤU TẠO CỦA THIẾT BỊ ĐỊNH TUYẾN (ROUTER) ---- 264
4.6.1 Cổng vào (Input port) ---- 266
4.6.2 Kết cấu chuyển (Switching fabric) ---- 268
4.6.3 Cổng ra (Output port) ---- 270
4.6.4 Hàng đợi ở router ---- 270
4.7 IPv6 ---- 273
4.7.1 Định dạng gói tin IPv6 ---- 273
4.7.2 Chuyển từ IPv4 sang IPv6 ---- 276
4.8 CƠ CHẾ DỊCH CHUYỂN ĐỊA CHỈ (NAT) ---- 279
4.9 KIỂM SOÁT TẮC NGHẼN ---- 285
4.9.1 Các nguyên lý Kiểm soát tắc nghẽn. ---- 286
4.9.2 Chính sách ngăn chặn tắc nghẽn ---- 289
4.9.3 Kiểm soát tắc nghẽn trong mạch ảo ---- 290
4.9.4 Kiểm soát tắc nghẽn trong mạng chuyển mạch gói. ---- 292
4.9.5 Cắt tải ---- 293

CHƯƠNG 5 -TẦNG LIÊN KẾT DỮ LIỆU ----------295

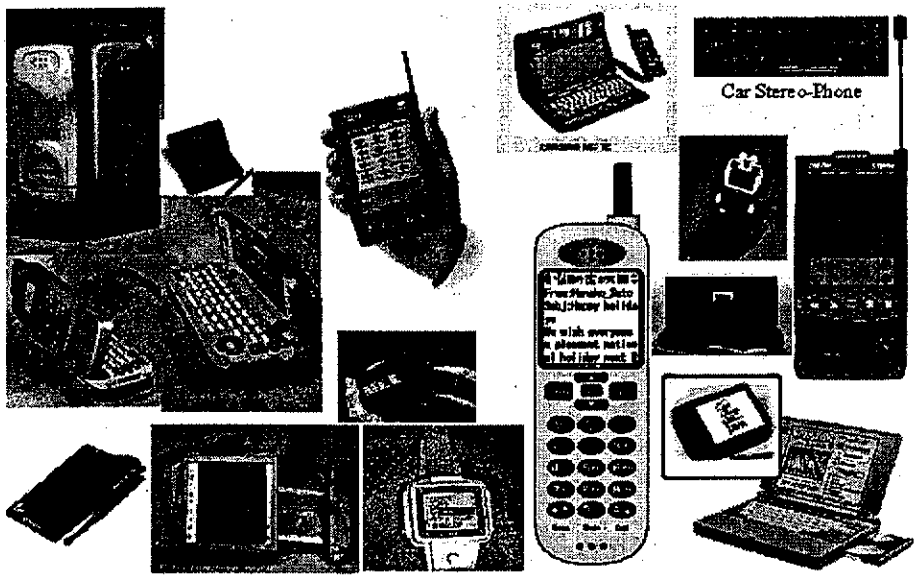
5.1 CÁC KHÁI NIỆM CHUNG, DỊCH VỤ CỦA TẦNG DATALINK----------295
5.1.1 Những dịch vụ của tầng liên kết dữ liệu----------296
5.1.2 Bộ điều hợp (Adapter)----------300
5.2 KỸ THUẬT PHÁT HIỆN VÀ SỬA LỖI----------302
5.2.1 Kiểm tra tính chẵn lẻ----------304
5.2.2 Phương pháp tính tổng kiểm tra (checksum) ----------306
5.2.3 Kiểm tra dư thừa vòng (CRC)----------306
5.2.4 Sửa lỗi bằng mã Hamming ----------309
5.3. GIAO THỨC ĐA TRUY CẬP VÀ MẠNG CỤC BỘ----------313
5.3.1 Giao thức phân chia kênh truyền (channel partitioning) ----------317
5.3.2 Giao thức truy cập ngẫu nhiên (random access) ----------318
5.3.3 Giao thức truy cập lần lượt (Taking – turns) ----------326
5.3.4 Mạng cục bộ LAN (Local Area Network) ----------327
5.4 ĐỊA CHỈ LAN VÀ ARP ----------328
5.4.1 Địa chỉ LAN----------329
5.4.2. Giao thức giải mã địa chỉ (ARP)----------331
5.5. ETHERNET----------336
5.5.1 Những khái niệm cơ bản của Ethernet----------338
5.2.2 CSMA/CD: Giao thức đa truy cập của Ethernet----------342
5.5.3 Những công nghệ Ethernet ----------345
5.6 HUB, BRIDGE VÀ SWITCH----------349
5.6.1 Hub----------349
5.6.2 Bridge----------351
5.6.3. Switch----------360
5.8. MẠNG LAN KHÔNG DÂY ----------363
5.8.1 Giới thiệu chung----------363
5.8.2 Lớp giao thức IEEE 802.11----------366
5.8.3 Một số vấn đề hay gặp đối với mạng không dây----------369
5.9. PPP – GIAO THỨC ĐIỂM NỐI ĐIỂM ----------371
5.9.1 Khuôn dạng gói gữ liệu (Frame PPP)----------373
5.9.2 Giao thức điều khiển đường truyền PPP (LCP) và kiểm soát mạng ----------375
5.10 MẠNG RIÊNG ẢO (VPN)----------378
5.10.1 Các mạng riêng ảo truyền thống ----------378
5.10.2 Mô phỏng giả dây dẫn (Pseudowire Emulation Overview) ----------381
TÀI LIỆU THAM KHẢO ----------387

Chương 1
GIỚI THIỆU CHUNG

1.1 MẠNG TRUYỀN THÔNG VÀ CÔNG NGHỆ MẠNG

1.1.1 Giới thiệu chung

Truyền thông máy tính (computer communications) là quá trình truyền dữ liệu từ thiết bị này sang thiết bị khác. Trước đây chúng ta thường hiểu thiết bị là máy tính, nhưng ngày nay thiết bị (end-system, device) không chỉ đơn thuần là máy tính mà bao gồm nhiều chủng loại thiết bị khác, ví dụ điện thoại di động, máy tính PAM,... (Xem Hình 1.1). Số lượng các kiểu thiết bị có khả năng lấy thông tin từ Internet ngày càng tăng. Một từ phổ biến có nghĩa tương tự như vậy là truyền dữ liệu. Mặc dù hai cụm từ này có thể sử dụng thay thế cho nhau, một số người coi thuật ngữ dữ liệu (data) chỉ bao gồm những sự kiện đơn giản và thô (chưa được xử lý), và sử dụng thuật ngữ thông tin (information) để chỉ việc tổ chức những sự kiện này thành dạng thông tin có nghĩa đối với con người.



Hình 1.1 Các thiết bị có khả năng kết nối Internet

Khái niệm **mạng (networking)** chỉ khái niệm kết nối các thiết bị lại với nhau nhằm mục đích chia sẻ thông tin. Khái niệm mạng liên quan đến nhiều vấn đề, bao gồm:

Giao thức truyền thông (protocol): mô tả những nguyên tắc mà tất cả các thành phần mạng cần tuân thủ để có thể trao đổi được với nhau;

Topo (mô hình ghép nối mạng/hình trạng mạng): mô tả cách thức nối các thiết bị với nhau.

Địa chỉ: mô tả cách thức định vị một đối tượng trên mạng.

Định tuyến (routing): mô tả cách thức dữ liệu truyền từ thiết bị này sang thiết bị khác trên mạng.

Tính tin cậy (reliability): giải quyết tính toàn vẹn của dữ liệu, đảm bảo dữ liệu nhận được chính xác như dữ liệu gửi đi.

Khả năng liên tác (interoperability): chỉ mức độ các sản phẩm phần mềm và phần cứng của các hãng sản xuất khác nhau có thể làm việc cùng nhau.

An ninh (security): đảm bảo an toàn hoặc bảo vệ tất cả các thành phần của mạng.

Chuẩn (standard): thiết lập các quy tắc và luật lệ cụ thể cần phải tuân theo.

Trên thực tế, khái niệm mạng xuất hiện ở nhiều kiểu ứng dụng khác nhau. Ví dụ, trong công nghiệp giải trí, các công ty truyền thanh, truyền hình, và công ty cáp đều có những mạng độc lập riêng của mình với nhiều trạm phát. Thông qua những mạng này, các chương trình tin tức, thể thao, điện ảnh, phim truyện... được dùng chung giữa các trạm phát. Mạng truyền thông ra đời sớm nhất và phổ biến nhất là mạng điện thoại. Khi nói đến mạng điện thoại, người ta muốn nhắc đến hệ thống điện thoại kiểu cũ (plain old telephone system - POTS) hoặc mạng điện thoại chuyển mạch công cộng (PSTN – public switched telephone network). Mạng PSTN mô tả hệ thống điện thoại truyền thống dựa trên tín hiệu tương tự được sử dụng để truyền tiếng nói. Một mạng truyền thông khá quen thuộc ngày nay là mạng máy tính Internet - là một tập hợp các mạng, hay mạng mạng.

1.1.2 Mạng máy tính

Mạng bao gồm nhiều thành phần, các thành phần được nối với nhau theo một cách thức nào đó và cùng sử dụng chung một ngôn ngữ:

Các thiết bị đầu cuối (end system) kết nối với nhau tạo thành mạng có thể là các máy tính (computer) hoặc các thiết bị khác. Ngày càng có nhiều loại thiết bị có khả năng kết nối vào mạng máy tính như điện thoại di động, PDA, ti vi...

Môi trường truyền (media) thực hiện việc truyền dẫn các tín hiệu vật lý. Môi trường truyền có thể là các loại dây dẫn (cáp), sóng (đối với các mạng không dây).

Giao thức (protocol) là quy tắc quy định cách thức trao đổi dữ liệu giữa các thực thể.

Nói chung, ba khái niệm trên đưa đến một định nghĩa chuẩn về mạng máy tính như sau:

Mạng máy tính là tập hợp các máy tính và các thiết bị phụ trợ khác sử dụng chung một nhóm giao thức để chia sẻ tài nguyên thông qua các phương tiện truyền thông mạng.

Các thành phần mạng: thiết bị, nút, máy tính

Theo nghĩa chung nhất, thuật ngữ **thiết bị** (device) chỉ bất cứ một thực thể phần cứng nào, chẳng hạn các thiết bị đầu cuối, máy in, máy tính, hoặc một thiết bị phần cứng đặc biệt liên quan đến mạng, ví dụ máy chủ (server), repeater (bộ lặp), bridge (cầu), switch, router (bộ định tuyến), và rất nhiều thiết bị đặc biệt khác. Tất cả các kiểu thiết bị này sẽ được thảo luận chi tiết ở các chương sau.

Nói chung có nhiều phương pháp gán cho thiết bị mạng một định danh duy nhất, thường thì thiết bị được chính hãng sản xuất gắn một số nhận dạng duy nhất. Việc này tương tự như việc in số seri trên tivi hoặc các đồ dùng điện tử khác. Ví dụ mỗi card Ethernet được hãng sản xuất gán cho một địa chỉ duy nhất - địa chỉ này không trùng với bất kỳ card Ethernet nào khác.

Khi mô tả các thành phần mạng, cần phân biệt giữa khái niệm thiết bị (device) và máy tính (computer). Nếu xét ở khía cạnh thiết bị máy tính thường được gọi là host (hoặc server) hay trạm làm việc (workstation) (cũng còn được gọi là desktop hay client). Thuật ngữ này thường dùng để chỉ những hệ thống máy tính có cài đặt hệ điều hành riêng của mình (ví dụ Windows, UNIX). Vì vậy workstation có thể là máy tính cá nhân như máy Apple Macintosh, hoặc bất cứ máy tính họ Intel nào (thường được gọi là IBM-PC); cũng có thể là một workstation đồ họa (ví dụ các workstation đồ họa được sản xuất bởi Sun Microsystems, Silicon Graphics, IBM, Hewlett-Packard, Compaq Computer Corporation), một superminicomputer như Compaq's VAX hay một hệ thống IBM AS/400, một super-microcomputer như Compaq's Alpha; hoặc có thể là một máy tính lớn (mainframe) như IBM ES-9000.

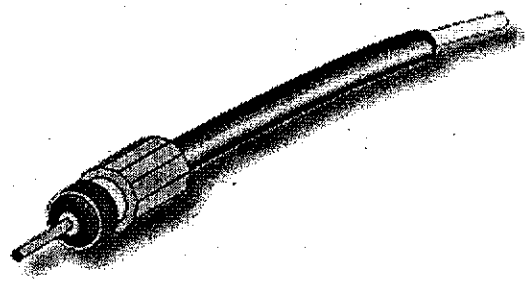
Phương tiện và các giao thức truyền thông mạng

Để chia sẻ thông tin và sử dụng dịch vụ trên mạng, các thành phần của mạng phải có khả năng truyền thông được với nhau. Để đáp ứng được yêu cầu này, chúng ta phải xét tới hai tiêu chí cụ thể của mạng: khả năng

liên kết (connectivity) và ngôn ngữ (language). Khả năng liên kết chỉ đường truyền hoặc kết nối vật lý giữa các thành phần; ngôn ngữ chỉ một bảng từ vựng cùng các quy tắc truyền thông mà các thành phần phải tuân thủ.

Phương tiện truyền thông (media)

Môi trường vật lý sử dụng để kết nối các thành phần của mạng thường được gọi là môi trường truyền thông (medium, media). Môi trường truyền thông mạng được chia thành hai loại: cáp (cable) và không dây (wireless). Ví dụ, cáp truyền thông có thể là cáp xoắn đôi (twisted-pair), cáp đồng trục (coaxial), và cáp sợi quang (fiber-optic cable)... Truyền thông không dây có thể là sóng radio (gồm sóng cực ngắn hay việc truyền thông qua vệ tinh), bức xạ hồng ngoại. Các môi trường truyền thông mạng được thảo luận chi tiết trong chương 5.



Hình 1.2 Môi trường truyền: Sợi cáp quang

Giao thức (Protocols). Ngôn ngữ được sử dụng bởi các thực thể mạng gọi là giao thức truyền thông mạng. Các bên truyền thông "hiểu nhau" do giao thức định nghĩa một ngôn ngữ chung giữa các thành phần mạng. Từ ý nghĩa khái quát này có thể hiểu giao thức truyền thông mạng là các thủ tục, quy tắc hoặc các đặc tả chính thức đã được chấp nhận nhằm xác định hành vi và ngôn ngữ trao đổi giữa các bên. Nói chung trong cuộc sống hàng ngày, chúng ta cũng áp dụng những quy tắc nào đó. Ví dụ, khi đi đến những nơi đòi hỏi tính trang trọng, mọi người phải tuân theo những nghi thức đặc biệt về ăn mặc (ví dụ nam giới phải mặc áo vét có thắt caravat). Nhưng khi đến các quán ăn bình dân thì không cần ăn mặc trang trọng như vậy. Trong mạng và truyền thông máy tính, giao thức mạng là bản đặc tả chính thức quy định cách thức "xử sự" của các thực thể tham gia truyền thông. Ở đây khái

niệm thực thể bao gồm cả thiết bị phần cứng cũng như tiến trình phần mềm. Giao thức mạng cũng định nghĩa khuôn dạng dữ liệu được trao đổi giữa các bên. Nói một cách ngắn gọn, giao thức mạng định nghĩa bảng từ vựng và các quy tắc áp dụng truyền thông dữ liệu.

Không có môi trường truyền, không thể trao đổi thông tin giữa các thực thể mạng; không có ngôn ngữ chung, không thể hiểu được nhau. Vì vậy, đường truyền cung cấp môi trường để thực hiện truyền thông, trong khi đó ngôn ngữ chung đảm bảo hai bên truyền thông hiểu được nhau. Điều này cũng giống như cuộc nói chuyện điện thoại giữa một người chỉ nói được tiếng Ý với một người chỉ nói được tiếng Nga. Nếu không có đường điện thoại (đường truyền mạng) cho cuộc nói chuyện thì hai người không thể nói chuyện được với nhau (không thể trao đổi dữ liệu). Đã có đường điện thoại rồi, lúc này hai người có thể nói và nghe thấy giọng nói của nhau (truyền dữ liệu được thực hiện) nhưng họ không giao tiếp được với nhau vì không ai trong số họ hiểu được ngôn ngữ của người kia - họ nói chuyện bằng hai thứ tiếng khác nhau.

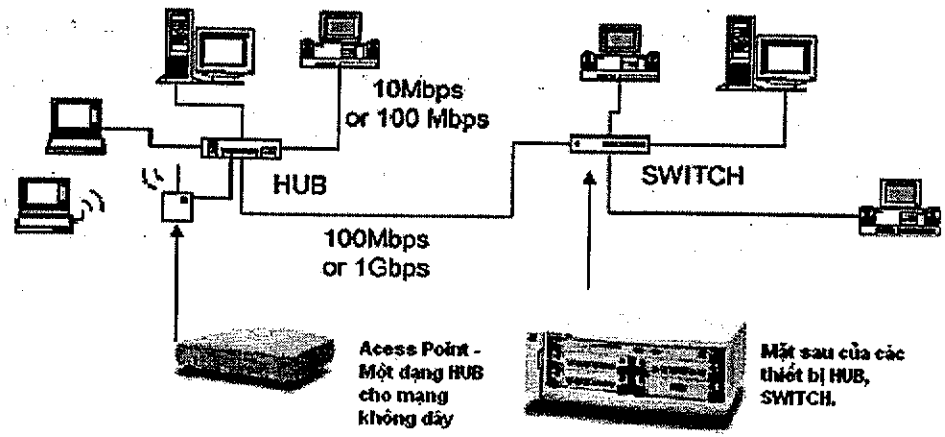
Ví dụ TCP/IP là một giao thức mạng quen thuộc - một trong những giao thức của bộ giao thức TCP/IP (Transmission Control Protocol/Internet Protocol) TCP/IP được coi là xương sống của Internet. Tuy tên gọi TCP/IP chỉ gồm hai giao thức cụ thể là TCP và IP nhưng thường được sử dụng để chỉ nhóm gồm nhiều giao thức khác ngoài TCP và IP. Tập hợp các giao thức này được gọi là bộ giao thức TCP/IP. Có thể kể đến một số giao thức trong bộ giao thức TCP/IP như FTP (Transfer Protocol) định nghĩa cách chuyển file; HTTP (the Hypertext Transport Protocol) được dùng cho World Wide Web (WWW), định nghĩa cách các server cần phải truyền các tài liệu (trang Web) tới các client (Web browser) như thế nào. Ngoài ra cũng phải kể đến ba giao thức được sử dụng cho thư điện tử (email) là Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP) và Internet Mail Access Protocol (IMAP). Ngày nay các mạng sử dụng rất nhiều giao thức khác nhau từ đơn giản đến phức tạp. Có thể nói các giao thức là "keo dán" ràng buộc mạng máy tính lại với nhau bởi vì chúng định nghĩa cách thực hiện các hoạt động cụ thể.

Một số bộ giao thức khác có thể kể đến là **AppleTalk** - bộ giao thức mạng của công ty Apple Computer, đầu tiên chỉ sử dụng cho các máy tính

Macintosh, không thể phục vụ cho các hệ điều hành khác; bộ giao thức của Hệ điều hành **Windows 2000** của Microsoft; **DECnet** của Digital Equipment Corporation (hiện giờ là Compad) - sử dụng giao thức mạng Digital Network Architecture (DNA). DECnet được thiết kế cho các máy VAX hoặc Alpha chạy dưới hệ điều hành Open-VMS, hoặc cho các hệ thống DEC dùng hệ điều hành DEC trước đây, ví dụ như RSX-11M, RT-11, RSTS/E cũng như một số hệ điều hành phổ biến khác như MS-DOS, Windows, và một vài biến thể của Unix. Mạng máy tính đôi khi cũng được đặt tên theo giao thức chúng dùng. Ví dụ, một mạng gồm các thiết bị hỗ trợ Apple Talk thường được gọi là một mạng Apple Talk. Tương tự, mạng TCP/IP là mạng của các thiết bị được liên kết với nhau và sử dụng bộ giao thức TCP/IP để truyền thông.

1.1.3. Phân loại mạng máy tính

Có rất nhiều kiểu mạng máy tính khác nhau. Việc phân loại chúng thường dựa trên các đặc điểm chung. Ví dụ, mạng máy tính thường được phân loại theo vùng địa lý (diện hoạt động) (ví dụ: mạng cục bộ, mạng diện rộng,...); theo topo (mô hình ghép nối mạng) (ví dụ: điểm-điểm (point-to-point) hay broadcast); hoặc theo kiểu đường truyền thông mà mạng sử dụng và cách chuyển dữ liệu đi (ví dụ mạng chuyển mạch ảo, hay chuyển mạch gói).



Hình 1.3 Một mạng LAN đơn giản

Phân loại mạng theo diện hoạt động

Nếu phân loại theo diện hoạt động, mạng máy tính có thể được phân chia thành:

Mạng cục bộ (Local Area Network - LAN);

Mạng diện rộng (Wide Area network - WAN);

Mạng đô thị (Metropolitan Area network - MAN);

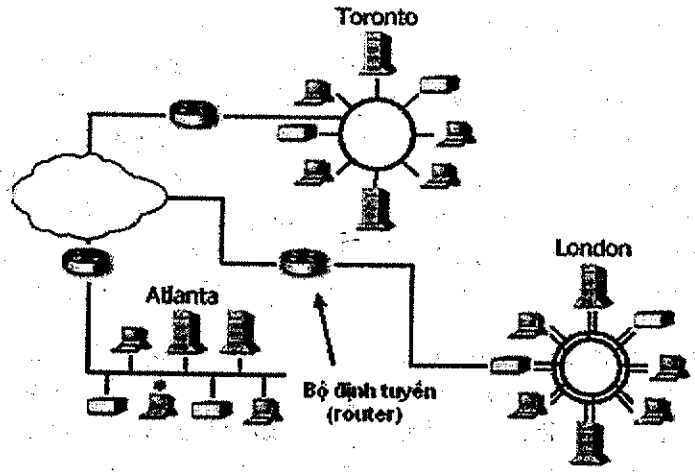
Mạng toàn cầu (Global Area network - GAN);

Mạng cá nhân (Personal Area network - PAN);

Mạng lưu trữ (Storage Area Network - SAN).

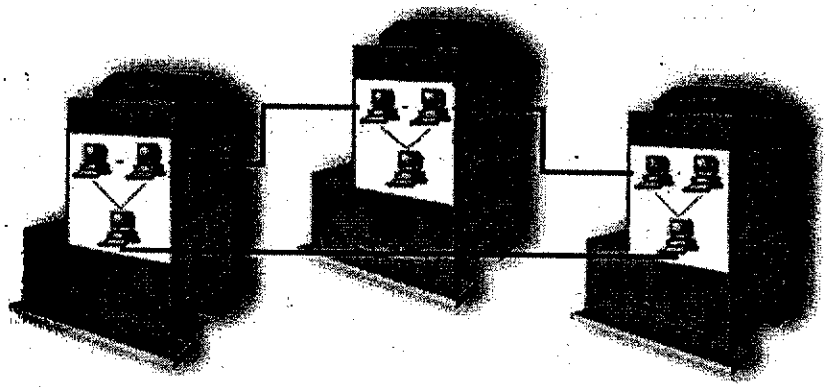
Mạng cục bộ (LAN) liên kết các tài nguyên máy tính trong một vùng địa lý có kích thước hạn chế. Đó có thể là một phòng, vài phòng trong một tòa nhà, hoặc vài tòa nhà trong một khu nhà. Cụm từ "kích thước hạn chế" không được xác định cụ thể nên một số người xác định phạm vi của mạng LAN bằng cách định bán kính nằm trong khoảng vài chục mét đến vài km. IEEE (Institute of Electrical and Electronics Engineers) xác định bán kính của mạng LAN nhỏ hơn 10km. Ví dụ về một số công nghệ mạng LAN: Ethernet/802.3, Token Ring, mạng FDDI (Fiber Distributed Data Interface).

Mạng diện rộng (WAN) liên kết các tài nguyên máy tính trong một vùng địa lý rộng (có bán kính trên 100km) như thị xã, thành phố, tỉnh/bang, quốc gia. Có thể coi mạng WAN gồm nhiều mạng LAN khác nhau. Ví dụ một số công nghệ mạng WAN: ISDN (Integrated Services Data Network), frame relay, SMDS (Switched Multimegabit Data Service) và ATM (Asynchronous Transfer Mode).



Hình 1. 4 Mạng WAN - kết hợp của nhiều mạng LAN qua các router

Một số người phân biệt kỹ hơn giữa mạng LAN và WAN. Do vậy xuất hiện phân loại ***Mạng đô thị (MAN)***. MAN liên kết các tài nguyên máy tính trong một thành phố. Giả sử có một công ty kinh doanh có nhiều tòa nhà trong tỉnh/thành phố. Mỗi tòa nhà có một mạng LAN riêng, những mạng LAN này được kết nối với nhau, kết quả ta có một mạng MAN vì tất cả các tòa nhà là ở trong cùng một tỉnh/thành phố. Nhìn chung, MAN được dùng để chỉ các mạng có diện hoạt động lớn hơn LAN nhưng nhỏ hơn WAN.



Hình 1. 5 Mạng MAN - kết hợp nhiều mạng LAN trong một khu vực địa lý

Một loại mạng nữa là **Mạng cá nhân (PAN)**, chỉ mạng máy tính nhỏ sử dụng trong gia đình. Giá máy tính ngày càng rẻ làm cho số gia đình có

nhiều máy tính ngày càng tăng nhanh, dẫn đến nhu cầu xuất hiện mạng PAN vì người ta nhận ra ưu điểm của việc kết nối các máy tính trong gia đình. Ví dụ, có thể nối các máy tính trong nhà đến cùng một máy in, không cần phải mua máy in cho mỗi máy tính. PAN cũng cho phép cả gia đình sử dụng một máy làm file server chứa tất cả phần mềm ứng dụng và dữ liệu người dùng; có thể truy cập đến server này từ bất cứ máy nào nối với mạng gia đình. PAN cũng giúp các thành viên trong gia đình truy cập đến bất cứ tài nguyên nào được dùng chung trong gia đình từ các vị trí khác nhau trong nhà.

Mạng toàn cầu (GAN) là mạng của các mạng WAN, trải rộng trên phạm vi toàn cầu. Ví dụ, nhiều công ty xuyên quốc gia hoạt động trên nhiều nước trên thế giới. Việc kết nối mạng của các công ty con lại với nhau tạo thành mạng GAN. Mạng toàn cầu Internet cũng là một mạng GAN đặc biệt.

Phân loại mạng theo mô hình ghép nối

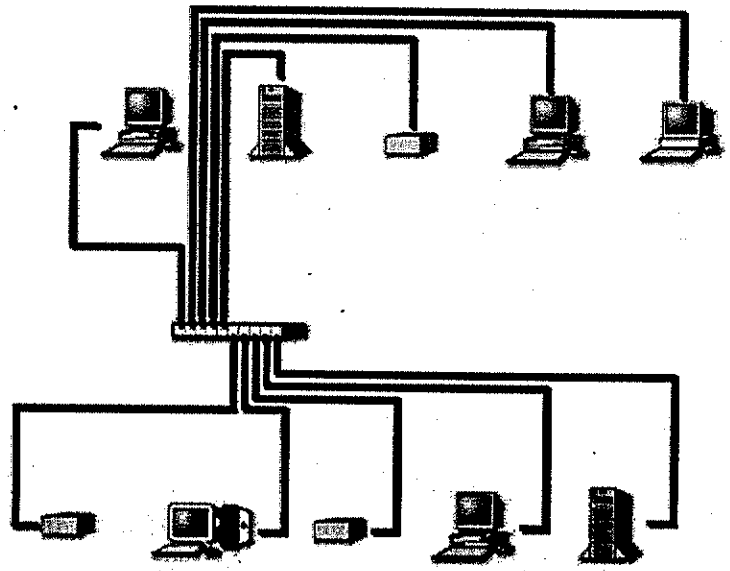
Một cách khác để phân loại mạng là theo topo - mô hình ghép nối mạng, hay còn gọi là hình trạng mạng. Topo mạng gần giống như bản đồ đường phố. Nó mô tả chi tiết cách thức kết nối các thành phần chính của mạng (các nút) và các đường truyền. Có thể so sánh topo mạng với bản thiết kế của một ngôi nhà, trong đó hệ thống điện, sưởi, điều hòa, và nước được tích hợp với nhau trong một thiết kế chung nhất, hoàn chỉnh. Có 3 chiến lược kết nối tổng quát: điểm-điểm (point-to-point), broadcast (điểm-nhiều điểm) và multidrop (đa chặng).

Mô hình điểm-điểm (point-to-point)

Mạng point-to-point gồm nhiều nút, mỗi nút chỉ có thể liên lạc với nút liền kề qua đường liên kết trực tiếp. Mạng point-to-point có thể bao gồm hàng ngàn nút, mỗi nút nối trực tiếp với một số nút nào đó. Nếu một nút cần liên lạc với nút không liền kề, nó buộc phải liên lạc gián tiếp thông qua chuỗi các nút khác. Đầu tiên, nút nguồn chuyển thông điệp tới nút liền kề với mình. Sau đó thông điệp này sẽ được chuyển tuần tự qua dãy các nút liền kề nhau cho đến khi đến được nút đích. Việc chuyển dữ liệu thông qua nút liền kề đến một nút khác thường được gọi là bridging hoặc routing (định tuyến) – tùy thuộc vào kỹ thuật truyền tin (bridging được thảo luận trong chương 5; routing được thảo luận trong chương 4). Có một số topo mạng dựa trên mô hình point-to-point. Xét hai dạng topo mạng point-to-point phổ biến: star (hình sao) và tree (dạng cây).

Star (Hình sao)

Đặc điểm chính của mạng hình sao là có một hub xử lý trung tâm – hub này là trung tâm truyền tin cho tất cả các nút. Cấu hình mạng hình sao đơn giản được minh họa trong hình 1.6. Để các nút có thể truyền thông cho nhau, tất cả dữ liệu phải được chuyển qua hub. Do đó khi hub ngừng hoạt động thì toàn bộ mạng sụp đổ. Hình 1.6 minh họa một mạng 10BASE-T (một dạng Ethernet), gồm các nút được nối trực tiếp với một Ethernet switch thông qua một cáp xoắn đôi trần (UTP – unshielded twisted-pair cable). (Mạng 10BASE-T, switched Ethernet và cáp UTP được thảo luận chi tiết trong chương 5).

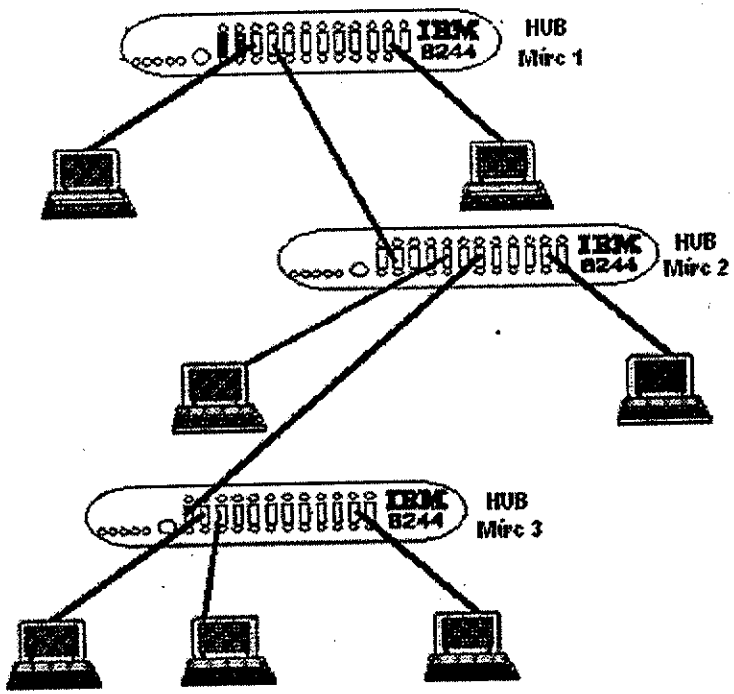


Hình 1. 6 Các thiết bị nối vào một HUB duy nhất. Mô hình sao

Tree (Cây)

Mô hình cây là mô hình phân cấp, gồm một nút gốc hoặc một hub nối đến các nút mức hai hoặc hub mức hai. Các thiết bị ở mức hai lại được nối đến các thiết bị ở mức ba, mức ba được nối đến các thiết bị ở mức bốn... Mạng hình cây đơn giản được minh họa trên hình 1.7. Kiến trúc mạng IEEE 802.12, hay còn gọi là 100VG-AnyLAN, áp dụng mô hình này trong đó các hub được sắp thành tầng tạo thành mô hình phân cấp. Mô hình cây được minh họa trong hình 1.7.

THƯ VIỆN TRƯỜNG ĐHSPKT
SKV 057729



Hình 1. 7 Mô hình cây

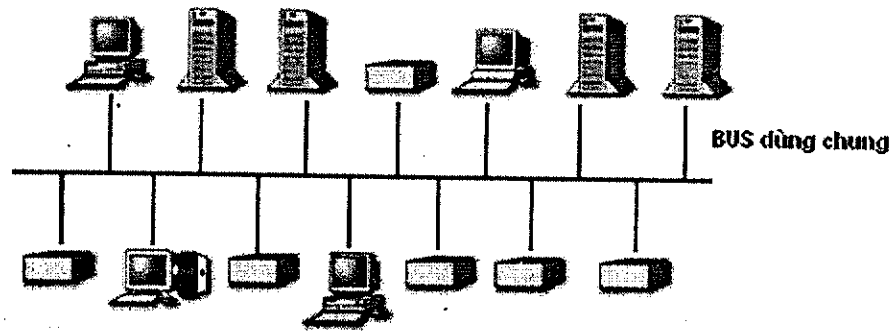
Mô hình điểm - nhiều điểm (Broadcast)

Mô hình này gồm các nút dùng chung một kênh truyền thông. Khác với mô hình điểm - điểm, dữ liệu từ một máy gửi đi sẽ được truyền đến tất cả các nút tham gia kênh truyền dùng chung. Các máy sẽ kiểm tra xem liệu chúng có phải là đích đến của thông điệp nhận được hay không bằng cách kiểm tra địa chỉ đến (destination address) của thông điệp. (Khái niệm địa chỉ sẽ được thảo luận trong phần tiếp theo). Các máy không phải là đích của thông điệp sẽ bỏ qua thông điệp. Chỉ có nút đích của thông điệp mới tiếp nhận thông điệp. Điều này cũng tương tự như một lớp học gồm nhiều sinh viên và một giáo viên. Nếu giáo viên đưa ra một câu hỏi, tất cả sinh viên đều nghe thấy câu hỏi nhưng chỉ sinh viên được giáo viên chỉ định mới trả lời câu hỏi này. Môi trường dùng chung ở đây chính là không khí, câu hỏi của giáo viên là một dạng thông điệp, lan truyền trong không khí và đến tai tất cả các sinh viên (các nút).

Mô hình điểm - nhiều điểm có một số dạng topo phổ biến, đó là bus và ring. Các hệ thống truyền thông vệ tinh cũng dựa trên mô hình điểm - nhiều điểm.

Bus

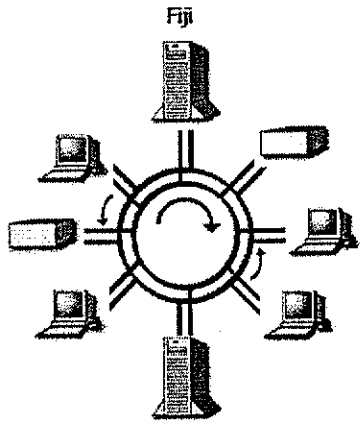
Một cấu hình bus điển hình được minh họa trong hình 1.8. Rõ ràng topo dạng bus thuộc mô hình điểm - nhiều điểm: các nút mạng được nối đến cùng một kênh truyền. Ví dụ điển hình về mạng có topo dạng bus là mạng Ethernet đồng trục (xem chương 5).



Hình 1.8 Mạng Ethernet với Bus dùng chung

Ring

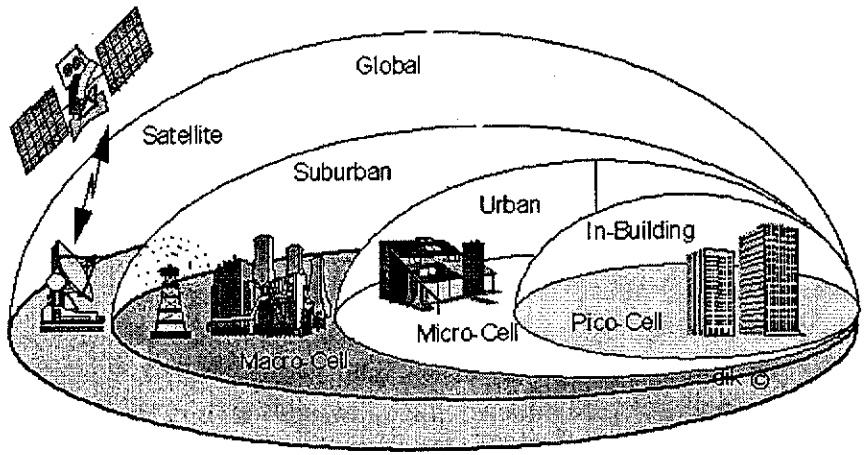
Trong cấu hình ring, tất cả các nút được nối đến một vòng – môi trường truyền thông dùng chung. Trong topo dạng ring truyền thống, thông điệp được truyền lần lượt qua các nút trên vòng. Phụ thuộc vào công nghệ sử dụng, hướng truyền có thể thuận hay nghịch chiều kim đồng hồ. Chú ý rằng mặc dù dữ liệu được chuyển từ nút nọ đến nút kia, ring vẫn không phải là một topo thuộc mô hình điểm - điểm vì các nút dùng chung một kênh truyền. Vì vậy, về mặt logic, trong topo dạng ring tất cả các nút dùng chung một kênh truyền, nhưng về mặt vật lý, việc truyền thông thuộc mô hình điểm - điểm. Trường hợp này cũng giống như topo dạng bus và tất cả các hệ thống điểm - nhiều điểm khác, mạng dạng ring cần một số phương pháp để quản lý việc truy cập vòng đồng thời.



Hình 1.9 Mạng FDDI có topo dạng Ring

Vệ tinh

Trong hệ thống truyền thông vệ tinh, việc truyền dữ liệu từ một ăng-ten trên mặt đất đến vệ tinh thường là mô hình điểm-điểm. Tuy nhiên, tất cả các nút nằm trong mạng đều có thể nhận được dữ liệu từ vệ tinh truyền xuống - vệ tinh phát quảng bá xuống một hoặc nhiều trạm trên mặt đất. Do đó, các hệ thống truyền thông vệ tinh được xếp vào mô hình điểm - nhiều điểm (broadcast). Ví dụ, rất nhiều trường học ở Mỹ có khả năng nhận tin từ vệ tinh. Bất cứ chương trình giáo dục nào được phát quảng bá qua hệ thống vệ tinh đều được các trường học thu được bằng cách điều chỉnh thiết bị nhận đến một tần số thích hợp. Mạng vệ tinh được minh họa trên hình 1.10.



Hình 1. 10 Vệ tinh và các khu vực phủ sóng

Trong mô hình điểm - nhiều điểm có rất nhiều kiểu truyền thông điệp khác nhau:

unicast - chỉ có một thiết bị nhận thông điệp.

multicast - một nhóm thiết bị nhận thông điệp. Chính tầng network của thiết bị nhận sẽ kiểm tra xem thiết bị nhận đó có nằm trong nhóm nhận thông điệp không.

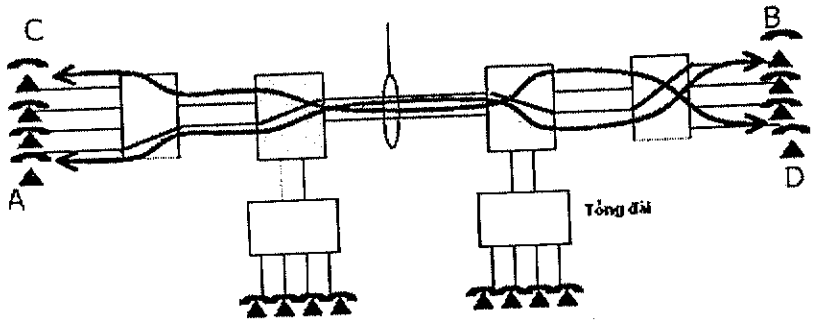
broadcast - tất cả các thiết bị trong mạng nhận thông điệp. Thông điệp broadcast là một thông điệp multicast đặc biệt.

Một đặc điểm khác của mô hình điểm - nhiều điểm là khái niệm tranh chấp (contention). Do tất cả các nút cùng dùng chung một kênh truyền, chúng phải "tranh nhau" kênh truyền khi có nhu cầu gửi giữ liệu. Do vậy mạng dựa trên mô hình broadcast cần giải quyết vấn đề nhiều nút muốn truyền dữ liệu tại cùng một thời điểm. Rất nhiều giao thức được đưa ra để giải quyết tranh chấp giữa các nút và sẽ được trình bày trong chương 5.

Phân loại mạng theo kiểu chuyển

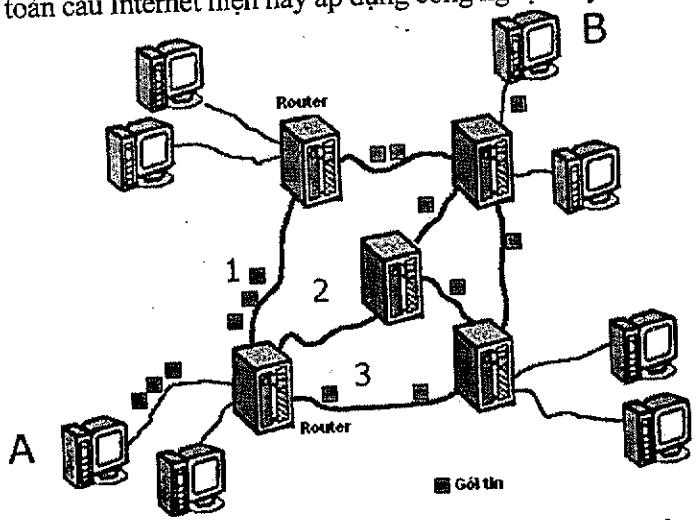
Ngoài việc phân loại theo diện hoạt động và topo, mạng còn được phân loại theo kiểu truyền thông mà chúng sử dụng, cùng với cách dữ liệu được truyền đi. Hai phân loại điển hình là mạng chuyển mạch ảo (virtual circuit-switched) và mạng chuyển gói (packet-switched).

Trong mạng ***chuyển mạch ảo (circuit-switched)***, phải thiết lập mạch vật lý giữa nút nguồn và đích trước khi chuyển dữ liệu thực sự. Mạch này tồn tại trong suốt thời gian chuyển dữ liệu. Mạng điện thoại công cộng là một ví dụ về mạng chuyển mạch ảo. Khi gọi điện thoại, một đường truyền vật lý trực tiếp được thiết lập giữa máy điện thoại của người gọi và máy điện thoại của người nhận. Đường truyền này là một kết nối điểm - điểm, liên kết các bộ chuyển mạch (switch) trong các tổng đài của công ty điện thoại lại với nhau. Sau khi đã được thiết lập, đường truyền chỉ dành riêng để truyền dữ liệu cho cuộc gọi hiện thời. Sau khi truyền xong dữ liệu (cuộc gọi kết thúc), mạch được giải phóng và có thể được cấp phát cho cuộc gọi khác. Như vậy, chuyển mạch làm tăng khả năng chia sẻ đường truyền (link) vì cùng một mạch có thể được sử dụng cho nhiều quá trình truyền khác nhau, mặc dầu không cùng một thời điểm.



Hình 1.11 Mạng điện thoại - chuyển mạch ảo

Trong ***mạng chuyển gói (packet-switched network)***, thông điệp đầu tiên được chia thành những đơn vị nhỏ hơn gọi là gói (packet), sau những packet lần lượt được gửi tới nút nhận qua mạng lưới các thiết bị chuyển mạch trung gian (switch). Packet là một đơn vị dữ liệu nhỏ nhất có thể truyền trên mạng. Mỗi packet mang thông tin về địa chỉ nút nhận cùng số thứ tự của mình. Khi packet đến thiết bị trung gian, thiết bị này căn cứ vào địa chỉ đích của packet để quyết định xem sẽ chuyển packet đi theo hướng nào để đến được thiết bị kế tiếp. Do cấu hình của toàn bộ hệ thống có thể thay đổi nên các packet của cùng một thông điệp có thể đến đích theo những tuyến đường khác nhau. Điều này cũng giống như việc gửi thư. Khi nhận được thư, bưu cục sẽ căn cứ vào địa chỉ người nhận để chuyển đến nơi thích hợp. Mạng toàn cầu Internet hiện nay áp dụng công nghệ chuyển mạch gói này.



Hình 1.12 Mạng chuyển mạch gói – các gói tin đi theo nhiều tuyến đường khác nhau từ A đến B

1.1.4 Địa chỉ mạng, định tuyến, tính tin cậy, tính liên tác và an ninh mạng

Khái niệm mạng máy tính liên quan đến nhiều yếu tố, trong đó có địa chỉ, định tuyến, tính tin cậy, tính liên tác, và an ninh mạng. Phần dưới đây trình bày ngắn gọn về những yếu tố này.

Địa chỉ (Address)

Khái niệm địa chỉ liên quan đến việc gán cho mỗi nút mạng một địa chỉ duy nhất - cho phép các thiết bị khác định vị được nó. Điều này giống như địa chỉ của một ngôi nhà - tên phố sẽ chỉ cho biết khu vực cần đi đến, số nhà xác định chính xác nhà cần đến. Một ví dụ khác là hệ thống điện thoại. Mỗi điện thoại có mã vùng và một số (địa chỉ). Mã vùng cung cấp thông tin về vị trí vùng của điện thoại, còn số điện thoại là số xác định duy nhất máy điện thoại trong vùng. Hệ thống các thiết bị chuyển mạch trong công ty điện thoại được lập trình để tạo nên một kênh truyền giữa hai thiết bị. Về thực chất mã vùng lại được phân cấp thành mã quốc gia và mã khu vực.

Routing - Định tuyến

Định tuyến xác định tuyến đường mà dữ liệu sẽ đi qua trong quá trình chuyển từ nút nhận đến nút gửi. Chức năng định tuyến được thực hiện bởi một thiết bị phần cứng đặc biệt: router (thiết bị định tuyến). Việc lựa chọn tuyến đường tốt nhất phải dựa trên một tiêu chuẩn cụ thể - được gọi là độ đo (metric). Các độ đo định tuyến phổ biến là: khoảng cách, số chặng (hop) và băng thông.

Tính tin cậy

Tính tin cậy chỉ tính toàn vẹn dữ liệu - đảm bảo dữ liệu nhận được giống hệt dữ liệu gửi đi. Mạng máy tính không phải là hệ thống không có lỗi. Trong thực tế, lỗi có thể xuất hiện trên tất cả các môi trường truyền. Vì vậy cần phải thiết kế sao cho hệ thống có khả năng xử lý lỗi. Một trong những phương pháp điển hình là thêm thông tin dư thừa vào dữ liệu chuyển đi sao cho phía nhận phát hiện được lỗi (nếu có). Khi phát hiện ra lỗi, phía

nhận có thể: (1) yêu cầu truyền lại dữ liệu bị lỗi, hoặc (2) kiểm tra xem dữ liệu đúng là gì và sửa lại dữ liệu bị lỗi. Cách thứ nhất là sửa lỗi bằng cách yêu cầu truyền lại, cách thứ hai là tự sửa lỗi. Để sửa được lỗi, phải dò tìm lỗi. Việc tự sửa lỗi nói chung khó thực hiện. Hầu hết các mạng ngày nay đều được thiết kế có khả năng phát hiện lỗi (error detection). Có hai cách để phát hiện lỗi thông dụng là kiểm tra bit chẵn/lẻ và mã dư thừa vòng (CRC – Cyclic Redundancy Check). Hai kỹ thuật này được trình bày trong chương 5.

Tính liên tác (interoperability)

Tính liên tác (interoperability) chỉ khả năng các sản phẩm (phần cứng và phần mềm) của các hãng sản xuất khác nhau có thể giao tiếp được với nhau trong mạng. Trong thời kì hoàng kim của các mạng độc quyền (của tư nhân, hãng sản xuất, hoặc một tổ chức), không cần phải quan tâm đến tính liên tác, miễn là các thành phần cấu thành mạng đều là sản phẩm và giao thức của cùng một hãng sản xuất. Khi hãng sản xuất thứ ba phát triển ứng dụng có tính năng được cải tiến hơn ứng dụng của hãng sản xuất độc quyền, hãng sản xuất thứ ba phải được sự đồng ý của nhà sản xuất độc quyền - tức là hãng sản xuất thứ ba phải trả phí bản quyền. Ngày nay, với bộ giao thức "mở" TCP/IP, các hãng sản xuất - những người viết và bán các ứng dụng dựa trên TCP/IP được tự do làm những điều họ muốn, không phải lo ngại về việc vi phạm bản quyền. Hầu hết các hãng sản xuất máy tính đều cố gắng để sản phẩm của mình tương thích với sản phẩm của hãng sản xuất khác.

An ninh

An ninh mạng chỉ việc bảo vệ mọi thứ trong mạng, bao gồm dữ liệu, phương tiện truyền thông và các thiết bị. An ninh mạng còn bao gồm các chức năng quản trị, các công cụ kỹ thuật và thiết bị như các sản phẩm mã hóa, các sản phẩm kiểm soát truy cập mạng (ví dụ: tường lửa firewall - thiết bị phần cứng đặc biệt bảo vệ mạng với thế giới bên ngoài). An ninh mạng cũng bao gồm việc định ra những chính sách sử dụng tài nguyên mạng, kiểm tra xem tài nguyên mạng có được sử dụng phù hợp với chính sách đã định trước hay không, quy định và kiểm tra chỉ những người có đủ quyền mới được sử dụng các tài nguyên đó…

1.1.5 Chuẩn mạng

Chuẩn mạng định nghĩa các giao tiếp phần cứng, giao thức truyền thông, kiến trúc mạng... Chuẩn mạng thiết lập những quy tắc hay các quy ước cụ thể mà các bên tham gia truyền thông cần tuân thủ. Chúng làm tăng khả năng giao tiếp giữa sản phẩm phần cứng và phần mềm của các hãng sản xuất khác nhau. Chuẩn được xây dựng thông qua các tổ chức chuẩn hóa. Những tổ chức này được chia thành bốn loại chính: (a) quốc gia, (b) vùng, (c) quốc tế và (d) ngành/hiệp hội thương mại/hiệp hội nghề. Thành viên của tổ chức chuẩn thường là đại diện của chính phủ, viện nghiên cứu và hãng sản xuất. Quá trình xây dựng chuẩn phải đảm bảo được tính thống nhất, vì vậy thường kéo dài, đôi khi phải mất nhiều năm mới cho ra đời được một chuẩn chính thức. Quá trình này cũng bị ảnh hưởng bởi các yếu tố khác như kinh tế, chính trị.

Chuẩn chính thức (De jure standard)

Chuẩn chính thức được công nhận bởi những tổ chức chuẩn hóa chuyên nghiệp. Ví dụ, những giao thức về modem được xây dựng bởi Hiệp hội truyền thông quốc tế (International Telecommunications Union – ITU), hay chuẩn EIA/TIA-568 dùng cho Commercial Building Telecommunications Wiring được xây dựng bởi Electronic Idustries Association - EIA và Telecommunications Inustries Association – TIA, hoặc các chuẩn cho mạng cục bộ được xây dựng bởi Institute for Electrical and Electronic Engineers – IEEE. (Các chuẩn này sẽ được trình bày chi tiết trong các chương sau).

Chuẩn thực tế (De facto standard)

Chuẩn thực tế là chuẩn tồn tại trong thực tế chứ không phải do các tổ chức chuẩn hóa xây dựng nên. Chúng được phát triển thông qua sự chấp nhận của toàn ngành đối với chuẩn nào đó của một hãng nhà sản xuất cụ thể. Ví dụ một chuẩn thực tế là Network File System (NFS) - giao thức chia sẻ file của hãng Sun Microsystems. Sun đã công khai đặc tả của giao thức này, do đó những nhà sản xuất khác có thể tự do triển khai. Kết quả NFS được sử dụng rộng rãi và được coi như một chuẩn thực tế. Hiện tại, NFS được cài đặt trên rất nhiều hệ thống UNIX khác nhau (Sun, IBM, Silicon Gracphics, Compaq, và HP), cũng như các hệ thống dựa trên Macintosh và Intel. Một chuẩn thực tế khác là Java - ngôn ngữ lập trình Web được phát triển bởi hãng Sun Microsystems.

Chuẩn riêng của hãng

Chuẩn của hãng quy định những yêu cầu cụ thể của một nhà sản xuất nào đó. Những đặc tả này không được công khai, chỉ được tuân theo và chấp nhận bởi chính hãng sản xuất đề nghị ra nó. Trong thời kì đầu của mạng, các chuẩn của hãng thống trị. Mặc dầu ngày nay những chuẩn như vậy không còn được tán thành nữa song chúng vẫn tồn tại rất nhiều. Được biết đến nhiều nhất phải kể đến các chuẩn của IBM (ví dụ: SNA - kiến trúc hệ thống mạng của IBM, giao thức IPX của Novell - dựa trên giao thức XNS của Xerox). Chuẩn riêng của hãng trói buộc khách hàng vào giải pháp của một nhà sản xuất cụ thể, làm cho họ gặp khó khăn khi sử dụng sản phẩm (phần cứng hoặc phần mềm) của các hãng sản xuất khác.

Chuẩn hiệp hội

Chuẩn hiệp hội tương tự như chuẩn chính thức theo nghĩa chúng là sản phẩm của quá trình chuẩn hóa. Điểm khác nhau là quá trình lập kế hoạch và chuẩn hóa những chuẩn này không chịu sự quản lý của các tổ chức chuẩn hóa chuyên nghiệp. Thay vào đó, đặc tả cho các chuẩn được thiết kế và thỏa thuận bởi nhóm các nhà sản xuất thành lập nên hiệp hội, với một mục đích cụ thể: đạt được mục tiêu chung. Những nhà sản xuất này cam kết hỗ trợ cho các chuẩn được phát triển bởi hiệp hội, và phát triển những sản phẩm tuân theo chuẩn này. Ví dụ về các chuẩn hiệp hội như Fast Ethernet, Asynchronous Transfer Mode (ATM Forum) hay Gigabit Ethernet.

1.2 MÔ HÌNH OSI

Tổ chức ISO (International Standards Organization) được thành lập năm 1971 với mục đích xây dựng các tiêu chuẩn quốc tế. Một trong các chuẩn ISO định nghĩa các mặt của truyền thông mạng là mô hình OSI - Open Systems Interconnection (Mô hình liên kết giữa các hệ thống mở). Đây là mô hình cho phép bất cứ hai hệ thống nào (cho dù khác nhau) có thể truyền thông với nhau mà không cần quan tâm đến kiến trúc bên dưới của chúng. Các giao thức của riêng một hãng sản xuất thường ngăn ngừa việc truyền thông giữa hai hệ thống không cùng một kiểu. Mô hình OSI ra đời

với mục đích cho phép hai hệ thống bất kì truyền thông với nhau mà không cần thay đổi bất cứ phần cứng hoặc phần mềm nào bên dưới. Mô hình OSI không phải là một giao thức; nó là một mô hình để nhận biết và thiết kế một kiến trúc mạng linh động, vững chắc và có khả năng liên tác. Chú ý ISO là tên tổ chức; OSI là một mô hình.

1.2.1 Mô hình

Mô hình OSI được phân tầng với mục đích thiết kế các hệ thống mạng cho phép tất cả các kiểu hệ thống máy tính khác nhau có thể truyền thông với nhau. Mô hình gồm 7 tầng riêng biệt nhưng có liên quan đến nhau, mỗi tầng định nghĩa một phần của quá trình truyền thông tin trên mạng. Những quy tắc cơ bản của mô hình OSI là nền tảng cơ bản để nghiên cứu chi tiết truyền thông dữ liệu.

Thực ra trong cuộc sống, chúng ta gặp khá nhiều ví dụ về việc phân tầng. Giả sử người A viết thư gửi cho người B. Sau khi viết thư xong, A cho thư vào phong bì, dán kín, ghi địa chỉ của B, dán tem và nhét bức thư vào hộp thư ở bưu điện. Giữa A và B, đơn vị dữ liệu trao đổi là các lá thư. Bức thư có thể xem là dữ liệu thực sự trong khi phong bì thư có thể xem là một loại tiêu đề chứa các thông tin điều khiển. Hệ thống bưu điện (bao gồm nhiều bưu cục – là các trạm trung gian mà bức thư sẽ đi qua) chịu trách nhiệm chuyển bức thư tới địa chỉ của B. Với ví dụ này tầng dưới (hệ thống bưu điện) sẽ cung cấp dịch vụ chuyển thư cho tầng trên (A và B). A và B chỉ quan tâm đến nội dung bức thư, khuôn dạng thư, ngôn ngữ viết trong thư... mà không cần quan tâm đến làm thế nào để thư có thể chuyển tới B. Đây chính là ưu điểm của việc phân tầng: tầng trên sử dụng dịch vụ của tầng dưới nhưng không cần quan tâm đến cách thức thực hiện dịch vụ đó.

Kiến trúc phân tầng

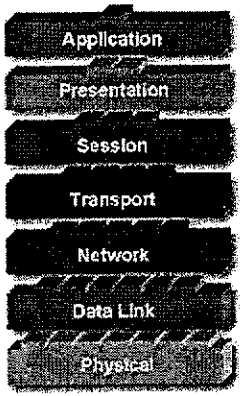
Mô hình OSI gồm 7 tầng (Hình 1.13):

Tầng vật lý (Physical layer)

Tầng liên kết dữ liệu (Datalink layer)

Tầng mạng (Network layer)

Tầng giao vận (Transport layer)
Tầng phiên (Session layer)
Tầng trình diễn (Presentation layer)
Tầng ứng dụng (Application layer).



Hình 1.13 Bảy tầng trong mô hình OSI

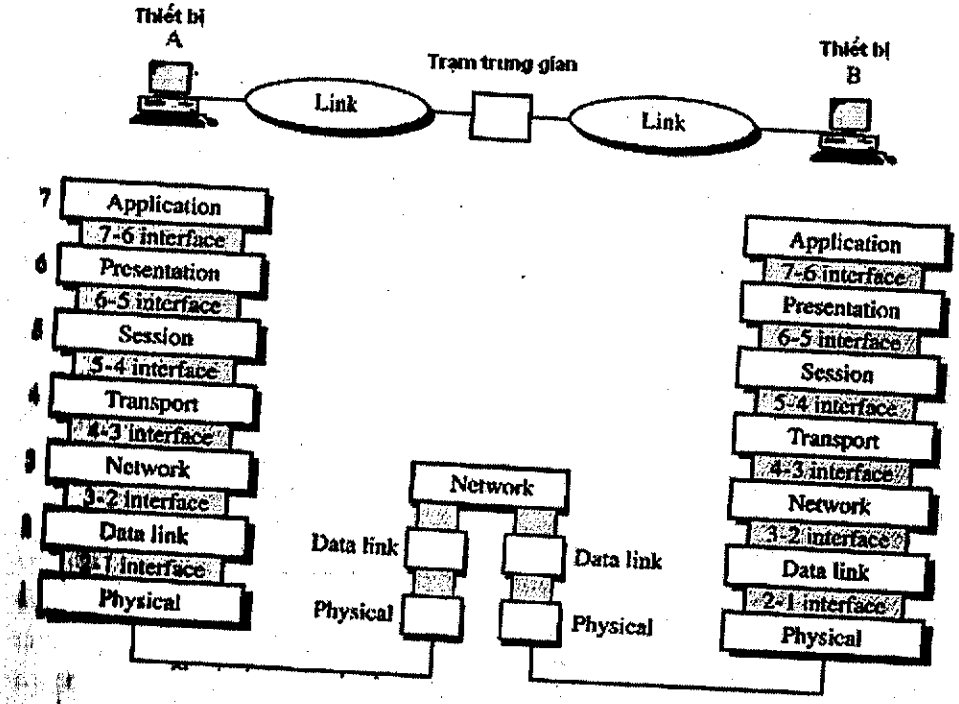
Trong lịch sử LỊCH SỬ MÔ HÌNH 7 TẦNG

Phần lớn công việc thiết kế mô hình OSI được thực hiện bởi hai nhà nghiên cứu Mike Canepa và Charlie Bachman tại công ty Honeywell Information Systems. Mục tiêu của nhóm nghiên cứu này là phát triển một hệ thống nguyên mẫu cùng một bản kế hoạch phát triển sản phẩm. Nửa đầu thập niên 70, nhóm tập trung vào thiết kế cơ sở dữ liệu có hỗ trợ truy cập từ xa – và do đó cần tới một kiến trúc truyền thông phân tán nhưng có cấu trúc. Sau khi có những nghiên cứu về kiến trúc mạng SNA của IBM, các giao thức đề xuất cho mạng ARPANET, nhóm đã phát triển kiến trúc hệ thống phân tán (DSA – Distributed Systems Architecture) - tiền thân của mô hình 7 tầng ngày nay.

Trong khi đó, Cơ quan chuẩn hóa Anh quốc đề nghị Tổ chức chuẩn quốc tế (ISO) ban hành một kiến trúc chung cho truyền thông phân tán. Theo yêu cầu này, ISO thành lập tiểu ban Kết nối các Hệ thống mở (Open Systems Interconnection – OSI) và yêu cầu Cơ quan chuẩn hóa Hoa Kỳ (ANSI) đưa ra một đề xuất cho phiên họp đầu tiên của OSI.

Bachman và Canepa tham dự những phiên họp đầu tiên với ANSI và đã trình bày mô hình bảy tầng của mình và ngay lập tức mô hình này được lựa chọn để đề xuất cho tiểu ban OSI. Vào tháng 3/1978, khi tiểu ban OSI họp tại Washington, Bachman đã trình bày mô hình của mình. Với một sự nhất trí cao, tiểu ban đã chấp nhận kiến trúc phân tầng này đáp ứng đầy đủ phần lớn các yêu cầu và có khả năng mở rộng để thỏa mãn các yêu cầu mới. Phiên bản đầu tiên của mô hình này được công bố vào tháng 3/1978. Phiên bản kế tiếp (với một vài chỉnh sửa) công bố vào tháng 6/1979 và tiếp tục được chỉnh sửa thêm.

Hình 1.14 minh họa mối quan hệ giữa các tầng khi thông điệp được gửi từ thiết bị A đến thiết bị B. Khi đi từ A đến B, thông điệp có thể đi qua nhiều nút trung gian khác. Những nút trung gian này thường chỉ liên quan đến 3 tầng đầu của mô hình OSI. Khi phát triển mô hình, các nhà thiết kế đã phân tích quá trình truyền dữ liệu ra những chức năng cơ bản nhất, và nhóm những chức năng có mục đích (sử dụng) liên quan đến nhau vào các nhóm riêng - còn gọi là tầng (layer). Mỗi tầng đều có chức năng, nhiệm vụ xác định. Bằng cách xác định và khoanh vùng các chức năng trong mô hình, nhà thiết kế đã đưa ra một kiến trúc đạt được cả tính toàn diện và linh hoạt. Quan trọng nhất, mô hình OSI tạo ra tính trong suốt hoàn toàn giữa hai hệ thống không tương thích với nhau.



Hình 1.13 Mô hình OSI

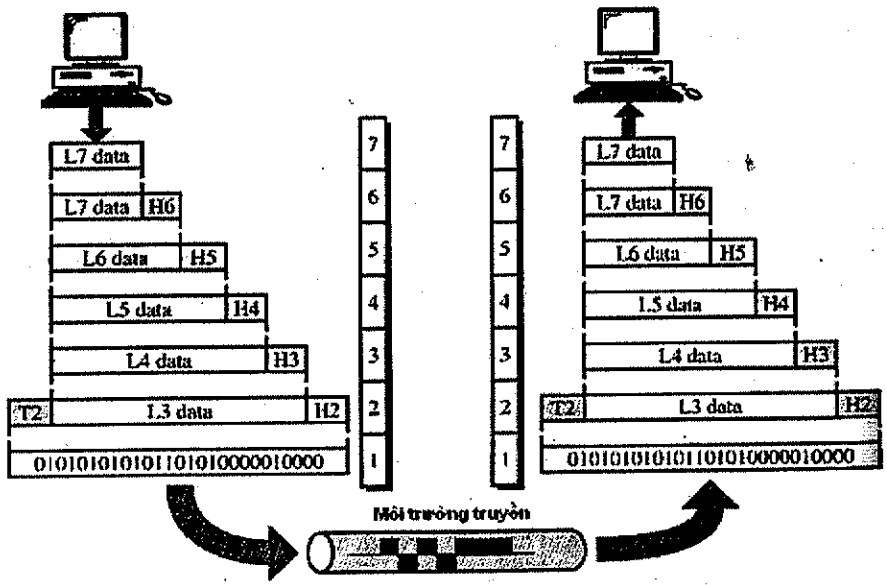
Giao tiếp trình ngang hàng (peer-to-peer)

Tại thiết bị đầu cuối, mỗi tầng sử dụng các dịch vụ do tầng bên dưới cung cấp. Ví dụ, tầng 3 sử dụng các dịch vụ do tầng 2 cung cấp và đến lượt

mình lại cung cấp dịch vụ cho tầng 4. Giữa các máy tính, tầng N trên một thiết bị giao tiếp với tầng N trên thiết bị khác. Việc giao tiếp này được tiến hành theo các quy tắc và quy ước đã được thỏa thuận trước-gọi là giao thức.

Tại tầng vật lý, truyền thông diễn ra trực tiếp: A gửi một luồng bit đến B dưới dạng tín hiệu. Tuy nhiên tại các tầng cao hơn trên máy A, dữ liệu được chuyển dần xuống các tầng bên dưới, đến máy B và tiếp tục đi ngược lên các tầng cao hơn (của B). Mỗi tầng trong máy gửi (A) thêm các thông tin của mình vào thông điệp nhận được từ phía trên rồi sau đó chuyển toàn bộ gói dữ liệu xuống tầng phía dưới. Các thông tin được thêm vào này - được gọi là header (tiêu đề chèn trước) và trailer (tiêu đề chèn sau) - là các thông tin điều khiển được thêm vào đầu hay cuối gói dữ liệu. Header được thêm vào thông điệp tại mỗi tầng 6, 5, 4, 3, và 2; trailer được thêm vào tại tầng 2.

Tại tầng 1, gói dữ liệu được chuyển thành dạng tín hiệu có thể truyền đi tới máy nhận. Tại thiết bị nhận, các tiêu đề được lấy ra dần dần trong quá trình chuyển dữ liệu lên trên. Ví dụ, tầng 2 loại bỏ các tiêu đề của tầng 2 và chuyển phần còn lại (dữ liệu) cho tầng 3. Tầng 3 loại bỏ các tiêu đề tầng 3 và chuyển phần dữ liệu cho tầng 4...



Hình 1. 14 Dữ liệu được chuyển dọc theo các tầng đi xuống phía dưới

Giao diện giữa các tầng

Trên cùng một máy tính, hai tầng kề nhau trao đổi dữ liệu với nhau qua các giao diện (interface). Giao diện định nghĩa cách thức và khuôn dạng dữ liệu trao đổi giữa hai tầng kề nhau trên cùng một thiết bị. Định nghĩa giao diện giữa các tầng một cách rõ ràng cho phép thay đổi cách thức triển khai tại một tầng mà không ảnh hưởng đến các tầng khác.

Trong thuật ngữ mạng, người ta thường gọi giao diện giữa các tầng là điểm truy cập dịch vụ (Service Access Point – SAP) vì tầng trên yêu cầu dịch vụ của tầng dưới thông qua giao diện.

Tổ chức các tầng

Có thể chia bảy tầng vào ba nhóm. Nhóm tầng hỗ trợ mạng gồm ba tầng: vật lý, liên kết dữ liệu và mạng chịu trách nhiệm về các mặt liên quan đến khía cạnh vật lý khi truyền dữ liệu từ thiết bị này sang thiết bị khác (ví dụ: đặc tả điện, các kết nối vật lý, định địa chỉ vật lý, định thời gian truyền và tính tin cậy). Tầng phiên, trình diễn, và ứng dụng thuộc nhóm tầng hỗ trợ người dùng; chúng tạo ra khả năng liên tác giữa các hệ thống phần mềm khác nhau. Tầng 4 - tầng giao vận đảm bảo việc chuyển dữ liệu đầu cuối (end-to-end) tin cậy, trong khi tầng 2 đảm bảo việc truyền dữ liệu tin cậy trên một đường truyền vật lý riêng lẻ. Nói chung, các tầng trên của mô hình OSI thường được triển khai qua phần mềm trong khi nhóm các tầng dưới được triển khai bằng sự kết hợp của cả phần cứng lẫn phần mềm. Tầng vật lý hầu như được triển khai bởi phần cứng.

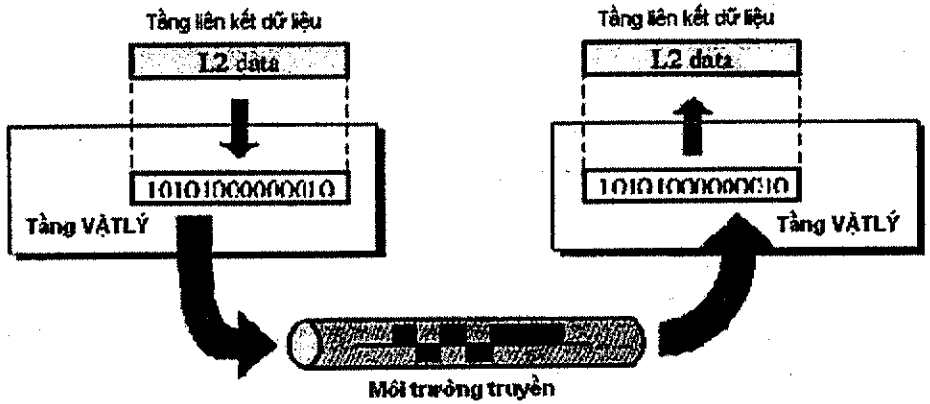
Hình 1.15 mô tả tổng quan các tầng trong mô hình OSI. Trong hình vẽ, (dữ liệu) L7 là đơn vị dữ liệu tại tầng 7, (dữ liệu) L6 là đơn vị dữ liệu tại tầng 6,... Trong thuật ngữ mạng người ta gọi Li là đơn vị dữ liệu giao thức tầng i (iPDU – Protocol Data Unit). Đối với tầng thứ i+1, dữ liệu tầng i truyền cho được gọi là đơn vị dữ liệu dịch vụ (service data unit – SAP). Nói chung PDU chứa SAP và một số thông tin tiêu đề khác. Quá trình được bắt đầu tại tầng 7 (tầng ứng dụng), sau đó chuyển xuống các tầng dưới. Tại mỗi tầng ngoại trừ tầng 7 và tầng 1, header được chèn thêm vào đơn vị dữ liệu. Tại tầng 2, ngoài header, trailer cũng được thêm vào đơn vị dữ liệu. Khi đi qua tầng vật lý (tầng 1), đơn vị dữ liệu (đã định khuôn dạng) được chuyển thành tín hiệu vật lý và truyền đi qua môi trường vật lý.

Tại nơi nhận, tín hiệu vật lý đến tầng 1 và được chuyển ngược lại thành chuỗi bit. Các đơn vị dữ liệu sau đó sẽ được chuyển dần từ tầng 1 lên các tầng trên trong mô hình OSI. Tại mỗi tầng, các header và trailer được thêm vào khối dữ liệu ở tầng tương ứng bên máy gửi được lấy ra. Khi đến tầng 7, thông điệp đã ở dạng dữ liệu phù hợp và ứng dụng có thể sử dụng.

1.2.2 Chức năng các tầng

Trong phần này, chúng ta sẽ mô tả chi tiết chức năng của từng tầng trong mô hình OSI.

Tầng vật lý



Hình 1. 15 Vị trí, vai trò của tầng vật lý

Tầng vật lý thực hiện các chức năng cần thiết để truyền luồng bit dữ liệu đi qua môi trường vật lý. Nó giải quyết những vấn đề liên quan đến đặc điểm kỹ thuật về cơ và điện giữa card ghép nối (interface) với môi trường truyền dẫn. Nó cũng xác định các thủ tục, chức năng mà thiết bị vật lý và thiết bị giao tiếp cần phải tuân thủ. Hình 1.16 minh họa mối quan hệ giữa tầng vật lý với môi trường truyền dẫn và tầng liên kết dữ liệu.

Trong ví dụ chuyển thư, tầng vật lý liên quan đến công nghệ chuyển thư, chẳng hạn là xe đạp, máy bay, tàu hỏa, tàu thủy... dùng chuyên chở các túi thư. Tầng liên kết dữ liệu chuyển lá thư cho tầng vật lý và hy vọng tầng vật lý chuyển lá thư sang phía bên kia của kênh truyền.

Tầng vật lý liên quan đến:

Đặc điểm vật lý của môi trường (thiết bị) giao tiếp và truyền thông: Tầng vật lý xác định đặc điểm/đặc tính giao diện giữa thiết bị và môi trường truyền dẫn. Nó cũng xác định kiểu môi trường truyền dẫn thông tin (xem chương 5).

Biểu diễn bit: Dữ liệu tầng vật lý là luồng bit liên tục (các chuỗi 0 và 1). Để truyền đi, bit phải được mã hóa thành tín hiệu điện hoặc quang. Tầng vật lý xác định phương thức mã hóa (các bit 0 và 1 được chuyển thành tín hiệu như thế nào).

Tốc độ dữ liệu: Tốc độ truyền dẫn - số bit được gửi đi trong một đơn vị thời gian. Nói cách khác, tầng vật lý xác định khoảng thời gian để truyền đi một bit.

Đồng bộ hóa các bit: Máy gửi và nhận phải được đồng bộ hóa ở mức bit. Nói cách khác, đồng hồ của máy gửi và nhận phải được đồng bộ hóa.

Cấu hình đường truyền: Tầng vật lý liên quan đến việc kết nối các thiết bị vào môi trường truyền thông. Trong cấu hình điểm-điểm (point-to-point), hai thiết bị được nối với nhau qua một đường truyền riêng. Trong cấu hình điểm-nhiều điểm (multipoint), một đường truyền được nhiều thiết bị dùng chung.

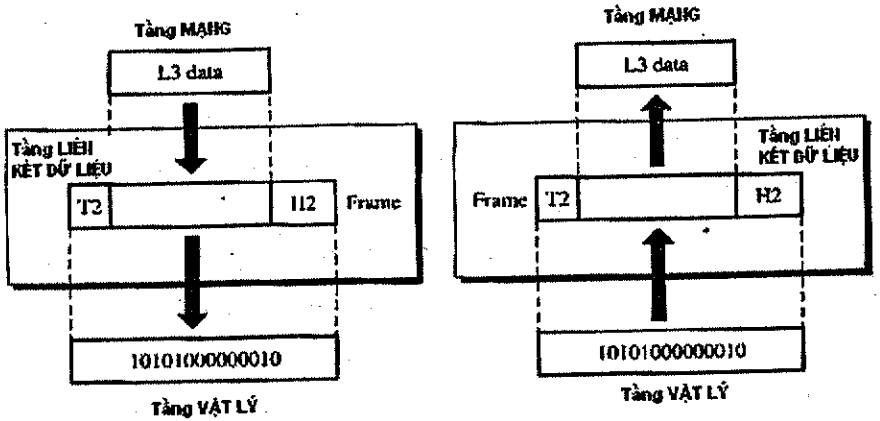
Topo (Mô hình ghép nối) vật lý: Topo vật lý xác định cách nối các thiết bị với nhau để tạo thành mạng. Có thể sử dụng topo dạng lưới (mesh topology) (mỗi thiết bị được nối với tất cả các thiết bị còn lại), topo dạng sao (star topology) (mỗi thiết bị được nối với một thiết bị trung tâm), topo dạng vòng (ring topology) (mỗi thiết bị được nối với một thiết bị bên cạnh, cứ như vậy tạo thành vòng), hay topo dạng bus (mỗi thiết bị được nối đến một đường truyền chung).

Chế độ truyền dẫn: Tầng vật lý xác định hướng truyền dữ liệu giữa hai thiết bị: đơn công (simplex), bán song công (half-duplex), hay song công (full-duplex). Trong chế độ đơn công, một thiết bị chỉ có thể gửi hoặc nhận dữ liệu. Chế độ đơn công là truyền thông một chiều. Trong chế độ bán song công, thiết bị có thể gửi và nhận dữ liệu, nhưng không phải tại cùng một thời

điểm. Trong chế độ song công, thiết bị có thể nhận và gửi dữ liệu tại cùng một thời điểm.

Tầng liên kết dữ liệu

Nhiệm vụ của tầng liên kết dữ liệu là truyền thông giữa hai nút nối trực tiếp với nhau. Nó biến tầng vật lý không tin cậy thành đường truyền tin cậy cho tầng mạng bên trên. Hình 1.17 minh họa mối quan hệ giữa tầng liên kết dữ liệu với tầng mạng và tầng vật lý.



Hình 1.16 Vị trí, vai trò của tầng liên kết dữ liệu

Tầng liên kết dữ liệu chịu trách nhiệm:

Framing – Đóng gói dữ liệu: Tầng liên kết dữ liệu chia luồng bit nhận được từ tầng mạng thành các đơn vị dữ liệu gọi là frame.

Định địa chỉ vật lý: Nếu gói dữ liệu được chuyển đến thiết bị khác trong mạng, tầng liên kết dữ liệu thêm vào tiêu đề của frame địa chỉ vật lý của nơi nhận (địa chỉ đích) và có thể địa chỉ vật lý của nơi gửi (địa chỉ nguồn). Nếu gói dữ liệu được chuyển đến các thiết bị bên ngoài mạng, địa chỉ nhận sẽ là địa chỉ của thiết bị trung gian kết nối mạng ra bên ngoài.

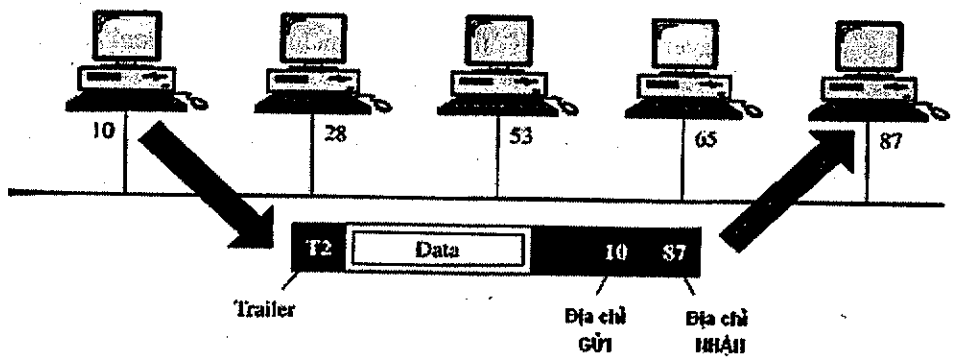
Kiểm soát lưu lượng: Nếu tốc độ nhận dữ liệu nhỏ hơn tốc độ gửi dữ liệu, tầng liên kết dữ liệu phải thực hiện kỹ thuật kiểm soát lưu lượng để ngăn ngừa tình trạng quá tải tại nơi nhận.

Kiểm soát lỗi: Tầng liên kết dữ liệu làm tăng tính tin cậy của tầng vật lý bằng cách sử dụng kỹ thuật phát hiện và truyền lại các frame bị lỗi

hoặc bị mất. Nó cũng sử dụng kỹ thuật xử lý các frame trùng lặp. Kiểm soát lỗi thường được thực hiện bằng cách thêm một trailer vào phần cuối của frame.

Kiểm soát truy cập: Khi nhiều thiết bị dùng chung đường truyền, các giao thức ở tầng liên kết dữ liệu sẽ quyết định thiết bị nào được quyền sử dụng đường truyền tại một thời điểm xác định.

Trong hình 1.18, nút có địa chỉ vật lý 10 gửi frame đến nút có địa chỉ vật lý 87. Hai nút này được nối với nhau qua đường truyền dùng chung. Ở tầng liên kết dữ liệu, header của frame chứa các địa chỉ vật lý. Phần còn lại của header chứa các thông tin cần thiết cho tầng liên kết dữ liệu. Trailer thường chứa các bit dư thừa để kiểm soát lỗi.



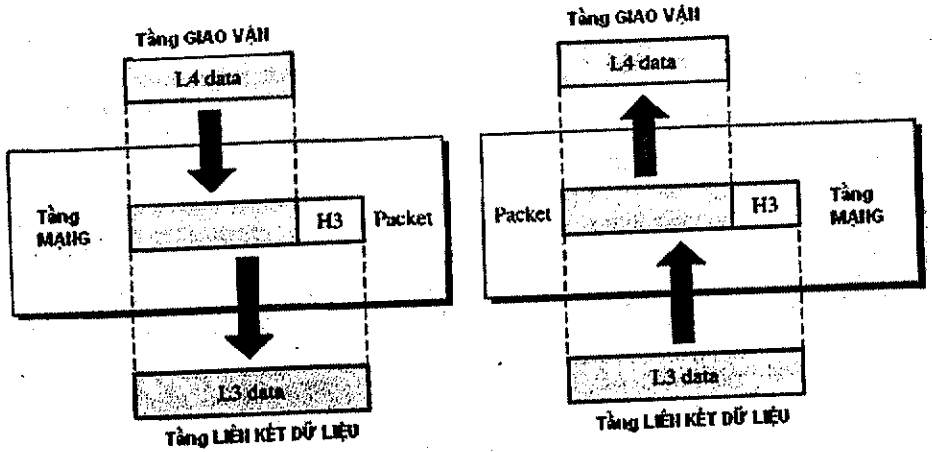
Hình 1. 17 Ví dụ về địa chỉ của tầng liên kết dữ liệu

Hãy xét tiếp ví dụ chuyển thư. Sau khi thu thập thư tại hòm thư, nhân viên bưu chính thực hiện việc phân loại thư ra hai nhóm: nhóm thứ nhất gồm các thư gửi tới địa chỉ nằm trong vùng do bưu cục quản lý, nhóm thứ hai chuyển ra phía ngoài. Đối với nhóm thứ hai, nhân viên bưu cục đặt tất cả thư trong một túi thư lớn và chuyển cho bưu cục cấp cao hơn. Các túi thư có thể xem là các "frame". Hơn thế nữa, địa chỉ trên túi thư sẽ cho phép thư được chuyển đến bưu cục cấp cao thích hợp.

Tầng mạng

Tầng mạng chịu trách nhiệm chuyển gói dữ liệu từ nơi gửi đến nơi nhận, gói dữ liệu có thể phải đi qua nhiều mạng (các chặng trung gian). Tầng liên kết dữ liệu thực hiện truyền gói dữ liệu giữa hai thiết bị trong cùng một mạng, còn tầng mạng đảm bảo gói dữ-liệu sẽ được chuyển từ nơi gửi đến đúng nơi nhận.

Nếu hai thiết bị nằm trên cùng một môi trường truyền thì rõ ràng không cần tầng mạng. Tuy nhiên, nếu hai thiết bị ở trên hai mạng khác nhau, và giữa chúng có nhiều thiết bị kết nối trung gian thì cần phải có tầng mạng để thực hiện việc chuyển dữ liệu từ nguồn đến đích. Hình 1.19 minh họa mối quan hệ giữa tầng mạng với tầng giao vận và liên kết dữ liệu.



Hình 1.18 Vị trí tầng mạng

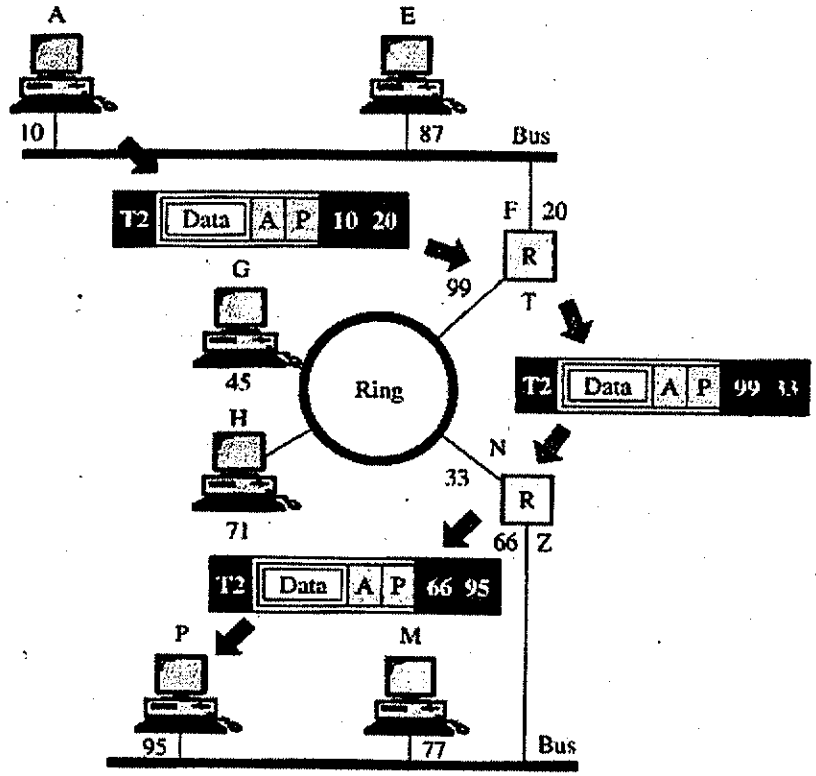
Tầng mạng có nhiệm vụ:

Định địa chỉ logic: Địa chỉ vật lý của tầng liên kết dữ liệu chỉ giải quyết được vấn đề định địa chỉ cục bộ trên một mạng nhỏ. Nếu gói dữ liệu được chuyển sang mạng khác, cần có hệ thống địa chỉ khác để phân biệt giữa hệ thống gửi và hệ thống nhận. Tầng mạng bổ sung thêm tiêu đề - có chứa địa chỉ logic của thiết bị nhận và thiết bị gửi vào mỗi gói dữ liệu gửi đi.

Định tuyến: Khi mạng hoặc các nút riêng rẽ nối với nhau tạo thành một liên mạng (mạng của các mạng), các thiết bị kết nối trung gian (router hoặc gateway) phải xác định tuyến đường (định tuyến) cho gói dữ liệu để chúng đến được đích.

Giả sử trong Hình 1.18, dữ liệu được gửi từ nút có địa chỉ mạng A với địa chỉ vật lý 10 trong một mạng cục bộ tới nút có địa chỉ mạng P với địa chỉ vật lý 95 trong mạng cục bộ khác. Do hai thiết bị thuộc hai mạng khác nhau, chúng ta không thể chỉ sử dụng địa chỉ vật lý vì địa chỉ vật lý chỉ có tác dụng trong mạng cục bộ. Cái chúng ta cần ở đây là một địa chỉ toàn

thể để có thể chuyển packet giữa các mạng khác nhau. Địa chỉ logic có đặc điểm này. Gói dữ liệu tại tầng mạng chứa địa chỉ logic - địa chỉ này không thay đổi khi packet đi từ nơi gửi đến nơi nhận (A và P). Địa chỉ logic không thay đổi khi gói dữ liệu đi từ mạng này sang mạng khác; ngược lại địa chỉ vật lý thay đổi khi packet đi từ mạng này sang mạng khác. Trong hình vẽ, R là router – kiểu thiết bị này sẽ được mô tả kỹ trong chương 3.

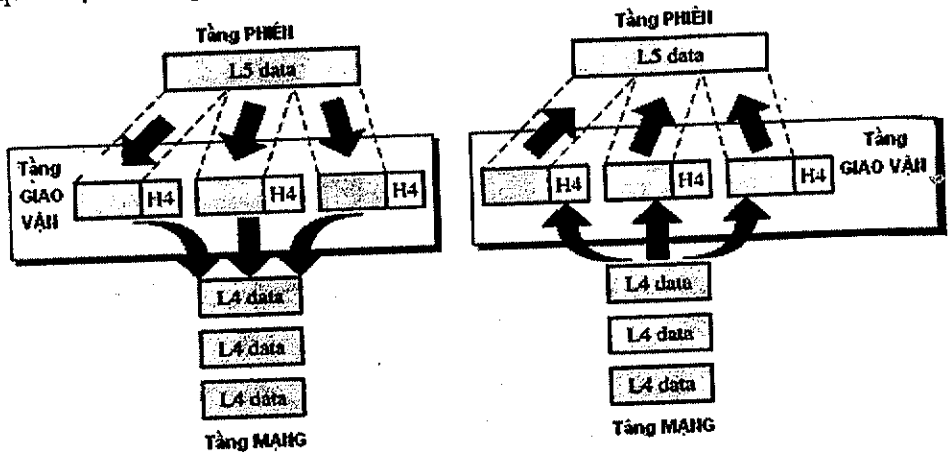


Hình 1.19 Ví dụ về địa chỉ tầng mạng

Các bưu cục sẽ cung cấp dịch vụ giống với tầng mạng. Trong mỗi bưu cục sẽ có một "bảng định tuyến" cho phép bưu tá xác định nơi cần chuyển tiếp bức thư đến. Bức thư sẽ được chuyển đến trạm kế tiếp (bưu cục hay địa chỉ nhận) nhờ vào dịch vụ của tầng liên kết dữ liệu. Rõ ràng rằng thư được chuyển trên các chặng có thể bằng những phương thức hoàn toàn khác nhau (bằng ô tô, máy bay...). Điểm khác biệt duy nhất giữa mạng bưu chính và mạng máy tính là topo của mạng bưu chính gần như không thay đổi theo thời gian, do vậy việc định tuyến gần như là "tĩnh".

Tầng giao vận

Tầng giao vận chịu trách nhiệm chuyển toàn bộ thông điệp từ nơi gửi đến nơi nhận. Tầng mạng chuyển từng gói dữ liệu riêng lẻ từ nơi gửi đến nơi nhận mà không quan tâm đến quan hệ giữa các gói dữ liệu. Tầng mạng xử lý mỗi gói dữ liệu một cách độc lập mà không quan tâm các gói có thuộc vào cùng một thông điệp hay không. Nói cách khác, tầng giao vận đảm bảo gửi thông điệp đến nơi nhận một cách toàn vẹn. Hình 1.21 minh họa mối quan hệ của tầng giao vận với tầng mạng và tầng phiên.



Hình 1. 20 Quan hệ giữa tầng giao vận, tầng phiên và tầng mạng

Tầng giao vận tạo ra một kết nối logic giữa hai cổng đầu cuối: tất cả các gói dữ liệu của cùng một thông điệp được truyền theo đường kết nối đó. Có ba giai đoạn của kết nối: thiết lập kết nối, truyền dữ liệu, giải phóng kết nối. Do phải truyền tất cả các gói dữ liệu trên một kết nối, tầng giao vận còn phải kiểm soát thứ tự truyền, lưu lượng, phát hiện và sửa lỗi.

Tầng giao vận chịu trách nhiệm:

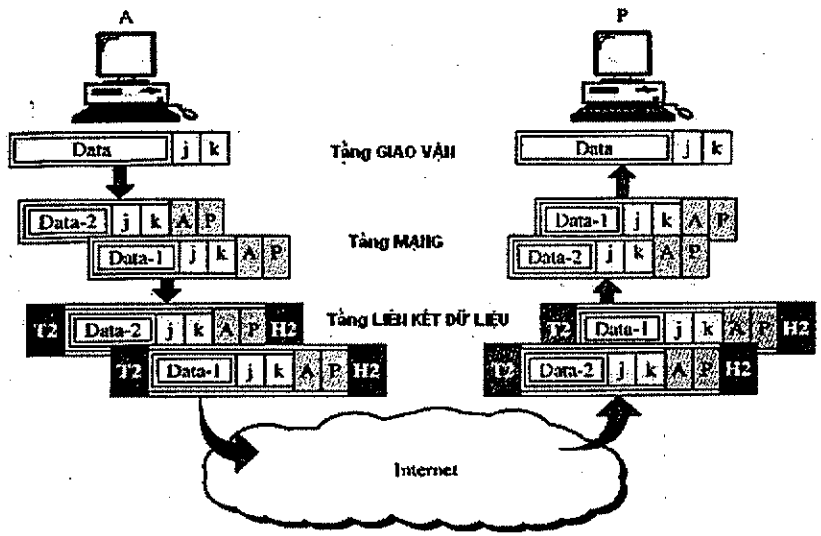
Địa chỉ cổng (port number): Các máy tính thường chạy nhiều chương trình tại cùng một thời điểm. Vì vậy việc chuyển thông điệp không chỉ là truyền dữ liệu từ một máy tính này đến một máy tính khác mà phải chuyển thông điệp từ tiến trình cụ thể trên máy tính này đến tiến trình cụ thể trên máy tính khác. Header được thêm vào tại tầng giao vận phải chứa thông tin về một kiểu địa chỉ - địa chỉ cổng hay địa chỉ tiến trình. Sau khi tầng mạng chuyển gói dữ liệu tới thiết bị nhận, tầng giao vận ở phía nhận phải chuyển toàn bộ thông điệp đến đúng tiến trình nhận.

Phân mảnh và tái hợp nhất: Mỗi thông điệp được chia thành các đoạn (segment) nhỏ, được truyền độc lập với nhau. Mỗi segment có một số thứ tự cho phép tầng giao vận phía nhận ghép các segment lại thành thông điệp hoàn chỉnh hay phát hiện segment nào bị mất trong khi truyền.

Kiểm soát kết nối: Tầng giao vận có thể hướng nối hoặc không hướng nối. Thực thể giao vận không hướng nối xử lý segment như một gói dữ liệu độc lập và chuyển nó đến tầng giao vận của máy nhận. Một tầng giao vận hướng nối thực hiện kết nối với tầng giao vận của máy nhận trước, sau đó mới chuyển các gói dữ liệu đi. Sau khi tất cả dữ liệu được chuyển đi, kết nối được giải phóng.

Kiểm soát lưu lượng: Giống như tầng liên kết dữ liệu, tầng giao vận chịu trách nhiệm kiểm soát lưu lượng. Tuy nhiên, việc kiểm soát lưu lượng được thực hiện ở thiết bị đầu cuối chứ không phải trên một đường truyền vật lý.

Kiểm soát lỗi: Giống tầng liên kết dữ liệu, tầng giao vận chịu trách nhiệm kiểm soát lỗi. Tuy nhiên việc kiểm soát lỗi ở tầng này được thực hiện tại các thiết bị đầu cuối chứ không phải trên đường truyền trung gian. Tầng giao vận ở phía gửi đảm bảo rằng toàn bộ thông điệp đến tầng giao vận phía nhận là không bị lỗi (hỏng, mất, dư thừa). Việc khắc phục lỗi thường được thực hiện bằng cách yêu cầu truyền lại.



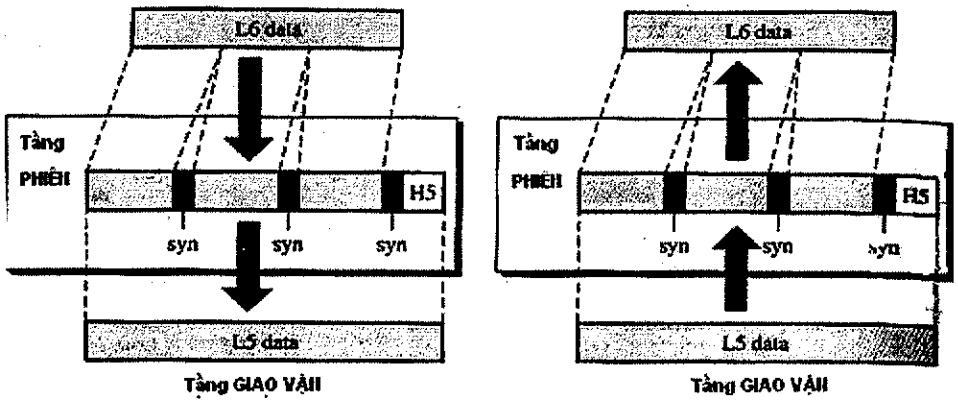
Hình 1. 21 Ví dụ về tầng giao vận

Hình 1.20 minh họa tầng giao vận. Dữ liệu đến từ các tầng trên có địa chỉ cổng là j (của ứng dụng gửi) và k (của ứng dụng nhận). Do kích thước dữ liệu lớn hơn khả năng xử lý của tầng mạng, dữ liệu được chia thành hai gói nhỏ, mỗi gói dữ liệu vẫn chứa địa chỉ cổng j và k. Tiếp theo, tại tầng mạng, địa chỉ mạng (A và P) được thêm vào mỗi packet. Các gói dữ liệu có thể đi theo các tuyến đường khác nhau, đến nơi nhận có thể không theo đúng thứ tự. Hai gói dữ liệu được chuyển đến tầng mạng của nơi nhận, tại đây header của tầng mạng được lấy ra khỏi gói dữ liệu. Hai gói dữ liệu tiếp tục được chuyển lên tầng giao vận, tại đây chúng được ghép lại để chuyển lên tầng trên.

Hệ thống bưu cục không có tầng giao vận. Trong ví dụ chuyển thư, tầng giao vận sẽ được triển khai ở người gửi và người nhận thư. Giả sử A gửi thư cho B mỗi ngày một lá thư. Hệ thống bưu cục có thể làm mất, hay gửi trễ một lá thư nào đó. B có thể phát hiện ra điều đó nếu A ghi ngày tháng viết thư trong mỗi lá thư. Nếu B không nhận thư của một ngày nào đó trong một khoảng thời gian tương đối dài, B có thể cho rằng thư đó bị mất và yêu cầu A gửi lại. Nói chung đây sẽ là cơ chế hoạt động của tầng giao vận.

Tầng phiên

Các dịch vụ của ba tầng đầu (vật lý, liên kết dữ liệu, và mạng) chưa đủ để hai tiến trình trên hai thiết bị có thể truyền thông. Tầng phiên đóng vai trò "kiểm soát viên" hội thoại (dialog) của mạng với nhiệm vụ thiết lập, duy trì và đồng bộ hóa tính liên tác giữa hai bên.



Hình 1. 22 Vai trò của tầng phiên

Tầng phiên chịu trách nhiệm về:

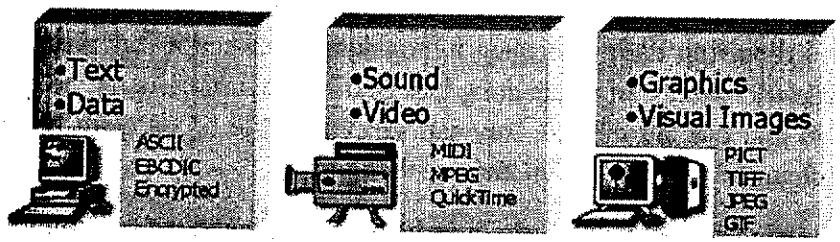
Kiểm soát hội thoại: Tầng phiên cho phép hai thực thể (tiến trình) cùng tham gia vào một cuộc hội thoại. Nó cho phép truyền thông giữa hai tiến trình được thực hiện hoặc theo chế độ bán song công hoặc song công. Ví dụ, hội thoại giữa một thiết bị đầu cuối với một mainframe có thể theo kiểu bán song công.

Đồng bộ hóa: Tầng phiên cho phép một tiến trình thêm các mốc (trong thuật ngữ mạng gọi là ***điểm đồng bộ - synchronization point***) vào luồng dữ liệu. Ví dụ, nếu hệ thống cần gửi đi một file có 2000 trang, cứ sau 100 trang nên chèn thêm các điểm đồng bộ để đảm bảo rằng việc nhận từng cụm 100 trang được thực hiện độc lập. Trong trường hợp này nếu như có lỗi khi đang truyền đi trang 523, việc truyền lại sẽ được bắt đầu từ trang 501, không cần phải truyền lại các trang từ 1 đến 500. Hình 1.23 minh họa mối quan hệ giữa tầng phiên và tầng trình diễn.

Trong một công ty nào đó có hai thư ký - một người chuyên nhận thư và một người chuyên gửi thư. Hai người thư ký này đóng vai trò tầng giao vận. Người thư ký trưởng phụ trách cả hai thư ký này đóng vai trò tầng phiên.

Tầng trình diễn

Tầng trình diễn thực hiện biểu diễn cú pháp và ngữ nghĩa các thông tin được trao đổi giữa hai hệ thống.



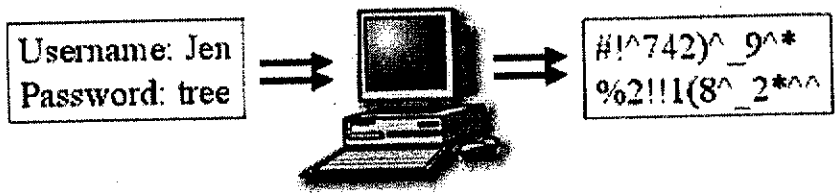
Hình 1. 23 Nhiệm vụ của tầng trình diễn

Tầng trình diễn có nhiệm vụ:

Phiên dịch (Translation): Tiến trình trên hai thiết bị trao đổi các thông tin dưới dạng chuỗi kí tự, số,... Các thông tin này sau đó được chuyển

thành chuỗi bit trước khi truyền. Do các hệ thống máy tính khác nhau sử dụng các hệ thống mã hóa khác nhau, tầng trình diễn chịu trách nhiệm chuyển đổi giữa các cách mã hóa khác nhau. Tầng trình diễn tại phía gửi chuyển thông tin theo khuôn dạng của mình thành thông tin theo khuôn dạng chung. Tầng trình diễn tại máy nhận sẽ chuyển thông tin trong khuôn dạng chung thành thông tin theo khuôn dạng của máy nhận.

Mã hóa: Hệ thống phải có khả năng đảm bảo tính bí mật khi chuyển những thông tin quan trọng. Do vậy phía gửi sẽ biến đổi thông tin ban đầu thành một dạng khác và gửi nó đến phía nhận – đây là công việc mã hóa. Phía nhận thực hiện quá trình ngược lại biến thông điệp nhận được thành dạng ban đầu. Quá trình này được gọi là giải mã.



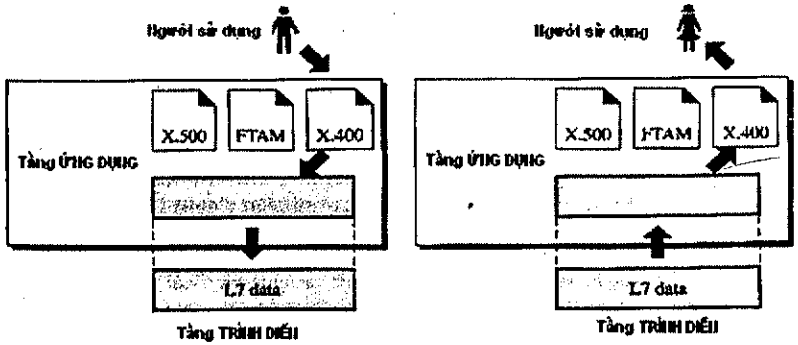
Hình 1. 24 Mã hóa dữ liệu

Nén: Nén dữ liệu làm giảm số lượng bit trên đường truyền. Nén dữ liệu ngày càng trở nên quan trọng, đặc biệt trong việc truyền các dữ liệu đa phương tiện âm thanh, hình ảnh.

Tầng ứng dụng

Tầng ứng dụng cho phép người dùng (con người hay phần mềm) truy cập vào mạng bằng cách cung cấp giao diện người sử dụng, hỗ trợ các dịch vụ như gửi thư điện tử, truy cập và chuyển file từ xa, quản lý CSDL dùng chung và một số dịch vụ khác về thông tin.

Hình 1.26 minh họa mối quan hệ giữa tầng ứng dụng với người dùng và với tầng trình diễn. Có rất nhiều ứng dụng có sẵn, ở đây chỉ đề cập đến 3 ứng dụng: X.400 (dịch vụ xử lý thông điệp), X.500 (dịch vụ thư mục), và dịch vụ truy cập, chuyển và quản lý file (FTAM). Người dùng trong ví dụ dưới đây dùng X.400 để gửi đi một thông điệp điện tử. Chú ý rằng tầng ứng dụng sẽ tạo ra dữ liệu thực sự chứ không có các thông tin tiêu đề.



Hình 1. 25 Tầng ứng dụng

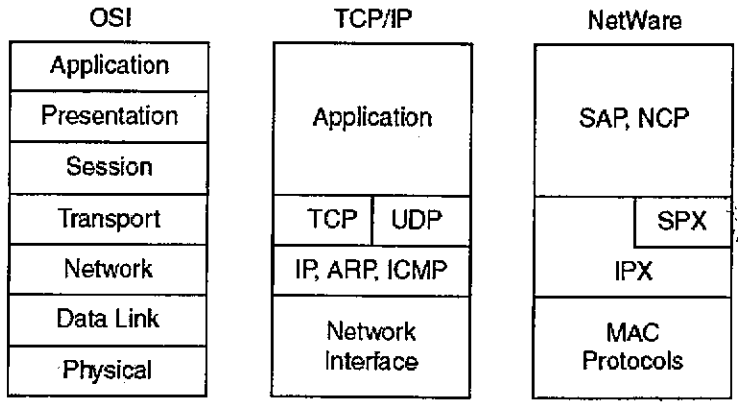
Tầng ứng dụng cung cấp các dịch vụ:

Thiết bị đầu cuối ảo của mạng: Thiết bị đầu cuối ảo là phiên bản phần mềm của thiết bị đầu cuối vật lý, cho phép người dùng đăng nhập vào một máy từ xa.

Quản lý, truy cập và chuyển file: Ứng dụng cho phép người dùng truy cập file (để viết hoặc đọc dữ liệu), lấy file, quản lý hoặc kiểm soát các file trên máy tính khác.

Các dịch vụ khác: Ứng dụng thư tín điện tử cho phép hai người trao đổi thư điện tử với nhau, ứng dụng Web cho phép người sử dụng xem trang Web được lưu trữ trên các server... Số lượng các ứng dụng mạng tăng lên rất nhanh.

1.2.3 Bộ giao thức TCP/IP – Mô hình Internet

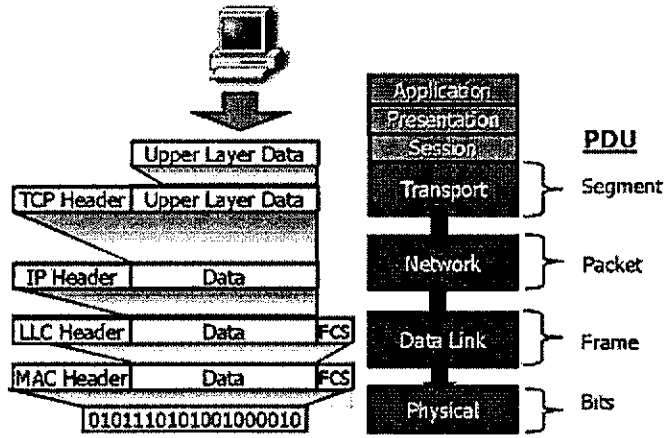


Hình 1.26 Đối chiếu mô hình OSI, mô hình Internet và NetWare

Bộ giao thức TCP/IP (được sử dụng trên Internet) ra đời trước khi có mô hình OSI. Các tầng trong bộ giao thức TCP/IP không giống hệt các tầng trong mô hình OSI. Bộ giao thức TCP/IP có 5 tầng: vật lý, liên kết dữ liệu, mạng, giao vận và ứng dụng. Bốn tầng đầu tiên cung cấp các chuẩn vật lý, giao tiếp mạng, liên mạng và chức năng giao vận tương ứng với 4 tầng đầu tiên trong mô hình OSI. Tuy nhiên 3 tầng trên cùng trong mô hình OSI được nhập thành tầng ứng dụng trong mô hình Internet (Hình 1.27).

TCP/IP là giao thức phân cấp, được tạo thành bởi các module độc lập, mỗi module cung cấp một chức năng nhất định, tuy nhiên các module này không nhất thiết phải độc lập với nhau. Mô hình OSI xác định rõ chức năng nào thuộc về tầng nào; trong khi đó các tầng của bộ giao thức TCP/IP chứa các giao thức tương đối độc lập với nhau, nhưng các giao thức này vẫn có thể kết hợp với nhau tùy thuộc nhu cầu hệ thống. Thuật ngữ "phân cấp" mang nghĩa mỗi giao thức ở tầng trên được hỗ trợ bởi một hoặc nhiều giao thức ở tầng dưới.

Tại tầng giao vận, mô hình Internet có hai giao thức: Transmission Control Protocol (TCP) và User Datagram Protocol (UDP). Tại tầng mạng là giao thức Internetworking Protocol, thường được gọi là IP.



Hình 1. 27 Dữ liệu đi từ trên xuống trong mô hình INTERNET

1.3 PHƯƠNG PHÁP TIẾP CẬN

Trong giáo trình này chúng ta sẽ giới thiệu về các công nghệ mạng theo cách tiếp cận các tầng từ trên xuống. Đầu tiên chúng ta sẽ học về tầng ứng dụng, giao vận, mạng và cuối cùng là liên kết dữ liệu. Chúng ta sẽ sử dụng mô hình Internet làm trọng tâm.

Chương 2
TẦNG ỨNG DỤNG

2.1 GIAO THỨC TẦNG ỨNG DỤNG

Sự phồng phú của các ứng dụng mạng chính là động lực phát triển của mạng máy tính. Có lẽ nếu không có chúng thì cũng sẽ không có các giao thức mạng. Trong hơn ba mươi năm qua, có nhiều phát minh đột phá trong việc phát triển các ứng dụng mạng. Bắt đầu từ thập niên 80, những ứng dụng đơn giản tương tác với người dùng qua chế độ lệnh (text-based) đã trở nên phổ biến như truy cập máy tính từ xa (telnet), thư điện tử (email), truyền file (ftp), nhóm thông tin (newsgroup), và trò chuyện từ xa (chat). Hiện nay, những ứng dụng đa phương tiện phức tạp hơn như World Wide Web, điện thoại trực tuyến, hội thảo từ xa, chia sẻ file... đã ngày càng trở nên quen thuộc.

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. Phần mềm ứng dụng mạng được cài đặt phân tán trên các **thiết bị đầu cuối (end-system)** như máy tính, điện thoại di động... Ví dụ, với Web, có hai phần mềm tương tác với nhau: phần mềm trình duyệt trong máy tính của người dùng (PC, Mac, hay trạm làm việc) và phần mềm Web server.

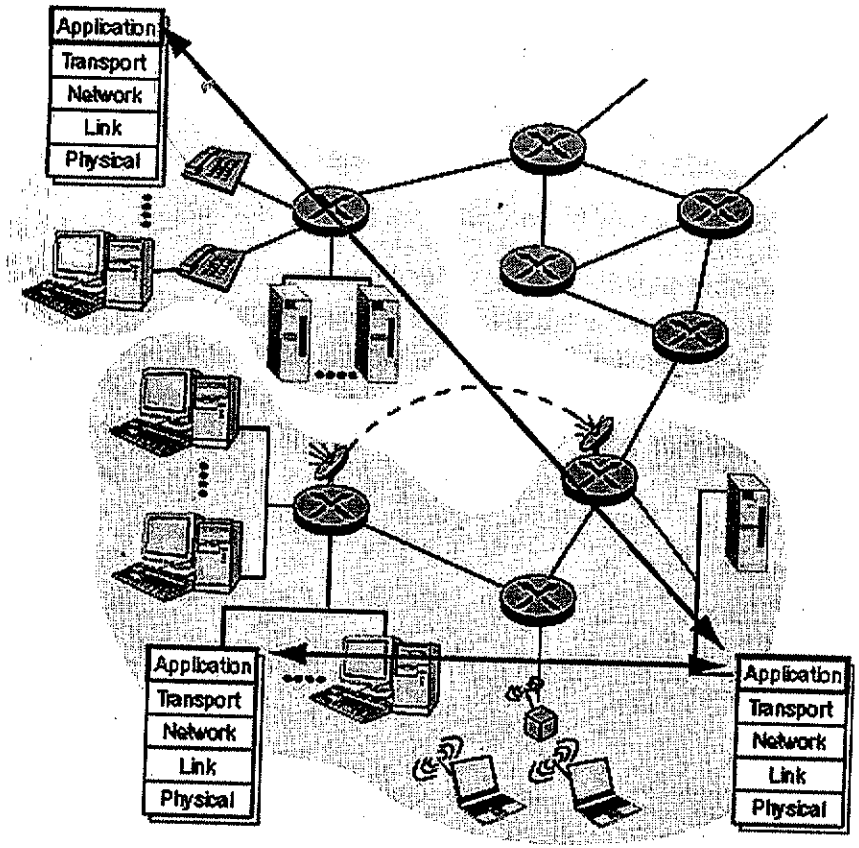
Trong thuật ngữ hệ điều hành, việc kết nối được thực hiện giữa các **tiến trình** (process) chứ không phải giữa các chương trình phần mềm. Tiến trình là một chương trình chạy trên thiết bị đầu cuối. Khi các tiến trình chạy trên cùng một thiết bị, chúng sẽ kết nối, trao đổi dữ liệu với nhau thông qua cơ chế truyền thông liên tiến trình (interprocess communication). Chính hệ điều hành của thiết bị sẽ kiểm soát cơ chế này. Trong cuốn sách này chúng

ta không quan tâm đến cách thức tiến trình trên cùng một máy tính kết nối với nhau như thế nào, mà chỉ quan tâm đến việc kết nối giữa các tiến trình trên những thiết bị khác nhau (và có thể trên những hệ điều hành khác nhau). Việc kết nối như vậy sẽ được thực hiện bằng cách trao đổi thông điệp qua mạng máy tính. Tiến trình gửi sẽ tạo và gửi thông điệp qua mạng, tiến trình nhận sẽ nhận thông điệp (message) và có thể phản hồi lại bằng cách gửi một thông điệp trả lời (xem Hình 2.1). Ứng dụng mạng có các giao thức định nghĩa khuôn dạng, thứ tự trao đổi các thông điệp cũng như hành vi của mỗi bên khi nhận được thông điệp.

Tầng ứng dụng là nơi đơn giản nhất để bắt đầu nghiên cứu về giao thức. Chúng ta sẽ làm quen với một vài ứng dụng cũng như các giao thức giữa chúng. Điều này giúp ta hiểu rõ hơn về giao thức.

2.1.1 Giao thức tầng ứng dụng

Cần phân biệt ứng dụng mạng và giao thức tầng ứng dụng. Giao thức tầng ứng dụng chỉ là một phần (cho dù là phần quan trọng) của ứng dụng mạng. Ví dụ Web - ứng dụng mạng cho phép người dùng lấy các đối tượng từ Web server bao gồm nhiều thành phần, như tiêu chuẩn định dạng văn bản (HTML), trình duyệt Web (Netscape Navigator hay Microsoft Internet Explorer), Web server (Apache, Microsoft, và Netscape server), và giao thức tầng ứng dụng. Giao thức tầng ứng dụng của Web–HTTP (HyperText Transfer Protocol [RFC 2616]), định nghĩa cách thức chuyển thông điệp giữa Web client (trình duyệt) và Web server. Như vậy HTTP chỉ là một phần của ứng dụng Web. Một ví dụ khác là ứng dụng thư điện tử. Thư điện tử cũng có nhiều thành phần, bao gồm mail server có chức năng như một hòm thư, mail reader cho phép người dùng đọc và gửi thư, chuẩn định nghĩa cấu trúc của thư điện tử và giao thức tầng ứng dụng định nghĩa cách thức chuyển thông điệp giữa mail server và mail reader, cũng như ý nghĩa của một số trường trong thư (ví dụ các tiêu đề thư: người nhận, người gửi...). Giao thức tầng ứng dụng cho thư điện tử là SMTP (Simple Mail Transfer Protocol [RFC 821]). Do đó, SMTP chỉ là một phần (cho dù quan trọng) của ứng dụng thư điện tử.



Hình 2.1 Các ứng dụng trên mạng

Như đã nói ở trên, giao thức tầng ứng dụng định nghĩa cách thức truyền thông điệp giữa các tiến trình ứng dụng chạy trên các thiết bị khác nhau. Nó xác định:

Kiểu thông điệp trao đổi, ví dụ như thông điệp yêu cầu hay thông điệp trả lời.

Cú pháp của thông điệp, ví dụ các trường trong thông điệp cũng như cách xác định chúng.

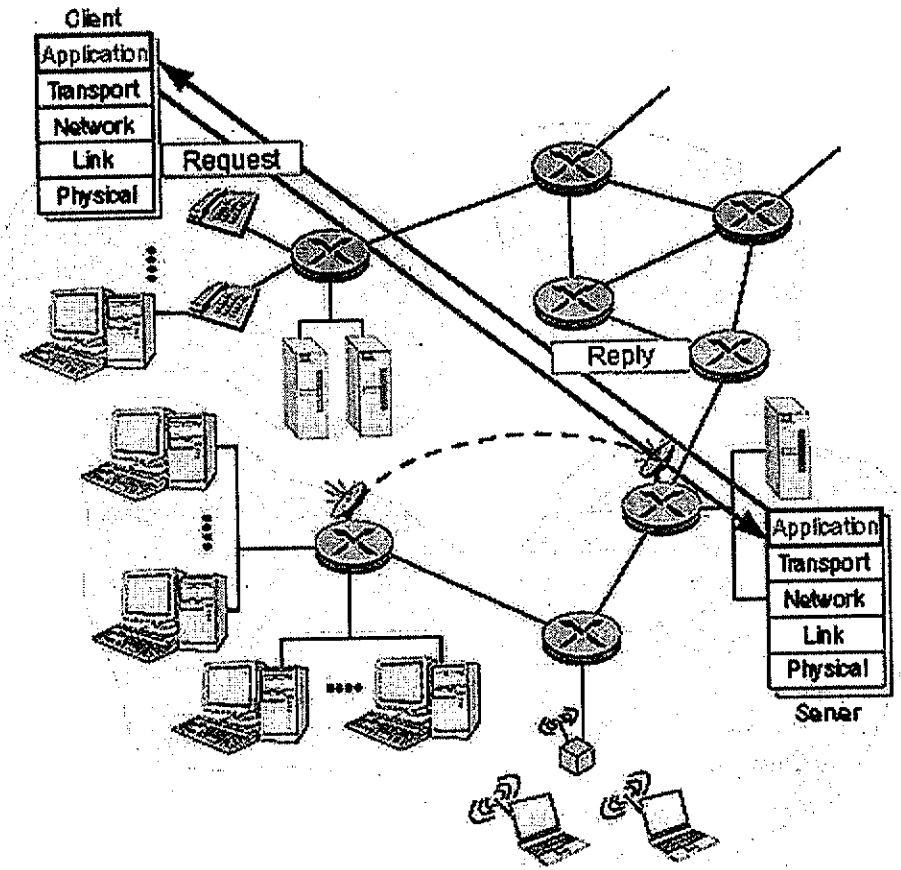
Ý nghĩa của các trường.

Qui tắc xác định khi nào và như thế nào tiến trình gửi và trả lời thông điệp.

Nhiều giao thức tầng ứng dụng được đặc tả trong các RFC. Ví dụ, đặc tả của HTTP là HTTP RFC. Nếu người thiết kế trình duyệt tuân theo các qui tắc của HTTP RFC, trình duyệt sẽ có thể lấy được trang Web từ bất kỳ Web server nào tuân theo các qui tắc HTTP RFC.

Mô hình Khách hàng / Người phục vụ (Client/Server)

Giao thức ứng dụng mạng thường chia ra hai phần hay hai phía, phía client và phía server (Xem Hình 2.2). Phía client trong thiết bị này liên lạc với phía server trong thiết bị khác. Ví dụ, trình duyệt Web là phía client, và Web server là phía server của HTTP. Trong ứng dụng thư điện tử, mail server gửi thư là phía client và mail server nhận thư là phía server của SMTP.

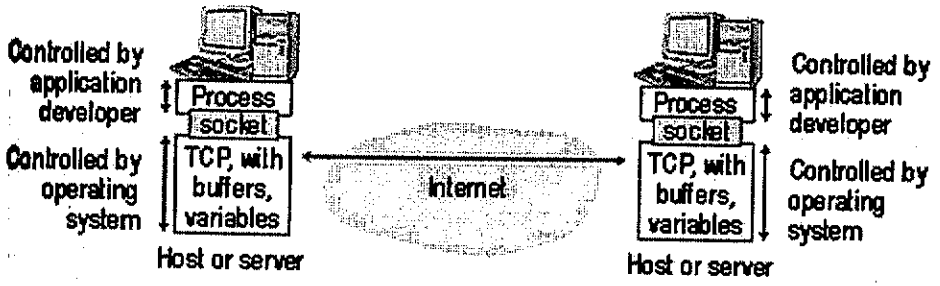


Hình 2.2 Tương tác client/server

Trong nhiều ứng dụng, máy tính sẽ thực hiện cả phần client và phần server của ứng dụng. Ví dụ xét phiên làm việc Telnet giữa máy A và máy B. (Telnet là ứng dụng đăng nhập từ xa). Nếu máy A bắt đầu trước (có nghĩa là người dùng ở máy A đăng nhập vào máy B), khi đó máy A chạy phía client và máy B chạy phía server của ứng dụng. Mặt khác, nếu máy B bắt đầu trước thì máy B chạy phía client của ứng dụng. Một ví dụ khác, FTP - được dùng để truyền file giữa hai máy. Sau khi thiết lập phiên làm việc FTP giữa hai máy tính, mỗi máy đều có thể truyền file tới máy kia trong suốt phiên làm việc. Tuy nhiên giống như hầu hết các ứng dụng mạng, máy nào bắt đầu trước được coi là client. Hơn nữa, máy tính có thể chạy cả phía client và server tại cùng một thời điểm. Ví dụ, mail server chạy phía client của SMTP khi gửi thư và chạy phía server của SMTP khi nhận thư.

Truyền thông giữa các tiến trình

Ứng dụng bao gồm hai tiến trình trên hai thiết bị khác nhau và liên lạc với nhau qua mạng. Hai tiến trình liên lạc với nhau bằng cách gửi và nhận thông điệp qua socket của chúng. Socket có thể xem như "cửa" của tiến trình vì tiến trình nhận và gửi thông điệp thông qua "cửa". Khi muốn gửi thông điệp tới tiến trình khác, tiến trình đẩy thông điệp cần gửi qua "cửa" với giả định rằng thực thể giao vận nằm bên kia "cửa" sẽ chuyển thông điệp đến "cửa" của tiến trình nhận.



Hình 2.3 Tiến trình ứng dụng, socket và giao thức giao vận

Hình 2.3 minh hoạ truyền thông qua socket giữa hai tiến trình trên Internet. (Tầng giao vận trên hình 2.3 là TCP, mặc dù ở đây có thể sử dụng giao thức khác như UDP). Qua hình vẽ ta thấy socket là giao diện giữa tầng ứng dụng và tầng giao vận trong máy tính. Nó được xem là API (giao diện lập trình ứng dụng) giữa ứng dụng và mạng. Người thiết kế ứng dụng kiểm

soát mọi khía cạnh phía bên trên socket – là tầng ứng dụng - nhưng chỉ có khả năng kiểm soát rất ít tầng giao vận phía dưới socket. Với tầng giao vận, người lập trình ứng dụng chỉ có thể kiểm soát được: (1) chọn giao thức giao vận nào và (2) xác định một vài tham số ở tầng giao vận như độ lớn bộ đệm và kích thước tối đa của gói tin. Khi người lập trình lựa chọn giao thức giao vận nào, ứng dụng được tạo ra sử dụng tầng giao vận ứng với giao thức đó. Chúng ta sẽ xem xét tỉ mỉ về socket trong phần sau.

Địa chỉ tiến trình

Để gửi thông điệp cho tiến trình trên máy tính khác thì tiến trình gửi phải xác định được tiến trình nhận. Tiến trình được xác định qua hai phần: (1) tên hay địa chỉ của máy tính, và (2) định danh xác định tiến trình trên máy tính nhận.

Đầu tiên chúng ta hãy xét địa chỉ máy tính. Trong ứng dụng Internet, máy tính được xác định qua địa chỉ IP. Địa chỉ IP sẽ được học trong chương 4. Bây giờ chúng ta chỉ cần biết địa chỉ IP là một số 32 bit dùng để xác định duy nhất một thiết bị (chính xác hơn, nó xác định duy nhất giao diện (interface) của thiết bị kết nối vào Internet). Vì địa chỉ IP của thiết bị mang tính xác định duy nhất, nên việc phân phối địa chỉ IP được quản lý chặt chẽ. Mạng ATM có một chuẩn địa chỉ khác. ITU-T lấy số điện thoại làm địa chỉ, gọi là địa chỉ E.164 [ITU 1997] sử dụng trên mạng ATM.

Ngoài địa chỉ thiết bị nhận, phía gửi phải thêm vào thông tin giúp phía nhận chuyển tiếp thông điệp cho tiến trình phù hợp (vì trong máy tính nhận có thể có nhiều tiến trình đồng thời hoạt động). Thông tin này là cổng phía nhận (destination port). Các giao thức tầng ứng dụng phổ biến đều được gán số hiệu cổng (port number) là một số cụ thể. Ví dụ, tiến trình Web server (giao thức HTTP) sử dụng cổng 80. Tiến trình mail server (giao thức SMTP) sử dụng cổng 25. Danh sách các cổng cho tất cả giao thức thường gặp trên Internet được liệt kê trong RFC 1700. Khi xây dựng ứng dụng mạng mới thì ứng dụng đó phải được đăng kí một số hiệu cổng mới.

Chương trình giao tiếp người dùng (user agent)

Trước khi bắt đầu trình bày các giao thức tầng ứng dụng, chúng ta phải nói tới khái niệm user agent. User agent là giao diện giữa người dùng

và ứng dụng mạng. Ví dụ trong ứng dụng Web, user agent là chương trình trình duyệt như Netscape Navigator hay Microsoft Internet Explorer. Trình duyệt cho phép người dùng xem trang Web, duyệt trên Web, cung cấp dữ liệu vào các form, tương tác với Java... Trình duyệt là phía client trong giao thức HTTP. Do đó khi kích hoạt, trình duyệt là tiến trình cung cấp giao diện cho người dùng, thay mặt người dùng nhận và gửi thông điệp qua socket, hiển thị thông điệp trả lời cho người dùng xem (chẳng hạn diễn dịch các mã HTML). Trong ứng dụng thư điện tử, user agent là mail reader, cho phép người dùng soạn và đọc thư. Một số phần mềm mail reader (như Eudora, Netscape Messenger, Microsoft Outlook) với hệ giao tiếp đồ hoạ có thể chạy trên PC, Mac. Mail reader chạy trên PC là phía client của giao thức tầng ứng dụng SMTP khi gửi thư và phía client của giao thức lấy thư (POP3 hoặc IMAP (xem phần 2.4)), khi nhận thư từ mail server.

2.1.2 Các yêu cầu của ứng dụng

Socket là giao diện giữa tiến trình ứng dụng và thực thể giao vận. Ứng dụng gửi thông điệp qua "cửa". Ở sau cánh cửa thực thể giao vận có trách nhiệm chuyển thông điệp qua mạng máy tính tới "cửa" tiến trình nhận. Nhiều kiểu mạng, kể cả Internet, có nhiều kiểu giao thức giao vận khác nhau. Khi thiết kế ứng dụng, bạn phải lựa chọn một giao thức giao vận có sẵn nào đó. Bạn thực hiện lựa chọn này như thế nào? Đầu tiên cần nghiên cứu các dịch vụ được các giao thức giao vận cung cấp, và sau đó sẽ chọn giao thức đáp ứng đầy đủ nhất các yêu cầu của mình. Điều này tương tự như việc chọn tàu hoả hay máy bay để di chuyển giữa hai thành phố (như Hà Nội và Huế). Bạn phải chọn một trong hai phương tiện và mỗi phương tiện cung cấp một dịch vụ khác nhau. (Ví dụ, tàu hoả giá rẻ trong khi máy bay tiết kiệm thời gian).

Ứng dụng đòi hỏi dịch vụ gì của giao thức giao vận? Về đại thể chúng ta có thể phân loại theo ba nhóm: mất mát dữ liệu, băng thông, và thời gian.

Mất mát dữ liệu (Data loss)

Một số ứng dụng như thư điện tử, truyền file, truy cập từ xa, truyền các đối tượng Web, và ứng dụng tài chính đòi hỏi dữ liệu phải được truyền

chính xác và đầy đủ, tức là không được mất dữ liệu. Đặc biệt, mất mát file dữ liệu hoặc dữ liệu trong các giao dịch tài chính có thể gây nên hậu quả nghiêm trọng. Tuy nhiên một số ứng dụng khác như ứng dụng đa phương tiện (real-time audio/video hay stored audio/video) chấp nhận mất mát dữ liệu. Trong các ứng dụng kiểu này, mất mát dữ liệu có thể gây nên một số nhiễu trong dữ liệu đa phương tiện – nhưng điều này có thể chấp nhận được trong giới hạn nào đó. Ảnh hưởng do mất mát dữ liệu tới chất lượng ứng dụng cũng như số lượng cho phép các gói dữ liệu bị mất phụ thuộc vào chính ứng dụng và phương pháp mã hoá.

Băng thông (bandwith)

Để hoạt động hiệu quả, một số ứng dụng phải truyền dữ liệu với một tốc độ nhất định. Ví dụ, ứng dụng gọi điện thoại qua Internet (Internet telephony) mã hoá âm thanh với tốc độ 32Kbs, thì sau đó dữ liệu tạo ra phải được chuyển tới ứng dụng nhận với tốc độ trên. Nếu không có đủ băng thông cần thiết, ứng dụng cần phải mã hoá âm thanh với tốc độ khác hay phải kết thúc - bởi vì nếu không đủ băng thông thì ứng dụng không thể đáp ứng yêu cầu người sử dụng. Những ứng dụng đa phương tiện hiện nay là ứng dụng phụ thuộc vào băng thông (bandwidth sensitive), nhưng trong tương lai ứng dụng đa phương tiện sẽ sử dụng các kỹ thuật mã hoá thích nghi để mã hoá tốc độ cho phù hợp với dải tần hiện có.

Thời gian (timing)

Những ứng dụng thời gian thực (real-time) mang tính chất tương tác, như Internet telephone, hội thảo qua điện thoại, hay các trò chơi nhiều người tham gia cùng một lúc (multiplayer game) yêu cầu những ràng buộc chặt chẽ về thời gian trong việc trao đổi dữ liệu. Ví dụ, những ứng dụng này đòi hỏi độ trễ (delay) từ tiến trình gửi đến tiến trình nhận không vượt quá vài trăm phần nghìn giây. Độ trễ lớn trong ứng dụng Internet telephony khiến cuộc đàm thoại bị đứt đoạn giữa chừng. Trong trò chơi nhiều người cùng tham gia hay trong môi trường tương tác ảo, độ trễ từ lúc đưa ra yêu cầu cho đến khi nhận được kết quả phản ứng từ môi trường (ví dụ, từ một người chơi khác) lớn sẽ làm giảm tính chân thực của trò chơi. Đối với ứng dụng không tính đến yếu tố thời gian thực, người ta vẫn mong muốn có một độ trễ thấp, song không có ràng buộc chặt chẽ đối với độ trễ.

Hình 2.4 tóm tắt các yếu tố như độ tin cậy, băng thông, và các đòi hỏi về thời gian của một số ứng dụng Internet phổ biến. Hình 2.4 chỉ phác hoạ một vài đòi hỏi quan trọng của những ứng dụng Internet này. Ở đây chúng ta không có đầy đủ các phân loại hoàn chỉnh, nhưng cũng đủ để nhận biết một vài đặc trưng quan trọng nhất để phân loại ứng dụng.

Ứng dụng	Mất mát dữ liệu	Băng tần	Thời gian
File transfer	Không	Elastic	Không
E-mail	Không	Elastic	Không
Web Documents	Không	Elastic (few Kbps)	Không
Real-time Audio/Video	Chấp nhận mất mát	Audio: Few Kbps - 1Mb Video: 10Kb-5 Mb	Có: 100s of msec
Stored Audio/Video	Chấp nhận mất mát	Same as Above	Có: Few Seconds
Interactive games	Chấp nhận mất mát	Few Kbps - 10Kb	Có: 100s of msec
Financial Applications	Không	Elastic	Có hoặc Không

Hình 2.4 Các yêu cầu cho một số ứng dụng

2.1.3 Dịch vụ của các giao thức giao vận Internet

Internet (và nói chung TCP/IP) cung cấp hai giao thức giao vận cho tầng ứng dụng: UDP và TCP. Khi xây dựng ứng dụng cho Internet, một trong những quyết định đầu tiên mà nhà thiết kế phải đưa ra là sử dụng UDP hay TCP. Mỗi giao thức cung cấp một kiểu phục vụ khác nhau cho ứng dụng.

TCP

Đặc trưng của giao thức TCP là hướng kết nối và cung cấp dịch vụ truyền dữ liệu tin cậy. Khi sử dụng giao thức giao vận TCP, ứng dụng sẽ nhận được cả hai loại dịch vụ này.

Dịch vụ hướng nối (connection oriented)

TCP client và TCP server trao đổi các thông tin điều khiển với nhau trước khi truyền dữ liệu ứng dụng. Quá trình "bắt tay" giữa client và server như vậy cho phép cả hai bên sẵn sàng xử lý các gói dữ liệu. Sau quá trình này, xuất hiện một đường kết nối TCP (TCP connection) giữa socket của hai tiến trình. Đây là **kết nối hai chiều (song công – full duplex)** vì cho phép hai tiến trình có thể đồng thời gửi và nhận thông điệp. Khi ứng dụng kết thúc việc gửi thông điệp, nó đóng kết nối lại. Dịch vụ này chỉ là hướng kết nối chứ không phải mạch ảo (virtual circuit) bởi vì hai tiến trình được kết nối một cách lỏng lẻo. Trong chương 3, chúng ta sẽ thảo luận chi tiết về dịch vụ hướng nối và cách thực hiện chúng.

Dịch vụ giao vận tin cậy

Tiến trình gửi có thể sử dụng TCP để truyền dữ liệu chính xác và đúng thứ tự. Gửi đi một luồng byte qua socket, tiến trình ứng dụng có thể tin tưởng TCP sẽ chuyển luồng byte này đến socket nhận, không bị lỗi hay trùng lặp byte.

TCP cũng có cơ chế kiểm soát tắc nghẽn, cơ chế này đáp ứng cho cả Internet chứ không phải cho hai tiến trình truyền thông với nhau. Kỹ thuật kiểm soát tắc nghẽn của TCP là giảm tốc độ gửi dữ liệu của mỗi tiến trình (client hay server) khi mạng bị tắc nghẽn. Đặc biệt, như chúng ta sẽ thấy trong chương 3, cơ chế kiểm soát tắc nghẽn của TCP cố gắng giới hạn mỗi kết nối TCP để chia sẻ công bằng băng thông giữa các tiến trình.

Giới hạn tốc độ truyền có thể không thoả mãn với các ứng dụng audio và video theo thời gian thực, những ứng dụng đòi hỏi phải có một băng thông tối thiểu. Hơn nữa, ứng dụng thời gian thực chấp nhận mất mát dữ liệu và không thực sự cần đến một dịch vụ giao vận tin cậy hoàn toàn. Vì các lý do đó, các ứng dụng thời gian thực thường chạy trên nền UDP.

Bây giờ chúng ta trình bày một số dịch vụ mà TCP không cung cấp. Thứ nhất, TCP không bảo đảm một tốc độ truyền tối thiểu. Tiến trình gửi không được phép truyền với bất kỳ tốc độ nào nó đề nghị, tốc độ này được kiểm soát bởi cơ chế kiểm soát tắc nghẽn của TCP. Đôi khi cơ chế này khiến tiến trình gửi phải gửi với tốc độ trung bình tương đối thấp. Thứ hai, TCP không đưa ra bất kỳ sự bảo đảm nào về độ trễ. Khi tiến trình gửi chuyển dữ liệu cho socket TCP, dữ liệu cuối cùng sẽ đến được socket nhận nhưng TCP

không bảo đảm dữ liệu sau bao lâu mới tới được đích. Với những quan sát trên môi trường Internet thực, có thể phải chờ vài giây thậm chí đến vài phút để TCP gửi thành công một thông điệp (ví dụ một trang Web HTML từ Web server đến Web client). Nói tóm lại, TCP bảo đảm việc truyền tất cả dữ liệu một cách chính xác, nhưng không bảo đảm về tốc độ truyền và độ trễ.

Dịch vụ UDP

UDP là giao thức giao vận khá đơn giản với mô hình phục vụ tối thiểu. UDP không hướng nối, nghĩa là không có giai đoạn "bắt tay" trước khi hai tiến trình bắt đầu trao đổi dữ liệu. UDP không cung cấp dịch vụ truyền tin cậy. Khi tiến trình gửi chuyển thông điệp qua cổng UDP, UDP không đảm bảo thông điệp sẽ đến được cổng tiến trình nhận. Hơn nữa, các thông điệp đến đích có thể không đúng thứ tự.

Mặt khác, UDP không có cơ chế kiểm soát tắc nghẽn, vì vậy tiến trình gửi có thể đẩy dữ liệu ra cổng UDP với tốc độ bất kì. Mặc dù không phải tất cả dữ liệu đều tới được đích, nhưng phần lớn dữ liệu có thể tới được. Ứng dụng thời gian thực thường lựa chọn UDP ở tầng giao vận. Giống TCP, UDP không bảo đảm về độ trễ.

Hình 2.5 trình bày các giao thức giao vận của các ứng dụng mạng phổ biến. Thư điện tử, truy cập từ xa, Web, và truyền file sử dụng TCP do TCP cung cấp dịch vụ truyền dữ liệu tin cậy, bảo đảm rằng mọi dữ liệu sẽ tới được đích.

Ứng dụng	Giao thức ứng dụng	Giao thức giao vận
Thư điện tử	SMTP [RFC 821]	TCP
Truy cập từ xa	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2068]	TCP
Truyền file	FTP [RFC 959]	TCP
Remote File Server	NFS	UDP hoặc TCP
Streaming Multimedia	Giao thức riêng, không công bố (ví dụ Real Networks)	UDP hoặc TCP
Điện thoại Internet	Giao thức riêng, không công bố (ví dụ Vocaltec)	Thường là UDP

Hình 2.5 Các ứng dụng phổ biến và giao thức giao vận tương ứng

Chúng ta cũng thấy rằng điện thoại qua Internet chạy trên nền UDP. Mỗi phía của ứng dụng này cần gửi dữ liệu qua mạng với tốc độ tối thiểu nào đó (Xem Hình 2.5). Hơn nữa, ứng dụng điện thoại qua Internet chấp nhận mất mát dữ liệu, vì thế chúng không cần dịch vụ truyền tin cậy của TCP.

Như đã lưu ý trước, TCP và UDP đều không bảo đảm về thời gian. Điều này có nghĩa là ứng dụng có ràng buộc về thời gian không thể chạy trên mạng ngày nay? Câu trả lời chắc chắn là không - Internet đã có một kế hoạch cho ứng dụng kiểu này trong nhiều năm tới.

2.1.4 Một số ứng dụng phổ biến

Các kiểu ứng dụng mạng ngày càng đa dạng và phong phú. Chúng ta sẽ tập trung nghiên cứu một số ứng dụng quan trọng thường gặp. Trong chương này chúng ta sẽ trình bày khá chi tiết bốn ứng dụng phổ biến: Web, truyền file, thư điện tử và dịch vụ tên miền (DNS). Web là ứng dụng đầu tiên là vì Web cực kỳ phổ biến và giao thức tầng ứng dụng của nó - HTTP, tương đối đơn giản và minh hoạ nhiều đặc trưng cơ bản của giao thức. Sau đó là ứng dụng truyền file, bởi vì ứng dụng này có nhiều đặc điểm trái ngược với HTTP. Chúng ta cũng sẽ nghiên cứu thư điện tử, một trong những ứng dụng xuất hiện đầu tiên và thông dụng nhất của Internet. Thư điện tử ngày nay sử dụng nhiều giao thức tầng ứng dụng. Web, truyền file, và thư điện tử đều yêu cầu dịch vụ truyền tin cậy, không có yêu cầu ràng buộc thời gian và yêu cầu về băng thông. Do vậy ba ứng dụng này sử dụng TCP ở tầng giao vận. Ứng dụng thứ tư là DNS (Domain Name System) cung cấp dịch vụ chỉ dẫn. Người dùng không tương tác trực tiếp với DNS mà yêu cầu dịch vụ DNS gián tiếp thông qua ứng dụng khác (ví dụ Web, truyền file, và thư điện tử). DNS minh hoạ cách thức triển khai một cơ sở dữ liệu phân tán trên mạng. Cả bốn ứng dụng trên không có yêu cầu chặt chẽ về thời gian. Các ứng dụng yêu cầu về thời gian được trình bày trong chương 6.

Tháng 4/1994, Marc Andreesen, chuyên viên máy tính (người sau này tạo ra trình duyệt Mosaic tại trường đại học Illinois•bang Urbana) cùng với Jim Clark - người sáng lập công ty Silicon Graphic (là cựu giáo sư Stanford) sáng lập tập đoàn truyền thông Nestcape. Nestcape sau đó tuyển dụng nhiều người trong nhóm dự án Mosaic ở trường đại học Illinois và cho ra đời phiên bản Beta của trình duyệt Nivigator 1.0 vào tháng 10 năm 1994. Trong những năm sau, Nestcape đã cải tiến đáng kể trình duyệt của mình, phát triển phần mềm Web server, commerce server, mail server, proxy server, mail reader và nhiều sản phẩm phần mềm ứng dụng khác. Netscape là một trong những công ty kinh doanh trên Internet đổi mới và thành công nhất trong giữa thập niên 90. Tháng 1 năm 1995, Barksdale trở thành tổng giám đốc của Nestcape và vào tháng 8, Nestcape bắt đầu bán cổ phiếu của mình trên thị trường .

Microsoft khởi đầu tương đối chậm trong lĩnh vực Internet khi đưa ra trình duyệt đầu tiên của mình - Internet Explorer 1.0 vào tháng 8 năm 1995. Internet Explorer cồng kềnh và chạy chậm nhưng Microsoft đã đầu tư lớn vào đây để đến năm 1997, Microsoft và Nestcape trở thành "kỳ phùng địch thủ" cạnh tranh thị trường trình duyệt. Ngày 11 tháng 6 năm 1997, Nestcape công bố phiên bản trình duyệt 4.0 và vào ngày 30 tháng 10 Microsoft đưa ra trình duyệt phiên bản 4.0. Tại thời điểm đó, khó có thể xác định chất lượng của trình duyệt nào tốt hơn và Microsoft - với sự thống trị của hệ điều hành Windows liên tục giành thêm được nhiều thị phần. Vào năm 1997, Nestcape mắc phải một số sai lầm nghiêm trọng trong đó có việc đầu tư rất lớn vào trình duyệt hỗ trợ JAVA. Trong suốt năm 1998, Nescape tiếp tục để mất thị phần trình duyệt Web và cả các sản phẩm khác. Cuối năm 1998 American Online đã mua lại Netscape. Marc Andreesen và các cộng sự ban đầu rời bỏ Netscape.

2.2 WORLD WIDE WEB: HTTP

Cho đến những năm 1990, Internet chỉ được sử dụng trong các cơ quan nghiên cứu, trường đại học với các dịch vụ đơn giản như truy cập từ xa, truyền file, nhận và gửi thư điện tử. Mặc dù những ứng dụng này đã (và vẫn) cực kỳ phổ biến - nhưng về cơ bản Internet vẫn chỉ được biết tới trong cộng đồng nghiên cứu. Vào đầu thập niên 90, ứng dụng quan trọng nhất của Internet - World Wide Web xuất hiện, và nhanh chóng được mọi người chấp nhận. Nó thay đổi cách thức tương tác giữa con người và môi trường làm việc. Chính điều này đã giúp đưa Internet từ một trong rất nhiều mạng thông tin (ví dụ mạng trực tuyến Prodigy, American Online hay Compuserve, hệ

thống thông tin quốc gia: Minitee/Tranpac ở Pháp, Private X25, Rrame Relay) thành một mạng thống nhất duy nhất.

Lịch sử phát triển của công nghệ viễn thông ảnh hưởng lớn đến xã hội loài người. Công nghệ đầu tiên là điện thoại - được phát minh vào năm 1870. Điện thoại cho phép hai người nói chuyện trực tiếp mà không cần ở trong cùng một vùng. Nó có những ảnh hưởng cả tốt lẫn xấu đến xã hội. Công nghệ tiếp theo là truyền thanh, truyền hình - ra đời vào những năm 1920-1930. Nó giúp con người thu nhận một lượng thông tin rất lớn bằng âm thanh và hình ảnh, và tác động lớn đến xã hội. Có lẽ công nghệ thứ ba làm thay đổi cuộc sống và công việc của con người chính là Web. Sức lôi cuốn của Web đối với con người là ở chỗ Web hoạt động theo yêu cầu (on demand). Nghĩa là có thể nhận được thông tin cần thiết vào các thời điểm cần thiết. Điều này khác so với công nghệ quảng bá (truyền thanh, truyền hình) chỉ phát đi những nội dung có sẵn tại những thời điểm định trước. Ngoài ra Web có nhiều đặc điểm lý thú khác. Ai cũng có thể dễ dàng trở thành các nhà xuất bản; các siêu liên kết và các công cụ tìm kiếm giúp ta tìm kiếm qua nhiều trang Web. Các hình ảnh đồ hoạ và hoạt hình "khuấy động" thị giác. Các thành phần khác như: Form, Java applet, Active X cho phép tương tác tới các Website khác.

2.2.1 Tổng quan về HTTP

Hyper Text Transfer Protocol (HTTP) - giao thức tầng ứng dụng của Web - là trái tim của Web. HTTP được triển khai trên cả hai phía client và server. Các tiến trình client và server trên các hệ thống đầu cuối khác nhau giao tiếp với nhau thông qua việc trao đổi các thông điệp HTTP. HTTP quy định cấu trúc thông điệp cũng như cách thức trao đổi thông điệp giữa client và server. Trước khi nói về HTTP, chúng ta hãy nói lại các thuật ngữ về Web.

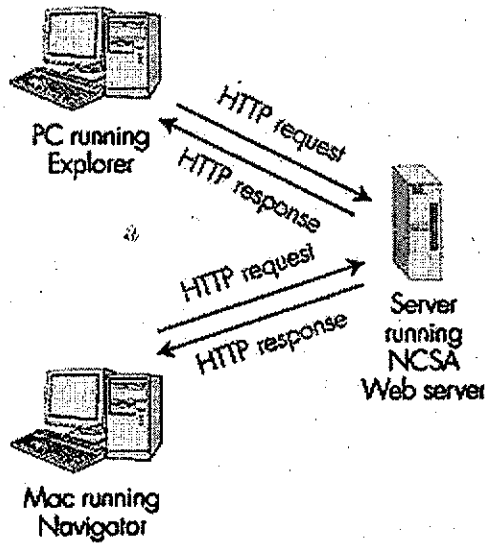
Trang Web (Webpage - hay còn gọi là một tập tin) chứa các đối tượng (Object). Đơn giản đối tượng chỉ là một file như file HTML, file ảnh JPEG, file ảnh GIF, file java applet, một đoạn âm thanh... Đối tượng được xác định qua địa chỉ URL. Trang Web chứa một file HTML cơ sở và tham

chiếu đến các đối tượng khác. Ví dụ một trang Web chứa một file HTML văn bản và 5 đối tượng ảnh JPEG, khi đó trang Web có 6 đối tượng: 1 file văn bản HTML và 5 file ảnh. File HTML cơ sở này tham chiếu đến các đối tượng khác thông qua địa chỉ URL. Mỗi địa chỉ URL có hai thành phần: tên của máy chủ và vị trí của đối tượng trên máy chủ. Đây là một địa chỉ URL

www. someschool. edu/somedepartment/picture. gif

www.someschool.edu là tên máy chủ và somedepartment/picture.gif là đường dẫn đối tượng.

Trình duyệt (Browser) - chương trình giao tiếp người dùng của ứng dụng Web cho phép hiển thị trang Web. Browser là phía client của giao thức HTTP. Hiện nay có rất nhiều phần mềm trình duyệt nhưng phổ biến nhất là Nestcape Communication và Microsoft Internet Explorer. Web server lưu giữ các đối tượng Web và được xác định qua địa chỉ URL. Phần mềm Web server là phía server của giao thức HTTP. Một số phần mềm Web server phổ biến là Apache, Microsoft Internet Information Server và Nestcape Enterprise Server.



Hình 2.6. Tương tác client/server

HTTP xác định cách thức trình duyệt yêu cầu trang Web từ Web server cũng như cách thức server gửi trang Web được yêu cầu tới trình

duyệt. Dưới đây chúng ta sẽ nói rõ hơn về quá trình trao đổi giữa client và server. Hình 2.6 minh họa quá trình này. Khi người dùng yêu cầu một đối tượng (ví dụ kích chuột vào một siêu liên kết), browser sẽ gửi thông điệp HTTP tới server yêu cầu đối tượng đó. Server nhận được yêu cầu và trả lời bằng cách gửi lại một thông điệp trả lời chứa đối tượng được yêu cầu. Cho tới những năm 1997, phần lớn các trình duyệt Web và Web server tuân thủ phiên bản HTTP 1.0 (đặc tả trong RFC 1945). Từ năm 1998 một số browser và Web server sử dụng phiên bản 1.1 theo khuyến nghị RFC 2616. Phiên bản mới này tương thích với phiên bản 1.0, nghĩa là Web server dùng phiên bản 1.1 có thể "nói chuyện" được với trình duyệt sử dụng phiên bản 1.0 và ngược lại.

Cả phiên bản 1.0 và 1.1 đều sử dụng TCP làm giao thức ở tầng giao vận phía dưới. HTTP client khởi tạo một kết nối TCP tới HTTP server. Sau khi thiết lập được kết nối, cả tiến trình browser và Web server đều truy cập tới TCP thông qua sòcket. Như đã nói ở phần 2.1, socket là "cửa" giữa tiến trình ứng dụng và thực thể TCP. Client gửi thông điệp yêu cầu qua socket. Server nhận thông điệp yêu cầu này và gửi thông điệp trả lời qua socket. Sau khi gửi thông điệp qua socket thì thông điệp nằm ngoài tầm "kiểm soát" của client và chính thực thể TCP chịu trách nhiệm chuyển nó sang phía bên kia. Trong phần 2.1 chúng ta thấy rằng TCP cung cấp dịch vụ truyền tin cậy cho HTTP, như vậy thông điệp của tiến trình client sẽ được chuyển tải nguyên vẹn đến server và ngược lại. Đến đây ta đã thấy được ưu điểm của kiến trúc phân tầng. HTTP không giải quyết việc mất mát dữ liệu mà việc này là trách nhiệm của TCP và các tầng bên dưới.

TCP sử dụng cơ chế tránh tắc nghẽn, cơ chế này sẽ được nghiên cứu chi tiết ở chương 3. Ở đây chúng ta chỉ cần biết rằng khi kết nối TCP mới khởi tạo cơ chế này đòi hỏi tốc độ truyền dữ liệu tương đối thấp nhưng sẽ tăng nhanh khi trên mạng không có tắc nghẽn. Giai đoạn bắt đầu với tốc độ thấp gọi là giai đoạn bắt đầu chậm (slow start).

Một chú ý quan trọng là server gửi các đối tượng được yêu cầu cho client mà không ghi lại bất kỳ một thông tin nào về trạng thái của client. Nếu client yêu cầu lại cùng một đối tượng thì server sẽ không thể trả lời cho client rằng đối tượng đó vừa được gửi cho client, server sẽ gửi lại cho client đối tượng đó như thể nó không biết việc gửi lần trước. HTTP server không nhớ các thông tin về client, vì thế HTTP được gọi là giao thức không trạng thái.

2.2.2 Kết nối liên tục và không liên tục (persistent / nonpersistent)

HTTP hỗ trợ cả hai cách kết nối liên tục và không liên tục. HTTP 1.0 sử dụng kết nối không liên tục. Chế độ mặc định của HTTP 1.1 là kết nối liên tục.

Kết nối không liên tục (nonpersistent)

Ta hãy xét các bước client thực hiện để yêu cầu trang Web từ server trong trường hợp sử dụng kết nối không liên tục. Giả sử trang Web có chứa một file HTML cơ sở và 10 file ảnh JPEG, đồng thời cả 11 đối tượng này cùng ở trên một server, địa chỉ của file HTML này là:

www. someschool. edu/somedepartment/home. index

Các bước thực hiện như sau:

HTTP client khởi tạo một kết nối TCP tới server có địa chỉ là **www.someschool.edu**. Cổng 80 là cổng được HTTP server sử dụng để "lắng nghe" các yêu cầu lấy trang Web từ client thông qua giao thức HTTP.

HTTP client gửi thông điệp yêu cầu qua socket tới thực thể TCP đã được kết nối ở bước trước. Thông điệp bao gồm đường dẫn **somedepartment/home.index** (ý nghĩa thông điệp sẽ được giải thích ở dưới).

HTTP server nhận được thông điệp yêu cầu từ socket, lấy đối tượng **somedepartment/home.index** trong bộ nhớ của mình (ổ cứng hoặc RAM), đặt đối tượng này vào trong một thông điệp trả lời và gửi đi qua socket.

HTTP server yêu cầu thực thể TCP kết thúc kết nối (nhưng nó không đóng lại thực sự cho đến khi client nhận được thông điệp).

HTTP client nhận được thông điệp trả lời, kết nối được đóng lại. Thông điệp chỉ ra rằng nó chứa một đối tượng là file HTML. Client sẽ lấy file đó ra từ thông điệp trả lời. File HTML tham chiếu đến 10 đối tượng ảnh JPEG.

4 bước đầu được lặp lại cho mỗi đối tượng ảnh được tham chiếu trong file HTML.

Khi nhận được thông điệp trả lời có chứa trang Web, browser sẽ hiển thị trang Web. Các browser khác nhau thì có thể có các cách hiển thị khác nhau đối với cùng một trang Web. HTTP không ảnh hưởng gì đối với cách hiển thị trang Web của client. Các đặc tả trong HTTP chỉ định nghĩa giao thức truyền thông giữa tiến trình client và server mà thôi.

Các bước ở trên sử dụng cách kết nối không liên tục vì sau khi gửi đi một đối tượng thì server sẽ đóng kết nối TCP lại, kết nối không được sử dụng để lấy các đối tượng khác. Lưu ý rằng mỗi kết nối TCP chuyển duy nhất một thông điệp yêu cầu và một thông điệp trả lời, như vậy trong ví dụ trên, client yêu cầu toàn bộ đối tượng trên trang Web thì sẽ có thể có tới 11 kết nối TCP được thiết lập.

Trong ví dụ trên, chúng ta không hề nói đến việc client nhận được 10 file ảnh JPEG qua 10 liên kết TCP riêng rẽ hay một số file được nhận qua cùng một kết nối. Trên thực tế, người dùng có thể cấu hình cho trình duyệt điều khiển mức độ song song của các kết nối. Chế độ mặc định của trình duyệt thường là từ 5 đến 10 kết nối TCP song song và mỗi kết nối kiểm soát một cặp thông điệp yêu cầu / trả lời. Nhưng nếu người dùng không thích thì có thể đặt số kết nối song song tối đa là 1, trong trường hợp này 10 kết nối được thiết lập riêng lẻ. Trong chương sau chúng ta sẽ thấy rằng cách kết nối song song làm giảm thời gian nhận được kết quả từ client.

Kết nối liên tục

Có một vài nhược điểm trong kết nối không liên tục: Thứ nhất, khi kết nối mới được tạo ra, phía client và server phải tạo ra vùng đệm TCP (buffer) cũng như lưu giữ các biến TCP. Điều này chính là gánh nặng cho server khi có nhiều client cùng yêu cầu một lúc.

Với cách kết nối liên tục, server không đóng liên kết TCP sau khi gửi thông điệp trả lời. Các thông điệp yêu cầu và trả lời sau đó (giữa cùng một client và server) được gửi qua cùng một kết nối. Trong ví dụ trên, toàn bộ đối tượng trong trang Web (một file HTML và 10 file ảnh JPEG) được truyền nối tiếp nhau trên cùng một kết nối TCP. Ngoài ra, có thể các trang Web khác trên cùng server có thể được truyền qua một kết nối TCP. Thông thường thì HTTP server đóng liên kết khi liên kết không được sử dụng trong một khoảng thời gian nào đó.

Chế độ làm việc mặc định của phiên bản HTTP 1.1 là gửi liên tục. Trong trường hợp này, HTTP client gửi yêu cầu khi nhận được một tham chiếu (ví dụ một siêu liên kết, hay tham chiếu đến file ảnh) vì vậy client có thể gửi các yêu cầu liên tiếp. Khi nhận được yêu cầu thì server sẽ gửi các đối tượng nối tiếp nhau.

2.2.3 Khuôn dạng thông điệp HTTP

Các đặc tả HTTP 1.0 (RFC 1945) và HTTP 1.1 (RFC 2016) đặc tả khuôn dạng thông điệp HTTP. Có hai kiểu khuôn dạng HTTP: thông điệp yêu cầu và thông điệp trả lời.

Thông điệp yêu cầu HTTP (HTTP request message)

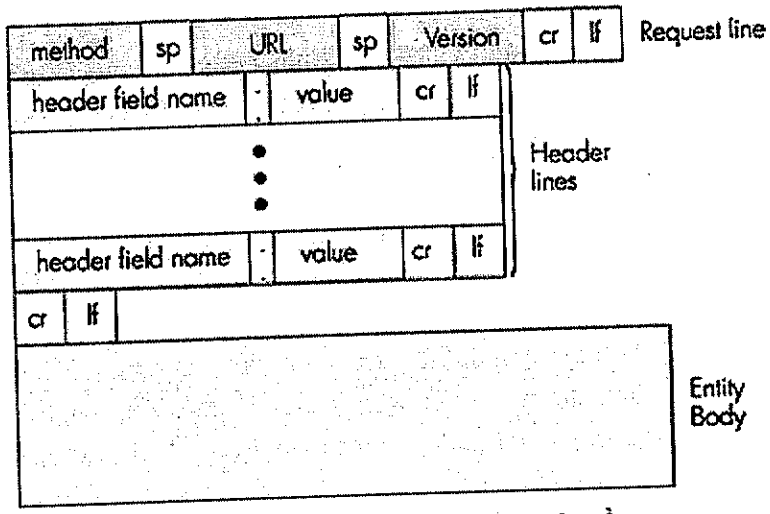
Một thông điệp yêu cầu thường có dạng sau:

```
GET /somedir. page. html HTTP/1.1
Host: www. someschool. edu
Connection:close
User-agent:Mozzilla /4.0
Accept- language:Fr
(extra carry return line feed)
```

Trước hết ta thấy rằng thông điệp được viết bằng mã ASCII - vì thế bất kỳ máy tính thông thường nào cũng có thể đọc được. Thứ hai, thông điệp gồm 5 dòng và mỗi dòng đều kết thúc bởi cặp ký tự đặc biệt Carriage Return (CR=13h) và Line Feed (LF=10h). Trên thực tế một thông điệp có thể có nhiều dòng hơn. Dòng đầu tiên của thông điệp được gọi là **dòng yêu cầu (request line)**, các dòng sau gọi là **tiêu đề (header)**. Dòng yêu cầu có 3 trường: trường method, trường địa chỉ URL và trường phiên bản HTTP. Trường method nhận một trong ba giá trị: GET, POST và HEAD. Phần lớn các yêu cầu sử dụng phương thức GET. Phương thức này được trình duyệt sử dụng để yêu cầu đối tượng có địa chỉ URL. Trong ví dụ trên thì trình duyệt yêu cầu đối tượng **somedir/page.html**. Trường phiên bản xác định phiên bản giao thức HTTP (trong ví dụ là 1.1).

Bây giờ hãy xét các trường trong tiêu đề. **Host: www.someschool.edu** là địa chỉ của máy tính có chứa đối tượng được yêu cầu. Ý nghĩa của trường Connection: close là trình duyệt yêu cầu server không sử dụng cách kết nối liên tục và yêu cầu server đóng kết nối lại sau khi đã gửi đi đối tượng được yêu cầu. Mặc dù client sử dụng phiên bản HTTP 1.1 nhưng lại không sử dụng kết nối liên tục. Trường **User-agent** xác định phần mềm trình duyệt của người sử dụng. Phần mềm trình duyệt ở đây là Mozzilla, một sản phẩm của hãng Nestcape. Trường này rất quan trọng vì server có thể gửi các bản khác nhau của cùng một đối tượng đến các trình duyệt khác nhau (các bản đối tượng này đều được xác định qua cùng một địa chỉ URL duy nhất). Cuối cùng là trường **Accept language**, trong ví dụ này người sử dụng yêu cầu bản tiếng Pháp của đối tượng - nếu server có bản này. Trong trường hợp không có thì server gửi đi bản mặc định.

Hình 2.7 minh hoạ khuôn dạng chung của thông điệp yêu cầu.



Hình 2.7 Khuôn dạng thông điệp yêu cầu

Khuôn dạng tổng quát của thông điệp có thêm trường **Entity Body** sau các dòng tiêu đề. Trường này không được sử dụng trong phương thức **GET** nhưng được sử dụng trong phương thức **POST**. HTTP client sử dụng phương thức **POST** khi người dùng điền vào một form - ví dụ khi muốn tìm kiếm qua máy tìm kiếm Google. Với phương thức **POST** người dùng vẫn yêu cầu trang Web nhưng nội dung cụ thể phụ thuộc vào nội dung điền trong form. Nếu giá trị của trường method là **POST** thì phần entity body sẽ chứa nội dung mà người dùng điền vào form. Phương thức **HEAD** cũng tương tự như phương thức **POST**. Khi nhận được yêu cầu với phương thức **POST**, server sẽ gửi lại thông điệp HTTP trả lời nhưng không gửi đối tượng được yêu cầu. Thường người ta sử dụng phương thức **HEAD** để gỡ lỗi.

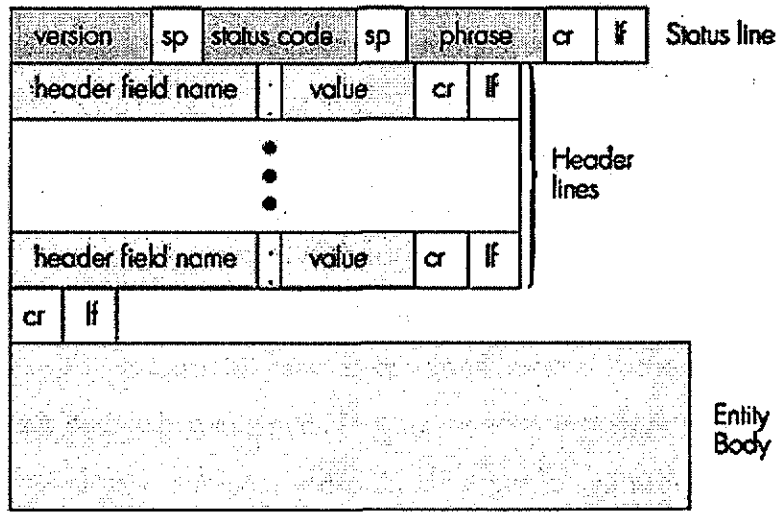
Thông điệp trả lời (HTTP response massage)

Sau đây là một ví dụ về thông điệp trả lời, thông điệp này có thể là trả lời cho thông điệp yêu cầu trên.

```
HTTP /1.1  200 OK
Connection:close
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server Apache/1. 3. 0 (unix)
Last modified:Mon, 22 Jun 1998 09:23:24 GMT
Connect lenght:6821
Connect type:text/html

( data data . . . . . )
```

Thông điệp trên gồm có 3 phần: Dòng đầu tiên là dòng trạng thái (status line), 6 dòng tiêu đề và cuối cùng là **phần thân (Entity body)** chứa đối tượng được yêu cầu (là phần **data data...**). Dòng trạng thái có 3 trường: trường phiên bản của giao thức, mã trạng thái và trường trạng thái thông điệp trả lời. Trong ví dụ này thì dòng trạng thái cho biết server sử dụng phiên bản HTTP 1.1 và trạng thái là sẵn sàng (server đã nhận được yêu cầu và gửi đối tượng được yêu cầu).



Hình 2.8 Khuôn dạng thông điệp trả lời

Trường **Connection: close** báo cho client biết server sẽ đóng kết nối sau khi gửi đi thông điệp. Trường **Date:** cho biết thời gian khi server tạo ra thông điệp và gửi đi, chú ý rằng đây không phải là thời gian khi đối tượng được tạo ra hay lần cuối cùng đối tượng được cập nhật. Đó là thời điểm mà server tìm thấy đối tượng trong hệ thống file của mình, chèn đối tượng vào thông điệp trả lời và gửi đi. Trường **Server** cho biết thông điệp trả lời này được tạo ra từ phần mềm Web server Apache, ý nghĩa của nó giống với trường **User agent** trong thông điệp yêu cầu. Trường **Last modified** là thời gian cuối cùng đối tượng được cập nhật. Ta sẽ nghiên cứu kĩ hơn về trường này nhưng chú ý rằng nó có vai trò quan trọng đối với cả client và Web cache (proxy server). **Content lenght:** cho biết độ dài của đối tượng được gửi. **Content type** xác định kiểu của đối tượng là file văn bản HTML (kiểu của đối tượng được đặt ở đây chứ không phải trong phần mở rộng của tên file).

Chú ý khi nhận được một thông điệp yêu cầu HTTP 1.0, server cũng sẽ không sử dụng kết nối liên tục ngay cả khi server dùng phiên bản 1.1. Server sẽ đóng kết nối ngay sau khi gửi đối tượng. Điều này cần thiết vì client sử dụng phiên bản HTTP 1.0 sẽ chờ server đóng kết nối lại.

Khuôn dạng chung của một thông điệp trả lời được minh hoạ trên Hình 2.8. Khuôn dạng này tương thích với ví dụ trên. Tuy nhiên cần phải nói thêm về mã trạng thái (status code) và ý nghĩa của chúng. Mã trạng thái cùng với cụm từ đi sau cho biết kết quả đáp ứng yêu cầu. Sau đây là một vài giá trị thông dụng và ý nghĩa của chúng:

200 OK: Yêu cầu được đáp ứng và dữ liệu được yêu cầu nằm trong thông điệp.

301 Moved permanetly: cho biết đối tượng đã được chuyển và địa chỉ URL mới của đối tượng được đặt trong trường **Location:** của thông điệp trả lời, phần mềm tại client sẽ tự động lấy đối tượng tại địa chỉ URL mới (đây là hiện tượng redirection thường gặp khi duyệt Web).

400 Bad Request: server không hiểu được yêu cầu từ client

404 Not found: đối tượng không còn được lưu trên server

505 HTTP version not support: server không hỗ trợ giao thức của client

Trong phần này chúng ta đã trình bày một số trường trong tiêu đề của thông điệp HTTP. HTTP (đặc biệt là bản 1.1) định nghĩa rất nhiều

trường có thể được browser, Web server và Web cache chèn vào trong thông điệp. Ở trên chúng ta mới đề cập đến một phần nhỏ, chi tiết có thể xem trong các đặc tả của HTTP.

Làm thế nào để trình duyệt cũng như server biết được phải chèn trường nào vào tiêu đề thông điệp? Thông điệp yêu cầu phụ thuộc vào chức năng trình duyệt cũng như phiên bản HTTP (HTTP 1.0 không thể tạo ra thông điệp kiểu HTTP1.1). Người sử dụng có thể định cấu hình cho trình duyệt.

2.2.4 Tương tác giữa người dùng và Hrver-server

Như đã nói trên, HTTP server không lưu giữ trạng thái. Điều này đơn giản hoá kiến trúc và làm tăng hiệu suất hoạt động của server. Tuy nhiên server muốn phân biệt người dùng không chỉ vì muốn hạn chế sự truy cập mà còn vì muốn phục vụ theo định danh người dùng. HTTP có 2 cơ chế để server phân biệt người dùng: Authentication và cookies.

Authentication (Kiểm chứng)

Nhiều server yêu cầu người dùng phải cung cấp tên (username) và mật khẩu (password) để có thể truy cập được vào tài nguyên trên máy chủ. Yêu cầu này được gọi là kiểm chứng. HTTP có các mã trạng thái và trường để thực hiện quá trình kiểm chứng. Giả sử client yêu cầu một đối tượng từ server và server yêu cầu client cung cấp tên và mật khẩu. Đầu tiên client vẫn gửi một thông điệp yêu cầu thông thường. Server sẽ trả lời với thông điệp có phần thân rỗng và trường mã trạng thái là **401 Authentication required.** Trong thông điệp trả lời này có trường **www-Authenticate**: xác định phương thức kiểm chứng mà người dùng phải thực hiện, thông thường là đưa tên và mật khẩu. Nhận được thông điệp này, client yêu cầu người dùng cung cấp tên và mật khẩu. Sau đó, client sẽ gửi lại thông điệp yêu cầu có trường **Authoziration:** trong tiêu đề, trường này chứa tên và mật khẩu của người dùng.

Sau khi nhận được đối tượng đầu tiên, client tiếp tục gửi tên và mật khẩu trong các thông điệp kế tiếp (thường thì cho đến khi người dùng đóng trình duyệt lại. Khi trình duyệt còn mở, tên và mật khẩu được lưu lại trong

cache để người dùng không phải đánh lại nữa). Theo cách này server có thể phân biệt các người dùng khác nhau. HTTP phân biệt người dùng khá lỏng lẻo và không khó để vượt qua. Chúng ta sẽ nghiên cứu thêm về vấn đề bảo mật và sơ đồ xác nhận người dùng trong chương sau.

Cookie

Cookie là kỹ thuật khác được sử dụng để ghi lại dấu vết của người truy cập. Nó được đặc tả trong RFC 2109. Ví dụ lần đầu tiên người dùng truy cập vào một server nào đó có sử dụng cookie. Thông điệp trả lời của server có trường **Set-cookies:** trong tiêu đề, cùng với một chuỗi ký tự do Web server tạo ra.

Ví dụ **Set-cookies:1678453**. Khi nhận được thông điệp trả lời, client xác định được trường **Set-cookies** và chuỗi ký tự đi kèm, trình duyệt sẽ thêm một dòng vào cuối file cookie (là một file đặc biệt nằm trên máy client). Dòng này thường là dòng chứa tên máy chủ và chuỗi ký tự cookie. Giả sử một tuần sau, client gửi thông điệp yêu cầu đến server, client sẽ tự động chèn trường **Cookies:** trong tiêu đề của thông điệp yêu cầu với giá trị là chuỗi giá trị cookie lưu trong file cookie. Trong ví dụ trên, tiêu đề chứa trường **cookies** là **Cookie:1678453.** Theo cách này, server không xác định được tên của người dùng (user name) nhưng xác định được user này chính là người đã truy cập một tuần trước đó.

Web server sử dụng cookie cho nhiều mục đích:

Nếu server yêu cầu kiểm chứng nhưng không muốn đòi hỏi người dùng đăng nhập qua tên và mật khẩu thì có thể sử dụng cookie cho mỗi lần người dùng truy cập vào server.

Server sử dụng cookie nếu muốn ghi nhớ các hoạt động của người dùng, phục vụ mục đích quảng cáo.

Nếu user mua hàng trên mạng (mua một đĩa CD chẳng hạn) thì server sử dụng cookie để ghi lại những gì mà user đã mua. Đó chính là các cửa hàng ảo.

Sử dụng cookie gây khó khăn cho người dùng không có máy cố định mà truy cập vào server từ nhiều máy khác nhau. Server sẽ coi đó là những người dùng phân biệt.

2.2.5 GET có điều kiện (Conditional GET)

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 Internet. Web cache được triển khai trên trình duyệt hay các cache server. Chúng ta sẽ nghiên cứu network cache ở phần sau. Trong phần này ta chỉ quan tâm đến cache tại trình duyệt.

Mặc dù Web cache làm giảm thời gian chờ nhận đối tượng nhưng vấn đề nẩy sinh là bản sao của đối tượng được lưu giữ trên client có thể đã "cũ", nói cách khác đối tượng trên server có thể đã thay đổi từ khi client lấy đối tượng đó về. Tuy nhiên HTTP có cơ chế cho phép sử dụng cache trong khi vẫn đảm bảo đối tượng trong cache chưa bị "cũ". Cơ chế này chính là **GET có điều kiện (conditional GET)**. Một thông điệp HTTP được gọi là có điều kiện nếu: (1) thông điệp sử dụng phương thức GET và (2) thông điệp có trường **If-modified-since** trong tiêu đề. Ví dụ, trình duyệt yêu cầu một đối tượng từ server mà trong cache của nó chưa có:

GET /fruit/banana.gif HTTP/1. 0
User-agent:Mozilla/4. 0

Sau đó server gửi thông điệp trả lời kèm với đối tượng

HTTP /1.0 200 OK
Date: wed 12 aug 1998 15:38:29
Server : Apache/1. 3. 0 (Unix)
Last-modified: mon, 22 jun 1998 09:23:24
Content-type:image/gif
(data data. . . . .)

Trình duyệt hiển thị đối tượng đồng thời lưu lại đối tượng trong cache cục bộ cùng với thời gian trong trường **Last-modified** kèm theo đối tượng.

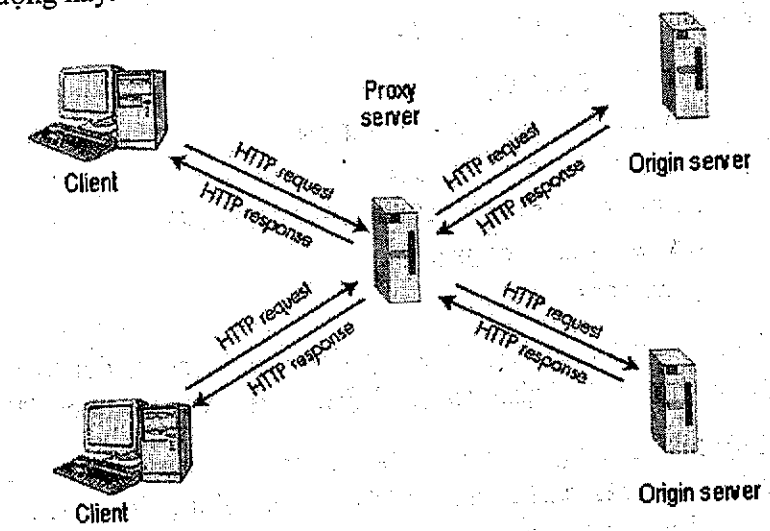
Một tuần sau, người sử dụng lại yêu cầu đối tượng này và đối tượng vẫn còn được lưu trên cache. Nhưng trên server đối tượng có thể đã bị thay đổi trong thời gian một tuần nên trình duyệt phải thực hiện kiểm tra bằng cách gửi một thông điệp GET có điều kiện, cụ thể browser gửi đi:

GET /fruit/kiwi. gif HTTP/1. 0
User-agent: Mozilla /4. 0
If-modified-since: Mon, 22 Jun 1998 09:23:24

Chú ý giá trị trường **If-modified-since:** là giá trị của trường **Last-modified:** trong tiêu đề mà server đã gửi cho client tuần trước. Thông điệp **GET** có điều kiện yêu cầu server chỉ gửi đối tượng cho client nếu như đối tượng đó bị thay đổi sau thời gian được chỉ ra trên. Giả sử đối tượng đó không thay đổi gì từ 9 giờ 23 phút 24 giây ngày 22 tháng 6 năm 1998 thì server sẽ gửi cho client thông điệp:

HTTP /1.0 304 Not modified
Date : wed, 19 Aug 1998 15:39:29
Server: Apache /1. 3. 0 (Unix)
(empty entity body)

Thông điệp trả lời này không kèm theo đối tượng. Việc gửi kèm đối tượng chỉ làm lãng phí đường truyền và làm tăng thời gian client phải chờ để nhận được đối tượng, đặc biệt khi đối tượng có kích thước lớn. Giá trị trường trạng thái là **304 Not modified** báo cho client biết đối tượng mà client lưu trong cache giống đối tượng gốc tại server, do đó client có thể sử dụng lại đối tượng này.



Hình 2.9 Client yêu cầu đối tượng thông qua cache

2.2.6 Web cache

Web cache (proxy server) là thực thể đáp ứng yêu cầu từ client. Máy tính làm nhiệm vụ Web cache có ổ đĩa riêng lưu trữ bản sao các đối tượng đã từng được yêu cầu. Như minh hoạ trên

Hình 2.9, người sử dụng có thể cấu hình trình duyệt sao cho tất cả các yêu cầu đều được gửi đến Webcache trước (việc này tương đối đơn giản với các trình duyệt của Microsoft và Nescape). Khi đó tất cả yêu cầu của trình duyệt về một đối tượng nào đó sẽ được chuyển đến Webcache trước. Giả sử trình duyệt yêu cầu đối tượng là một file ảnh có địa chỉ là **http://www.someschool.edu/campus.gif**

- Trình duyệt khởi tạo một kết nối TCP tới Webcache và gửi yêu cầu tới Webcache
- Webcache sẽ kiểm tra và tìm đối tượng, nếu tìm được thì Webcache sẽ gửi đối tượng cho client qua kết nối TCP đã được thiết lập.
- Nếu Webcache không có đối tượng đó thì nó sẽ khởi tạo một kết nối tới server thật chứa đối tượng, ở đây là **www.someschool.edu**. Sau đó Webcache gửi thông điệp yêu cầu tới cho server này thông qua kết nối TCP vừa khởi tạo. Sau khi nhận được yêu cầu từ Webcache, server sẽ gửi lại đối tượng cho Webcache
- Khi nhận được đối tượng, Webcache sẽ lưu lại bản sao của đối tượng và gửi đối tượng trong thông điệp HTTP trả lời cho máy client (thông qua kết nối TCP đã được thiết lập trước đó).

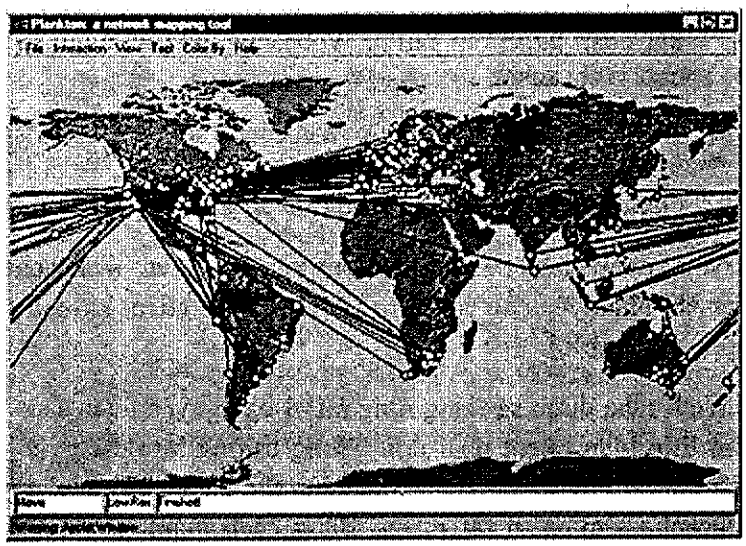
Như vậy Webcache vừa là client vừa là server. Webcache đóng vai trò server khi nhận yêu cầu và trả lời, đóng vai trò client khi gửi yêu cầu và nhận thông điệp trả lời.

Webcache được sử dụng rộng rãi vì ba nguyên nhân sau: Webcache làm giảm thời gian client phải đợi. Trong trường hợp cache có đối tượng được yêu cầu thì đối tượng này sẽ ngay lập tức được chuyển cho client. Thứ hai, Webcache làm giảm tải mạng. Bằng cách giảm tải đường truyền ra mạng Internet, cơ quan không cần phải nâng cấp đường truyền và giảm chi

phí. Webcache làm giảm lượng thông tin Web trao đổi trên Internet, do đó tăng hiệu suất hoạt động của tất cả các ứng dụng. Năm 1998, theo thống kê hơn 75% thông tin được truyền trên mạng là ứng dụng Web, vì vậy giảm tải Web cải thiện đáng kể hiệu suất toàn bộ Internet. Thứ ba, mạng Internet với nhiều Webcache giúp cho việc nhanh chóng phát tán thông tin - thậm chí ngay cho những nhà cung cấp thông tin có tốc độ server chậm hay tốc độ kết nối chậm. Nếu một nhà cung cấp có một nội dung cần phổ biến thì nội dung này ngay lập tức được chuyển đến các Webcache và yêu cầu của người dùng từ mọi nơi được đáp ứng nhanh chóng.

Cache liên hợp (Cooperative caching)

Có thể kết hợp nhiều Webcache đặt ở các vị trí khác nhau trên mạng nhằm nâng cao hiệu suất tổng thể. Ví dụ, cache của một cơ quan có thể được cấu hình sao cho các yêu cầu của nó được gửi tới cache của nhà cung cấp dịch vụ Internet cấp quốc gia (backbone ISP). Khi đó nếu cache của cơ quan không có đối tượng được yêu cầu thì nó sẽ gửi thông điệp yêu cầu HTTP đến cache của ISP. Cache ở ISP sẽ tìm đối tượng trong hệ thống lưu trữ của mình hoặc tại chính server có lưu giữ đối tượng. Sau đó nó sẽ gửi đối tượng trong thông điệp trả lời HTTP tới cache của cơ quan. Cache của cơ quan lại gửi đối tượng tới trình duyệt yêu cầu. Mỗi lần đối tượng khi qua cache đều được sao chép lại trong cache.



Hình 2.10 NLANR caching

Một ví dụ về hệ thống cache liên hợp là hệ thống cache NLANR. Hệ thống này có nhiều máy làm nhiệm vụ Webcache ở Mỹ, cung cấp dịch vụ cho các Webcache của các tổ chức và các khu vực trên toàn thế giới. Mô hình phân cấp của hệ thống này được mô tả trong Hình 2.10. Cache này lấy đối tượng từ cache khác bằng cách kết hợp sử dụng giao thức HTTP và ICP (Internet Caching Protocol). ICP là giao thức ở tầng ứng dụng cho phép cache nhanh chóng xác định một cache khác có đối tượng nào đó hay không, và nếu có thì cache có thể sử dụng giao thức HTTP để lấy đối tượng về. ICP được sử dụng rộng rãi trên rất nhiều hệ thống cache liên hợp.

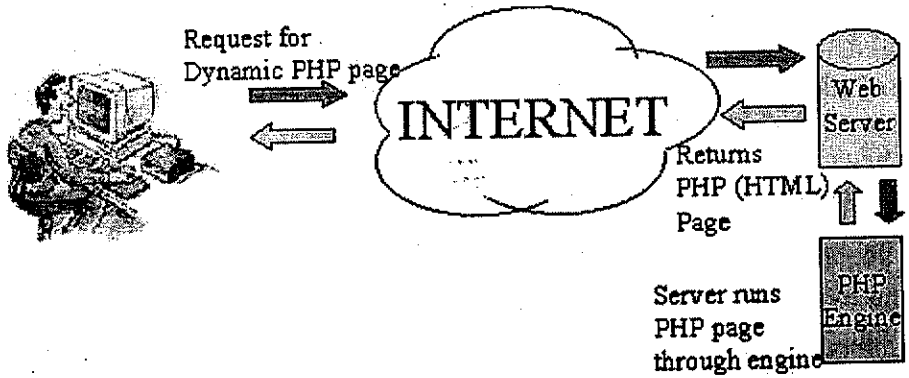
Một kiểu liên hợp khác là cụm cache (cache cluster), thường đặt trên cùng một mạng LAN. Cache được thay thế bởi cụm cache khi một cache duy nhất không đáp ứng hiệu quả trong trường hợp quá nhiều yêu cầu hay khi dung lượng thiết bị nhớ hạn chế. Tuy nhiên khi trình duyệt yêu cầu một đối tượng thì vấn đề nảy sinh là yêu cầu được gửi đến cache nào trong cụm cache. Vấn đề này có thể được giải quyết bằng cách tìm kiếm theo hàm băm (hash routing). Đơn giản nhất, 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. Nếu tất cả trình duyệt dùng cùng một thuật toán băm, đối tượng không bao giờ được lưu trên các cache khác nhau trong cụm. Nếu đối tượng thực sự được lưu trữ trong cụm thì trình duyệt luôn có thể gửi yêu cầu đến cache thích hợp. Tìm kiếm theo hàm băm là cốt lõi của giao thức Cache Array Routing (CARP).

Webcache là vấn đề lớn và phức tạp. Trong vài năm gần đây có nhiều nghiên cứu và sản phẩm về cache. Hướng nghiên cứu là xây dựng cache có khả năng xử lý được luồng âm thanh và hình ảnh. Cache sẽ đóng một vai trò quan trọng để Internet trở thành cơ sở hạ tầng cung cấp các dịch vụ hỗ trợ đa phương tiện theo yêu cầu (on demand).

2.2.7 Web động

Trong phần trên, chúng ta nói về trang Web tĩnh – là một file HTML cụ thể nằm trên Web server. Trong phần này chúng ta sẽ trình bày về Web động và Web tích cực. Một trang Web động không tồn tại dưới dạng một file cố định trên Web server. Trang Web động chỉ được server tạo ra khi

nhận được một yêu cầu cụ thể từ trình duyệt Web. Khi nhận được một yêu cầu, Web server sẽ chạy một chương trình ứng dụng nào đó để tạo ra nội dung một văn bản. Sau đó văn bản này được trả về cho trình duyệt.



Hình 2.11 Ví dụ về Web động, Web server sử dụng công nghệ PHP

Web tích cực (active Web) là loại văn bản có chứa chương trình. Chương trình này có khả năng tính toán và hiển thị thông tin. Khi trình duyệt yêu cầu, server sẽ gửi cho trình duyệt một văn bản có đính kèm chương trình. Trình duyệt sẽ chạy chương trình này tại máy tính cục bộ của mình, chương trình có thể tương tác với người sử dụng, tự động cập nhật thông tin theo nhu cầu người sử dụng. Do vậy nội dung trang Web tích cực không bất biến mà thay đổi khi chương trình tương ứng thực thi. Cơ chế Web động có ưu nhược điểm riêng so với Web tĩnh truyền thống.

Rõ ràng ưu điểm chính của Web tĩnh là tính đơn giản, tiện dụng và tin cậy. Sau khi được tạo ra, trang Web tĩnh có một định dạng cố định và bất biến. Trình duyệt có thể nhanh chóng hiển thị một trang Web tĩnh, và có thể tăng hiệu suất hệ thống bằng cách sử dụng cơ chế cache.

Nhược điểm của Web tĩnh là thiếu tính linh hoạt. Khi phải thay đổi, chúng ta phải chỉnh sửa lại mỗi trang Web tĩnh. Điều này không được làm tự động mà phải làm thủ công. Do đó trang Web tĩnh không thích hợp khi cần cung cấp các thông tin biến đổi thường xuyên.

Ưu điểm chính của Web động là khả năng hiển thị ngay lập tức thông tin hiện thời từ phía server. Những thông tin thay đổi thường xuyên, chẳng hạn dự báo thời tiết, giá các loại cổ phiếu,... có thể được một chương trình ứng dụng ở phía server sinh ra và chuyển cho trình duyệt khi có yêu cầu.

Triển khai Web động được thực hiện ở phía server. Phía client yêu cầu một trang Web động giống như khi yêu cầu một trang Web tĩnh. Đối với client, một trang Web động không khác gì một trang Web tĩnh. Do Web tĩnh hay Web động đều theo định dạng HTML, nên client không thể biết thông điệp trả về là server lấy từ file nằm trên ổ cứng hay là kết quả của một chương trình.

Nhược điểm chính của Web động là chi phí cài đặt tương đối cao và không linh hoạt khi phải hiển thị những thông tin hay thay đổi. Giống như Web tĩnh, một khi thông điệp được server sinh ra và chuyển cho trình duyệt thì nội dung thông điệp là cố định. Nếu mỗi lần thông tin thay đổi, trình duyệt phải cập nhật lại để lấy nội dung mới.

Xây dựng và bảo trì các Web server có khả năng tương tác động khá tốn kém, do ngoài khả năng phải lập trình thì máy tính làm Web server phải có cấu hình mạnh, ngoài ra phải tăng cường khả năng bảo mật của hệ thống. Việc tạo ra trang Web động cũng tốn nhiều thời gian hơn (do phía server phải chạy chương trình với mỗi yêu cầu từ client).

Đối với Web động, cho dù thông điệp trả lời chỉ được tạo ra khi có nhu cầu, nhưng thông tin trong đó vẫn có thể bị lạc hậu nhanh chóng. Ưu điểm chính của Web tích cực so với Web động chính là khả năng cập nhật thông tin liên tục do khả năng Web tích cực có khả năng tương tác trực tiếp với server để cập nhật thông tin. Ví dụ một trang Web hiển thị giá của thị trường chứng khoán có thể tự động cập nhật giá các loại cổ phiếu mà không cần bất kỳ sự can thiệp nào từ phía người dùng.

Nhược điểm chính của Web tích cực là chi phí xây dựng và khả năng an ninh hệ thống. Vì phải tải và sau đó thực thi một chương trình từ server, nên phải có khả năng đảm bảo chương trình này không làm gì có hại trên máy tính client.

Chủ yếu việc triển khai Web động được thực hiện ở phía server, server phải có bổ sung thêm khả năng chạy một chương trình ứng dụng nào đó để tạo ra nội dung một trang Web khi có yêu cầu từ trình duyệt. Mỗi một kiểu trả lời phải có một chương trình ứng dụng riêng. Do đó server phải có khả năng chuyển yêu cầu đến từ trình duyệt cho chương trình ứng dụng cụ thể.

Chuẩn CGI (Common Gateway Interface)

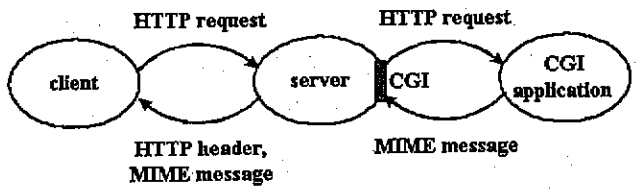
CGI là một trong những công nghệ đã từng được sử dụng rất rộng rãi khi xây dựng Web động. Chuẩn CGI quy định cách thức Web server tương tác với chương trình ứng dụng (chương trình CGI). Hình 2.12 minh họa cơ chế hoạt động của CGI.

Trên phần lớn Web server, cơ chế CGI được cài đặt như sau. Trong thư mục gốc cài đặt Web server, có thư mục cgi-bin. Khi client yêu cầu một file nào đó nằm trong thư mục cgi-bin thì Web server sẽ không gửi file được yêu cầu mà phải thực thi file này và kết quả của quá trình thực thi sẽ được gửi cho trình duyệt. Chương trình được gọi để thực thi ở đây có thể là một file khả thi hoặc một đoạn mã script (bằng Perl).

Giả sử bạn gõ **http://computer.howstuffworks.com/cgi-bin/search.pl** trên trình duyệt. Web server nhận thấy rằng file **search.pl** nằm trong thư mục cgi-bin, do vậy nó thực thi **search.pl** (là một Perl script) và sau đó gửi kết quả cho trình duyệt.

Bạn hoàn toàn có thể viết một đoạn script và thử cơ chế CGI, miễn là:

- Bạn biết một ngôn ngữ lập trình, chẳng hạn C hay PERL
- Bạn có quyền truy cập vào Web server có hỗ trợ CGI.



Hình 2.12 Vị trí của CGI

Một chương trình CGI đơn giản

Trong phần trước, chúng ta đã xem một ví dụ trang Web tĩnh đơn giản như sau:

```
<html>
 <body>
  <h1>Hello there!</h1>
 </body>
</html>
```


Nếu có quyền truy cập tới thư mục cgi-bin trên một Web server có hỗ trợ CGI, bạn có thể thử đoạn mã đơn giản được viết bằng ngôn ngữ lập trình C sau đây. Kết quả thực hiện của chương trình này tương đương với trang Web tĩnh ở trên.

```
#include <stdio.h>
int main()
{
 printf("Content-type: text/html\n\n");
 printf("<html>\n");
 printf("<body>\n");
 printf("<h1>Hello there!</h1>\n");
 printf("</body>\n");
 printf("</html>\n");
 return 0;
}
```

Dịch file mã nguồn này thành file **simplest.cgi** và đặt trong thư mục cgi-bin. Giả sử trình duyệt yêu cầu file **simplest.cgi** thì chương trình CGI tại server sẽ sinh ra một trang Web có dòng chữ "Hello there".

Chú ý rằng bắt buộc phải có dòng **printf("Content-type: text/html\n\n")** vì **"Content-type: text/html\n\n"** luôn là dòng đầu tiên Web server gửi cho trình duyệt.

Ưu điểm chính của chương trình CGI là tạo ra nội dung động – nghĩa là kết quả của mỗi lần chạy là khác nhau. Chương trình C sau đây minh họa điều này.

```
#include <stdio.h>
int incrementcount()
{
 FILE *f;
 int i;
 f=fopen("count.txt", "r+");
 if (!f)
 {
   sleep(1);
   f=fopen("count.txt", "r+");
   if (!f)
     return -1;
 }

 fscanf(f, "%d", &i);
 i++;
```

```
  fseek(f,0,SEEK_SET);
  fprintf(f, "%d", i);
  fclose(f);
  return i;
}

int main()
{
  printf("Content-type: text/html\n\n");
  printf("<html>\n");
  printf("<body>\n");
  printf("<h1>The current count is: ")
  printf("%d</h1>\n", incrementcount());
  printf("</body>\n");
  printf("</html>\n");
  return 0;
}
```

Sau đó dịch file này thành file **counter.cgi**. Tạo ra file **count.txt** có chứa số **0** trong đó và đặt cả hai file này vào thư mục cgi-bin. Sau đó thử kiểm tra bằng cách kích vào link **http://computer.howstuffworks.com/cgi-bin/count.cgi** một số lần. Mỗi lần sẽ hiện ra một kết quả khác nhau.

File **count.txt** giữ giá trị hiện thời của biến **count**, và hàm **incrementcount()** làm tăng biến **count** trong file **count.txt**. Hàm này mở file **count.txt**, đọc giá trị biến count, tăng biến count lên một sau đó ghi lại giá trị mới vào file.

Form: Gửi thông tin đến trình duyệt

Chúng ta thấy rằng chương trình CGI tương đối đơn giản. Web server chỉ gọi một chương trình CGI để thực thi và sau đó kết quả được gửi lại cho trình duyệt. Bây giờ chúng ta xem cách thức làm sao để trình duyệt gửi được thông tin cho chương trình CGI. Cách thức đơn giản ở đây chính là HTML form.

Để hiểu thế nào là form, chúng ta thử tạo ra một file HTML đơn giản có tên **simpleform.htm** với nội dung như sau

```
<html>
<body>
 <h1>A super-simple form<h1>
 <FORM METHOD=GET ACTION="http://computer.howstuffworks.com/
```



```
cgi-bin/simpleform.cgi">
 Enter Your Name:
 <input name="Name" size=20 maxlength=50>
 <P>
 <INPUT TYPE=submit value="Submit">
 <INPUT TYPE=reset value="Reset">
 </FORM>
</body>
</html>
```

File HTML này bạn đặt trong thư mục cgi-bin của Web server. Nếu để ý kỹ, bạn sẽ thấy trong đoạn mã HTML sử dụng phương thức GET để gửi đến CGI script có địa chỉ **http://computer.howstuffworks.com/cgi-bin/simpleform.cgi**. Trong form có vùng văn bản để người dùng đưa dữ liệu vào và hai nút chuẩn kiểu Submit và Reset.

File **Simpleform.cgi** được tạo ra bằng chương trình C đơn giản **simpleform.c** sau:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
 printf("Content-type: text/html\n\n");
 printf("<html>\n");
 printf("<body>\n");
 printf("<h1>The value entered was: ")
 printf("%s</h1>\n", getenv("QUERY_STRING"));
 printf("</body>\n");
 printf("</html>\n");
 return 0;
}
```

Và sau khi biên dịch được đặt trong thư mục cgi-bin. Chương trình này sẽ lấy giá trị được trình duyệt gửi về qua form để rồi hiển thị nó.

Các kỹ thuật phía Server

Một phương pháp giúp Web server tạo nội dung động là các công nghệ phía server (server side technology). Ngày nay có khá nhiều công nghệ như vậy.

ASP (*Active Server Pages*): Là công nghệ của Microsoft, có phần mở rộng là .asp.

PHP (*Personal Home Pages*): Công nghệ mã nguồn mở, phần mở rộng là php or.php3 (phụ thuộc vào cấu hình server).

JSP (*Java Server Pages*): Dựa trên ngôn ngữ lập trình Java, có phần mở rộng là .jsp.

Với những công nghệ trên, dễ dàng cài đặt và bảo trì cho một Website lớn. Người phát triển chỉ cần gắn các đoạn mã (phía server) vào các trang HTML. Đoạn mã này được đưa cho trình biên dịch tương ứng để từ những câu lệnh sinh ra các đoạn mã HTML và sau đó trang HTML được gửi về cho trình duyệt. Chú ý rằng trình duyệt sẽ không biết về đoạn mã được gắn bên trong trang Web, trình duyệt chỉ nhận được một trang Web với mã HTML thuần túy do server gửi về.

Chúng ta thử lấy PHP làm ví dụ. Yêu cầu của trình duyệt tới một trang PHP sẽ được Web server chuyển cho bộ biên dịch PHP (cùng với các dữ liệu có liên quan). Bộ biên dịch PHP sẽ chạy đoạn mã PHP để sinh ra nội dung HTML và chuyển kết quả này cho Web server. Web server tiếp tục chuyển kết quả nhận được cho trình duyệt yêu cầu. Trình duyệt chỉ nhận được mã HTML, hiển thị chúng mà không biết cách thức Web server tạo ra trang HTML.

Một chương trình PHP đơn giản

Tạo ra một file **hello.php** có nội dung sau và đặt trong thư mục chủ cài đặt PHP của Web server (**DOCUMENT_ROOT**)

```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
 <?php echo '<p>Hello World</p>'; ?>
 </body>
</html>
```

Sử dụng trình duyệt để truy cập vào file hello.php này (gõ **http://Webserver//hello.php** trên thanh địa chỉ của trình duyệt). Nếu phía server cấu hình PHP chính xác thì phía trình duyệt sẽ nhận được kết quả sau:


```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
 <p>Hello World</p>
 </body>
</html>
```

Chương trình trên cực kỳ đơn giản. Nó chỉ hiển thị dòng chữ "**Hello World**" trên trình duyệt bằng cách sử dụng lệnh **echo( )** của PHP.

Điểm quan trọng trong ví dụ này là các thẻ PHP đặc biệt. Trong ví dụ này ta sử dụng thẻ <?php để xác định điểm khởi đầu và thẻ ?> đánh dấu điểm kết thúc của một đoạn mã PHP. Đoạn mã PHP có thể được đặt vào bất kỳ đâu trong file mã nguồn PHP và đoạn mã này sẽ được trình biên dịch PHP xử lý.

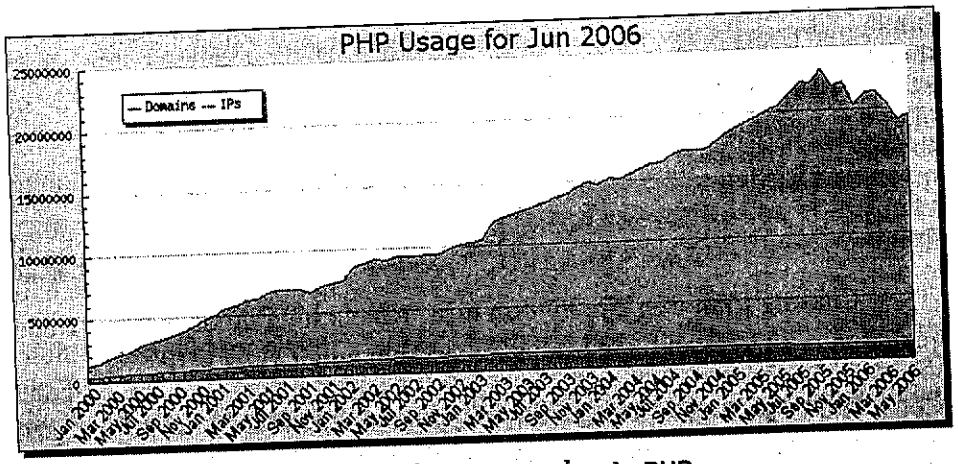
Lịch sử PHP

PHP được Rasmus Lerdorf công bố chính thức từ khoảng mùa thu năm 1994. Trước đó Rasmus đã sử dụng chương trình này để kiểm soát số người truy cập vào Website cá nhân của mình. Phiên bản PHP đầu tiên được công bố vào đầu năm 1995 và được xem là viết tắt của Personal Home Page. Nó chỉ bao gồm một bộ phân tích cực kỳ đơn giản với một vài macro, một vài tiện ích để hỗ trợ xây dựng một Website cá nhân có guessbook và counter. Bộ phân tích được viết lại vào giữa năm 1995 và trở thành phiên bản PHP/FI 2. Module FI thực hiện việc tạo ra file HTML từ một dữ liệu cho trước. Rasmus đã kết hợp module Form Interperter và thêm các hỗ trợ mSQL. Bắt đầu từ đó, công nghệ PHP được sử dụng cực kỳ rộng rãi và được nhiều người cùng bắt tay phát triển cùng.

Rất khó đưa ra các dự đoán thật chính xác, nhưng vào khoảng cuối năm 1996, có khoảng 15000 Website sử dụng PHP/FI. Đến giữa năm 1997, số lượng này tăng lên 50000. Tại thời điểm này, PHP không còn là dự án mang tính "ngẫu hứng" của Rasmus nữa mà trở thành một dự án có tổ chức với nhiều người cùng tham gia phát triển. Bộ phân tích được Zeev Suraski và Andi Gutmans viết lại từ đầu – và trở thành cơ sở cho PHP3. PHP3 – bên cạnh khá nhiều tiện ích của PHP/FI – cũng được viết lại và bổ sung khá nhiều phần.

Đến tháng 10/2000, PHP3 và PHP4 được cài đặt trên 3300000 Website (so với con số 3800000 Website cài IIS). Phiên bản 4 sử dụng cơ chế Zend để nâng cao hiệu suất. Tháng 12/2004, phiên bản 5 ra đời với nhiều tính năng nổi bật

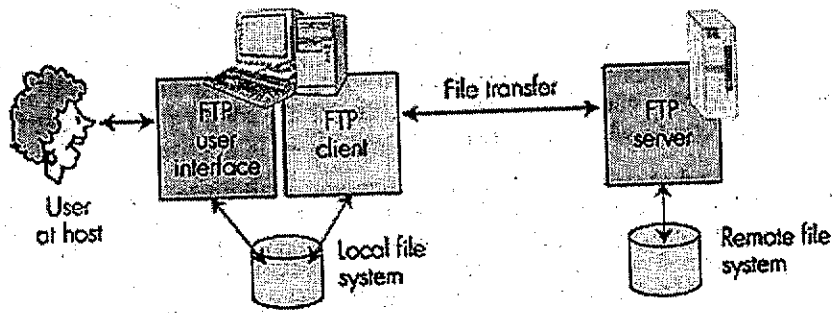
Hình 2.13 minh họa tốc độ phát triển của PHP



Hình 2.13 Tốc độ phát triển của PHP

2.3 TRUYỀN FILE (FILE TRANSFER) FTP

FTP (File Transfer Protocol) là giao thức truyền file giữa các máy tính. Giao thức này xuất hiện từ những năm 1971 (khi Internet vẫn chỉ là một dự án thử nghiệm) nhưng vẫn còn được sử dụng rộng rãi cho đến tận ngày nay. FTP được đặc tả trong RFC 959. Hình 2.14 minh hoạ các dịch vụ của FTP.

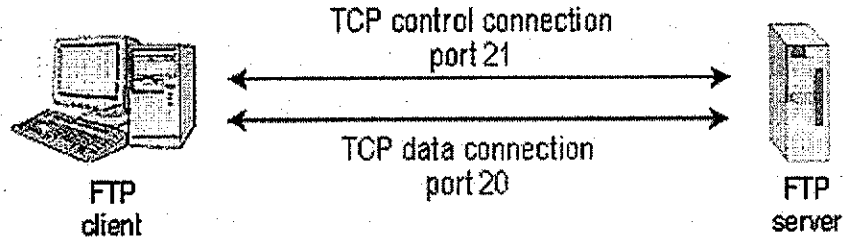


Hình 2.14 FTP cho phép trao đổi file giữa hai máy tính

Trong phiên làm việc của FTP, người dùng làm việc trên máy tính của mình và trao đổi file với một máy tính khác. Để truy cập tới máy tính khác, người dùng phải đăng nhập bằng cách cung cấp định danh người dùng

và mật khẩu. Sau khi những thông tin này được kiểm chứng thì công việc truyền file từ hệ thống file trên máy tính của mình đến hệ thống file ở đầu kia mới có thể được thực hiện.

Như mô tả trên Hình 2.15, người dùng tương tác với FTP thông qua chương trình giao tiếp người dùng của FTP. Đầu tiên người dùng đánh tên máy tính cần truyền file. Tiến trình FTP ở client khởi tạo một kết nối TCP tới tiến trình FTP server, sau đó người dùng đưa các thông tin về tên và mật khẩu để server kiểm chứng. Sau khi được server xác định, người dùng mới có thể thực hiện việc trao đổi file giữa hai hệ thống file.



Hình 2.15 FTP gồm hai đường: kiểm soát và dữ liệu

HTTP và FTP đều là giao thức truyền file và có rất nhiều đặc điểm chung như cả hai đều sử dụng các dịch vụ của TCP. Tuy vậy hai giao thức này có những điểm khác nhau cơ bản. Điểm khác nhau nổi bật nhất là FTP sử dụng hai kết nối TCP song song, một đường truyền thông tin điều khiển (control connection) và một đường truyền dữ liệu (data connection). Các thông tin điều khiển như thông tin định danh người dùng, mật khẩu truy nhập, lệnh thay đổi thư mục, lệnh "put" hoặc "get" file giữa hai máy tính được trao đổi qua đường truyền thông tin điều khiển. Đường truyền dữ liệu để truyền file dữ liệu thực sự. Vì FTP phân biệt luồng thông tin điều khiển với luồng dữ liệu nên nó được gọi là gửi thông tin điều khiển out-of-band. Giao thức RTSP dùng để truyền âm thanh và hình ảnh liên tục cũng sử dụng cách gửi thông tin điều khiển kiểu out-of-band. Như đã nói, HTTP gửi tiêu đề của thông điệp và file dữ liệu trên cùng một kết nối TCP. Vì vậy mà HTTP được gọi là gửi thông tin điều khiển in-band. Trong phần tiếp theo ta sẽ thấy rằng SMTP - giao thức gửi thư điện tử cũng sử dụng truyền thông tin điều khiển kiểu in-band. Đường truyền thông tin điều khiển và đường truyền dữ liệu của giao thức FTP được minh hoạ trong Hình 2.15.

Khi người dùng bắt đầu một phiên làm việc FTP, đầu tiên FTP sẽ thiết lập một đường kết nối thông tin điều khiển TCP qua cổng 21. Phía client của giao thức FTP gửi thông tin về định danh người dùng và mật khẩu cũng như lệnh thay đổi thư mục qua kết nối này. Khi người dùng có một yêu cầu trao đổi file (truyền từ/đến máy người dùng), FTP mở một kết nối TCP để truyền dữ liệu qua cổng 20. FTP truyền đúng một file qua kết nối này và ngay sau khi truyền xong thì đóng kết nối lại. Nếu trong cùng phiên làm việc, người dùng có yêu cầu truyền file thì FTP sẽ mở một kết nối khác. Như vậy với FTP, luồng thông tin điều khiển được mở và tồn tại trong suốt phiên làm việc của người dùng, nhưng mỗi kết nối dữ liệu được tạo ra cho mỗi một yêu cầu truyền file (kết nối dữ liệu là không liên tục).

Trong suốt phiên làm việc, FTP server phải giữ lại các thông tin về trạng thái của người dùng, đặc biệt phải kết hợp các thông tin điều khiển với tài khoản của người dùng. Server cũng lưu giữ thư mục hiện thời mà người dùng truy cập cũng như cây thư mục của người dùng. Ghi lại các thông tin trạng thái của mỗi phiên làm việc hạn chế đáng kể tổng số phiên làm việc đồng thời. HTTP không lưu giữ trạng thái nên không phải ghi lại bất kì thông tin nào về trạng thái người dùng.

2.3.1 Các lệnh FTP (FTP Commands)

Lệnh (yêu cầu) từ client đến server và kết quả (trả lời) từ server tới client được gửi thông qua kết nối điều khiển và được mã hoá bằng bảng mã ASCII 7 bit. Do vậy giống như lệnh HTTP, người ta có thể đọc được lệnh FTP. Trường hợp các lệnh viết liên tục thì cặp 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, một số lệnh có tham số. Sau đây là một số câu lệnh hay gặp:

USER username: sử dụng để gửi thông tin định danh người dùng cho server

PASS password: dùng để gửi password cho server

LIST: dùng để yêu cầu server gửi một danh sách các file trong thư mục hiện thời. Danh sách này được gửi thông qua một kết nối dữ liệu TCP

RETR filename: dùng để lấy một file từ thư mục hiện thời (trên máy ở xa)

STOR filename: dùng để tải một file vào thư mục hiện thời (trên máy ở xa)

Thông thường có quan hệ 1-1 giữa lệnh của người dùng và lệnh của FTP. Ứng với mỗi lệnh từ client là một trả lời của server. Câu trả lời là một mã 3 chữ số và có thể có một thông báo kèm theo. Điều này tương tự như trường mã trạng thái trong thông điệp trả lời HTTP. Dưới đây là một số câu trả lời thường gặp:

331 username OK, password requiered

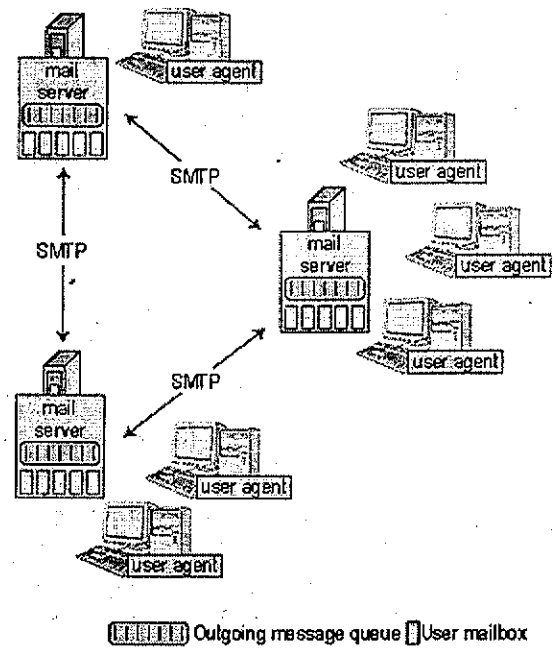
125 connection already open; Transfer staring

425 can't open data connection

452 error writing file

Chi tiết về FTP có thể tham khảo tại khuyến nghị RFC 959

2.4 THƯ TÍN ĐIỆN TỬ (E-mail) TRÊN INTERNET



Hình 2.16 Mô hình hệ thống email đơn giản

Cùng với Web, thư điện tử là một trong những ứng dụng Internet thông dụng nhất. Gần giống thư tín thông thường, e-mail là dịch vụ không

đòi hỏi đồng bộ - nghĩa là mọi người gửi và đọc thư khi thấy thuận tiện, không cần theo kế hoạch trước. Nhưng khác với thư tín thường, e-mail nhanh, dễ gửi và chi phí thấp. Hơn nữa, những thông điệp e-mail ngày nay có thể chứa đựng các hyperlink, văn bản định dạng HTML, hình ảnh, âm thanh và cả video. Trong phần này, chúng ta sẽ khảo sát các giao thức trao đổi thư thuộc tầng ứng dụng trên Internet.

Hình 2.16 minh hoạ hệ thống mail trên Internet gồm có 3 thành phần chính: user agent, mail server và SMTP (Simple Mail Transfer Protocol). Để tiện theo dõi, chúng ta sẽ lấy ví dụ Alice gửi e-mail cho Bob để mô tả 3 thành phần trên. Chương trình giao tiếp người dùng cho phép đọc, hồi âm, gửi, lưu giữ và soạn thảo các thư (user agent dành cho e-mail còn được gọi là mail reader – trình đọc thư. Mặc dù vậy, trong cuốn sách này, chúng ta sẽ tránh sử dụng thuật ngữ đó). Khi Alice soạn thảo xong thư, user agent của Alice sẽ gửi thư tới mail server của Alice, tại đây thư được đặt vào trong hàng đợi để gửi ra ngoài. Khi Bob muốn đọc thư, user agent của Bob sẽ lấy thư trên hộp thư (mail box) của Bob tại mail server. Trong những năm cuối thập kỷ 90, các user agent có giao diện đồ họa GUI (Graphic User Interface) khá thông dụng, chúng cho phép người dùng có thể xem và soạn thảo các thư có gắn tài liệu đa phương tiện. Hiện nay, những phần mềm soạn e-mail thông dụng là Eudora, Microsoft Outlook và Nescape Messenger. Có nhiều chương trình user agent có giao diện dựa trên nền văn bản gõ lệnh như là mail, pine và elm.

Trong lịch sử HOTMAIL

Tháng 12/1995, Sabeer Bhatia và Jack Smith đề nghị Draper Fisher Jurvetson - một nhà đầu tư mạo hiểm phát triển một hệ thống email miễn phí trên nền Web. Ý tưởng là cung cấp miễn phí tài khoản (hòm thư) và việc truy cập tới hòm thư có thể thực hiện thông qua Web. Khi đó bất kỳ ai truy cập được vào Internet – dù ở nhà hay ở cơ quan - đều có khả năng đọc và gửi thư. Hơn thế nữa, hình thức này khá tiện dụng đối với người dùng hay phải di chuyển. Draper Fisher Jurvetson tài trợ cho Bhatia and Smith lập công ty Hotmail đổi lấy 15% giá trị công ty. Với 3 nhân viên làm cả ngày và 12 đến 14 nhân viên bán công nhật trả lương bằng cổ phiếu của chính công ty, Hotmail đã phát triển và cung cấp dịch vụ đầu tiên vào tháng 7 năm 1996. Sau một tháng họ có 100000 người sử dụng dịch vụ. Số lượng người sử dụng tăng lên nhanh chóng – và tất cả người sử dụng đều phải đọc banner quảng cáo trong email. Tháng 12/1997, chưa đầy 18 tháng sau khi khai trương, Hotmail có 12 triệu người sử dụng và đã được Microsoft mua lại với giá 400 triệu đôla.

Có hai yếu tố quan trọng trong thành công của Hotmail: sự tiên phong và khả năng tự quảng cáo. Hotmail là "người tiên phong" vì Hotmail là công ty đầu tiên cung cấp dịch vụ Web mail. Các công ty khác, sử dụng ý tưởng của Hotmail - đều đi sau Hotmail sáu tháng. Email là một ví dụ điển hình về sự tự quảng cáo. Khi nhận được thư từ dịch vụ Yahoo, người nhận sẽ biết được về Yahoo.

Máy chủ phục vụ thư (Mail server) là thành phần cốt lõi trong hệ thống e-mail. Mỗi người có một hộp thư đặt trên mail server. Hộp thư của Bob quản lý, lưu giữ các thư gửi tới Bob. Thư được tạo ra tại user agent của người gửi, được gửi tới mail server của người gửi, rồi tới mail server của người nhận – và cuối cùng được chuyển vào hộp thư của người nhận. Khi Bob muốn truy cập vào hộp thư của mình, mail server chứa hộp thư của Bob sẽ kiểm chứng Bob (thông qua username và password). Mail server của Alice cần phải xử lý khi mail server của Bob gặp sự cố. Nếu không thể gửi thư cho mail server của Bob, mail server của Alice sẽ giữ những thư đó trong hàng đợi gửi thông điệp và sẽ cố gắng gửi lại thông điệp. Quá trình gửi lại được tiến hành thường xuyên 30 phút một lần trong năm ngày. Và sau đó, nếu vẫn không thành công thì server sẽ huỷ bỏ thư và gửi thư báo cho người gửi (Alice).

SMTP (Simple Mail Transfer Protocol) là giao thức gửi thư điện tử của tầng ứng dụng. SMTP sử dụng dịch vụ truyền dữ liệu tin cậy của TCP để truyền thư từ mail server của người gửi đến mail server của người nhận. Giống các giao thức khác ở tầng ứng dụng, SMTP có 2 phía: phía client, trên mail server của người gửi và phía server, trên mail server của người nhận. Tất cả các mail server đều chạy cả hai phía client và server của SMTP. Mail server đóng vai trò client khi gửi thư, đóng vai trò server khi nhận thư.

2.4.1 SMTP

SMTP là trái tim của dịch vụ gửi thư trên Internet và được đặc tả trong RFC 821. SMTP truyền các thông điệp (thư) từ mail server của người gửi đến mail server của người nhận. SMTP ra đời trước HTTP khá lâu (RFC đặc tả SMTP có từ năm 1982 và SMTP đã xuất hiện trước đó một thời gian dài). Mặc dù có nhiều ưu điểm nên được tất cả mail server trên Internet sử dụng, SMTP vẫn là một công nghệ cũ nên chắc chắn có những đặc tính "lạc hậu". Ví dụ SMTP đòi hỏi phần thân của tất cả các thông điệp e-mail phải mã hoá theo bảng mã ASCII 7 bit. Sự hạn chế này là do trong những năm đầu thập kỷ 80, với số đường truyền ít ỏi, không ai gửi thư cùng với những phần đính kèm lớn, hay gửi kèm các file hình ảnh, âm thanh có kích thước

lớn. Nhưng trong kỷ nguyên đa phương tiện ngày nay, việc giới hạn mã ASCII 7 bit là một hạn chế lớn vì dữ liệu đa phương tiện nhị phân phải được chuyển sang mã ASCII trước khi được gửi đi qua SMTP và sau đó lại phải giải mã thành mã nhị phân sau khi thư đến đích. Trong mục 2.3 ta đã biết rằng HTTP không yêu cầu dữ liệu đa phương tiện phải mã hoá sang mã ASCII trước khi truyền.

Để minh hoạ hoạt động cơ bản của SMTP, hãy xét ví dụ sau: giả sử Alice muốn gửi cho Bob một thông điệp ASCII đơn giản:

Đầu tiên, Alice sử dụng user agent của mình, đánh địa chỉ e-mail của Bob (bob@somechool.edu), soạn e-mail và yêu cầu user agent gửi thư đi.

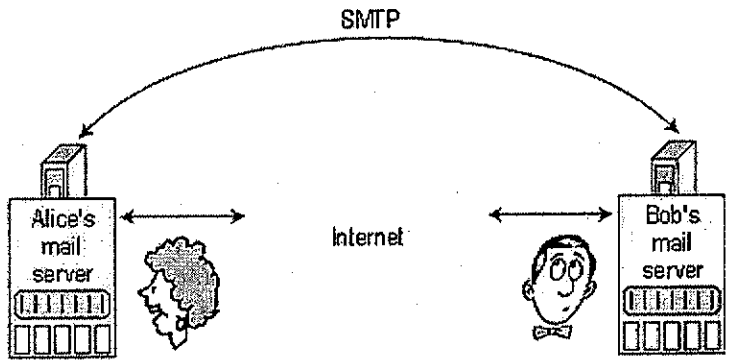
User agent của Alice gửi thư tới mail server của Alice. Tại đây thư được đặt vào hàng thư đợi gửi.

SMTP client chạy trên mail server của Alice thấy thư trong hàng đợi. Nó tạo kết nối TCP tới SMTP server trên mail server của Bob.

Sau giai đoạn khởi tạo 3 bước, SMTP client gửi thư của Alice qua kết nối TCP.

Tại mail server của Bob, SMTP server nhận thư và đặt thư vào mail box của Bob.

Cuối cùng, khi thuận tiện Bob sẽ sử dụng user agent của mình để đọc thư.



Hình 2.17 Thư được gửi từ mail server của Alice đến mail server của Bob

Kịch bản này được minh hoạ trên Hình 2.17. Một điểm quan trọng cần chú ý là SMTP không sử dụng mail server trung gian để gửi thư - ngay cả khi mail server gửi và nhận ở xa nhau. Ví dụ nếu mail server của Alice đặt ở Hồng Kông và mail server của Bob ở Mobile tiểu bang Alabama, thì giữa hai mail server ở Hồng Kông và Mobile vẫn có đường kết nối TCP trực tiếp. Đặc biệt nếu mail server của Bob bị hỏng thì thư vẫn còn trong mail server của Alice và đợi cho lần gửi sau. Thông điệp không được gửi qua mail server trung gian.

Bây giờ chúng ta có thể xem chi tiết cách thức các mail server gửi thư bằng SMTP. Chúng ta sẽ thấy rằng SMTP có nhiều đặc điểm tương tự như những quy tắc trong giao tiếp trực diện của con người. Đầu tiên, SMTP client (chạy trên mail server gửi) thiết lập kết nối TCP với cổng 25 tại SMTP server (chạy trên mail server nhận). Trong trường hợp server không làm việc, client sẽ cố gắng thử lại lần sau. Ngay khi kết nối được thiết lập, server và client thực hiện một vài thủ tục bắt tay. Quá trình này tương tự như hai người tự giới thiệu về bản thân trước khi tiến hành nói chuyện. Trong thủ tục trao đổi, SMTP client thông báo với SMTP server địa chỉ e-mail người gửi và địa chỉ email người nhận. Ngay sau quá trình giới thiệu, client sẽ gửi thư bằng dịch vụ truyền dữ liệu tin cậy của TCP. Sau đó, client sẽ lặp lại các bước này khi vẫn còn thông điệp khác để gửi tới server, còn nếu không, client yêu cầu TCP đóng kết nối lại.

Ví dụ sau là đoạn "hội thoại" giữa client (C) và server (S). Tên máy tính client là crepes.fr và server là hamburger.edu. Dòng hội thoại mở đầu bằng chữ C: là đoạn hội thoại client gửi qua socket TCP và dòng hội thoại bắt đầu với chữ S: là đoạn hội thoại server gửi đi thông qua socket TCP. Đoạn hội thoại bắt đầu ngay sau khi thiết lập được kết nối TCP:

S: 220 hambuger.edu
C: HELO crepes.fr
S: 250 Hello crepes. fr, pleased to meet you
C: MAIL FROM: <alice@crepes. fr>
S: 250 alice@crepes. fr... Sender ok
C: RCPT TO: <bob@hamburger. edu>
S: 250 bob@hamburger. edu ...Recipient ok
C: DATA
S: 354 Enter mail, end with "." On a line by itself
C: Do you like ketchup?
C: How about pickles ?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hambuger. edu closing connection.

Trong ví dụ trên, client gửi một thông điệp ("Do you like ketchup? How about pickles?") từ mail server crepes.fr tới mail server hamburger.edu. Client sử dụng 5 câu lệnh: HELO (viết tắt của HELLO), MAIL FROM, RCPT TO, DATA và QUIT. Ý nghĩa của những câu lệnh này có thể đoán được qua tên gọi của nó. Server gửi trả kết quả thực hiện mỗi lệnh, kết quả này chứa một mã trạng thái và một lời giải thích tiếng Anh. Ở đây SMTP sử dụng kết nối liên tục: Nếu có nhiều thư để gửi tới cùng một mail server thì mail server gửi sẽ gửi tất cả các thư trên cùng một kết nối TCP. Với mỗi thông điệp, client bắt đầu tiến trình gửi bằng lệnh HELO crepes.fr và chỉ gửi lệnh QUIT sau khi gửi tất cả thư.

Độc giả nên sử dụng Telnet để xem một đoạn hội thoại trực tiếp với SMTP server. Sử dụng telnet serverName 25 trong đó serverName là tên mail server. Khi đó bạn đã thiết lập kết nối TCP giữa máy tính của bạn và mail server. Sau khi đánh lệnh này, bạn sẽ nhận được ngay lập tức mã trả lời 220 từ server. Sau đó hãy sử dụng các lệnh SMTP: HELO, MAIL FROM, RCPT TO, DATA, QUIT ở những thời điểm tương ứng. Nếu bạn telnet qua SMTP server của ai đó, bạn có thể gửi tới họ theo cách này (không phải dùng user agent).

2.4.2 So sánh SMTP với HTTP

Chúng ta hãy cùng so sánh vắn tắt hai giao thức SMTP và HTTP. Cả hai giao thức đều được sử dụng để gửi file giữa các máy tính. HTTP chuyển file hoặc đối tượng từ Web server tới Web client (trình duyệt Web), SMTP chuyển file (là thông điệp thư điện tử) giữa các mail server. Khi truyền file cả 2 giao thức HTTP và SMTP cùng sử dụng kết nối liên tục. Điểm khác biệt cơ bản giữa hai giao thức là HTTP là giao thức kiểu kéo (Pull protocol) – client "kéo" thông tin từ server về. Phía nhận (client) là phía thiết lập kết nối TCP. SMTP lại là giao thức theo kiểu đẩy (Push protocol) – client "đẩy" thông tin lên server. Phía gửi (client) là phía thiết lập kết nối TCP trước.

Ngoài dữ liệu văn bản, thông điệp còn có thể chứa các kiểu dữ liệu khác như âm thanh, hình ảnh. HTTP đặt các đối tượng này trong các thông điệp riêng rẽ để gửi. Với SMTP tất cả các đối tượng này được đặt trong cùng một thư điện tử.

2.4.3 Khuôn dạng thư và chuẩn MIME

Khi Alice gửi thư cho Bob, Alice sẽ đặt thư vào phong bì, ghi rõ địa chỉ gửi và địa chỉ nhận, nhân viên bưu điện sẽ đóng dấu ngày tháng vào phong bì. Thư điện tử cũng giống như vậy, bên cạnh nội dung bức thư (phần thân) cũng cần có địa chỉ người gửi, địa chỉ người nhận. Những thông tin phụ trợ này sẽ được đặt trong các dòng tiêu đề. Các dòng tiêu đề và phần thân của thư được tách biệt với nhau bằng cặp ký tự CR-LF. RFC 822 đặc tả đầy đủ các dòng tiêu đề cũng như ý nghĩa của chúng. Giống HTTP, tiêu đề gồm từ khoá, theo sau là dấu hai chấm (":") và một giá trị nào đó. Với SMTP có một số trường bắt buộc, một số trường không bắt buộc. Tiêu đề phải có trường **From:** và trường **To:** Một số trường như **Subject:** có thể có hoặc không. Lưu ý rằng những trường này khác những lệnh SMTP mà chúng ta đã đề cập trong mục 2.4.1 (mặc dù chúng cũng có "From" và "To"). Các lệnh là một phần trong giai đoạn khởi tạo của SMTP trong khi các trường nằm ngay trong thư.

Một bức thư thường có tiêu đề như sau:

From: alice@crepes. fr
To: bob@hamburger. edu
Subject: Searching for the meaning of life

Sau phần tiêu đề thông điệp là một dòng trống, tiếp đến là thân thông điệp (dạng mã ASCII). Như đã nói ở trên, thông điệp kết thúc bằng một dòng chỉ chứa một dấu chấm câu. Bạn nên sử dụng Telnet để gửi tới mail server một thông điệp có chứa một vài dòng tiêu đề, bao gồm dòng tiêu đề **Subject:** Có thể thử điều này bằng cách telnet vào một mail server: telnet serverName 25 trong đó serverName là tên (hoặc địa chỉ IP) của máy tính.

Mở rộng MIME cho dữ liệu không thuộc dạng ASCII

Phần tiêu đề thông điệp 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, 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 Internet Mail 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. Ví dụ, nếu chỉ ra thân thông điệp chứa ảnh JPEG, user agent nhận có thể gửi thân thông điệp tới chương trình giải nén JPEG. Để gửi thông điệp văn bản không mã hoá theo bảng mã ASCII (ví dụ văn bản tiếng Trung Quốc, Nhật Bản), người ta phải mã hoá nó theo bảng mã ASCII để không làm ảnh hưởng tới SMTP. Trường **Content-Transfer-Encoding**: xác định phần thân thông điệp đã được mã hoá theo bảng mã ASCII và phương pháp mã hoá được sử dụng. Vì vậy khi user agent nhận được một thông điệp với hai tiêu đề trên, đầu tiên nó sử dụng giá trị của tiêu đề **Content-Transfer-Encoding**: để chuyển đổi thân thông điệp về dạng ban đầu (không theo định dạng ASCII) và sau đó sử dụng trường **Content-Type** để xác định thao tác thực hiện kế tiếp.

Xét ví dụ sau, giả sử Alice muốn gửi một ảnh JPEG cho Bob. Để thực hiện điều này, Alice sử dụng phần mềm Eudora, đánh địa chỉ email của Bob, chủ đề của e-mail và chèn ảnh JPEG vào thân thông điệp. Sau khi hoàn tất việc soạn thảo, Alice nhấn nút "Send". Sau đó, user agent của Alice tạo ra một thông điệp MIME có nội dung sau:

```
From: alice@crepes. fr
To: bob@hamburger. edu
Subject: Picture of yummy crepe.
MIME-Version: 1. 0
Content-Transfer-Encoding: base64
Content-Type: Image/jpeg

(base64 encoded data .....
.....
.... base64 encoded data )
```

Với thông điệp MIME trên, chúng ta thấy rằng user agent của Alice mã hoá ảnh JPEG sử dụng kỹ thuật mã hoá base64. Đây là một trong những kỹ thuật mã hoá chuẩn trong MIME [RFC 2045] để biến đổi sang định dạng mã ASCII 7 bit.

Khi Bob đọc thư, user agent của Bob xử lý thông điệp MIME này. Thấy trường **Content-Transfer-Encoding: base64**, nó thực hiện giải mã thân thông điệp đã được mã hoá bằng kỹ thuật base64. Trường **Content-Type: image/jpeg** giúp cho user agent của Bob xác định rằng thân của thông điệp phải được giải nén theo chuẩn JPEG. Cuối cùng, thông điệp chứa trường **MIME-Version:** xác định phiên bản MIME đang được sử dụng. Lưu ý rằng, thông điệp cũng phải tuân theo khuyến nghị RFC 822/SMTP.

Theo đặc tả MIME trong khuyến nghị RFC 2046, trường **Content-Type:** có khuôn dạng sau:

Content-Type: type/subtype; parmeters

Phần tham số (parameters) đi sau dấu chấm phẩy có thể không bắt buộc. Trong khuyến nghị RFC 2046, trường Content-Type được sử dụng để xác định kiểu dữ liệu trong phần thân của thông điệp MIME, gồm hai giá trị: kiểu dữ liệu và kiểu con. Sau phần kiểu và kiểu con là phần tham số. Nói chung, kiểu cao nhất (top-level) 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. Tập hợp tham số phụ thuộc vào kiểu và kiểu con.

Được thiết kế để có thể mở rộng, số lượng các cặp type/subtype và những tham số đi kèm trong MIME ngày càng tăng. Để bảo đảm là tập hợp này phát triển có trình tự, được đặc tả rõ ràng, MIME cần thiết lập quá trình đăng ký với 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.

Hiện nay, mới có định nghĩa cho bảy nhóm dữ liệu chính. Với mỗi kiểu lại có một danh sách các kiểu con và danh sách này đang tăng lên hàng năm. Dưới đây là 5 nhóm dữ liệu chính:

Văn bản (Text): Kiểu văn bản được sử dụng để xác định thân thông điệp chứa thông tin dạng văn bản. Một kiểu con thường gặp là plain (trơn). Văn bản trơn không có lệnh hay chỉ dẫn định dạng khuôn dạng và do đó không cần phần mềm đặc biệt nào để hiển thị. Nếu nhìn tiêu đề MIME của thư trong hộp thư, bạn có thể sẽ thấy trên tiêu đề có trường **text/plain;**

charset="us-ascii" hay **text/plain; charset="ISO-8859-1"**. Những tham số này xác định bộ mã mà thông điệp sử dụng. Một kiểu con khác cũng rất thông dụng là text/html. Kiểu con *html* yêu cầu mail server thông dịch những thẻ HTML gắn trong thông điệp. Điều này cho phép user agent nhận hiển thị thông điệp dưới dạng một trang Web (với font, hyperlink, applet).

Ảnh (Image): Kiểu ảnh được dùng để xác định thân thông điệp là ảnh. Hai kiểu con thông dụng là image/gif và image/jpeg. Với kiểu con image/gif, muốn hiển thị ảnh, user agent phải giải nén ảnh 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 μ 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. Nó thường được áp dụng cho loại dữ liệu phải qua một ứng dụng khác xử lý trước khi người nhận có thể sử dụng được. Ví dụ khi người gửi gắn một tài liệu MS Word vào thông điệp E-mail, user agent đặt giá trị application/msword vào trường type/subtype. Khi user agent thấy giá trị application/msword trong trường type/subtype, nó khởi động ứng dụng MS Winword và chuyển phần thân thông điệp MIME cho ứng dụng Word. Một kiểu con quan trọng khác là octerstream. Kiểu con này thường được dùng khi thân thông điệp chứa dữ liệu nhị phân tuỳ ý. Khi nhận kiểu con này, mail reader sẽ yêu cầu người nhận lựa chọn để lưu thông điệp trên đĩa xử lý sau.

Có một kiểu MIME đặc biệt quan trọng là kiểu **multipart**. Thông điệp e-mail cũng như trang Web có thể chứa nhiều đối tượng (như văn bản, ảnh, applet). Web gửi mỗi đối tượng trong một thông điệp trả lời độc lập nhưng thư điện tử đặt tất cả các đối tượng trong cùng một thông điệp. Đặc biệt, khi thông điệp đa phương tiện có nhiều đối tượng thì thông điệp đó có kiểu là **multipart/mixed**. Khi nhận được một thông điệp mà trường **content-type** có giá trị **multipart/mixed,** user agent nơi nhận biết thông điệp nhận được chứa nhiều đối tượng. Khi nhận được thông điệp như vậy, user agent phải xác định rõ:

Điểm đầu và điểm cuối của đối tượng.

Cách mã hoá các đối tượng không theo bảng mã ASCII.

Kiểu của mỗi đối tượng.

Công việc này được thực hiện nhờ ký tự phân cách giữa các đối tượng và trường **Content-type, Content-Transfer-Encoding** đứng trước mỗi đối tượng trong thông điệp. Xét ví dụ sau: Giả sử Alice muốn gửi thông điệp bao gồm một đoạn văn bản ASCII, một ảnh JPEG và cuối cùng là một đoạn văn bản ASCII cho Bob. Sử dụng user agent của mình, Alice đánh một đoạn văn bản, chèn ảnh JPEG sau đó đánh tiếp đoạn văn bản còn lại. Kết quả là user agent của Alice tạo ra một thông điệp như sau:

```
From: alice@crepes. fr
To: bob@hamburger. edu
Subject: Picture of yummy crepe with commentary
MIME-Version: 1. 0
Content-Type: multipart/mixed; Boundary=StartOfNextPart
--StartOfNextPart
Dear Bob,
Please find a picture of an absolutely scrumption crepe.
--StartOfNextPart
Content-Transfer-Encoding: base64
Content-Type: image/jpeg
base64 encoded data ....
.....
..... base64 encoded data
--StartOfNextPart
Let me know if you would like the recipe.
```

Qua thông điệp trên chúng ta thấy rằng trường **Content-Type**: trong tiêu đề xác định cách thức phân cách các phần khác nhau trong cùng một thông điệp. Việc phân cách được bắt đầu bằng 2 dấu gạch ngang (--) và kết thúc bằng cặp ký tự CRLF.

Nhận thông điệp

Thông điệp e-mail bao gồm nhiều phần, lõi của thông điệp là phần thân chứa dữ liệu thực sự được chuyển từ người gửi đến người nhận. Với

thông điệp nhiều phần, thân thông điệp gồm nhiều phần và trước mỗi phần có một hoặc vài trường xác định kiểu. Đứng trước thân thông điệp là cặp CRLF và một số trường. Những trường như **From:**, **To:**, và **Subject:** được đặc tả trong RFC 822 và những trường như **Content-type:** và **Content-Transfer-Encoding:** là tiêu đề MIME. Nhưng chính mail server nhận thông điệp cũng chèn vào thông điệp một số trường khác, ví dụ **Received:** ở đầu thông điệp. Trường này xác định tên của SMTP server gửi thông điệp ("from"), tên của SMTP server nhận thông điệp ("by") và thời gian khi thông điệp tới đích. Người đọc sẽ đọc được thông điệp như sau:

Received: from crepes. fr by hamburger. edu; 12 Oct 98
15:27:39 GMT
From: alice@crepes. fr
To: Bob@hamburger. edu
Subject: Picture of yummy crepe.
MIME-Version: 1. 0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.....
..... base64 encoded data

Thực ra tất cả mọi người khi dùng e-mail đều nhìn thấy trường **Received:** đứng trước thông điệp e-mail (trường này có thể nhìn thấy trực tiếp trên màn hình hoặc khi in thư). Có thể có những thông điệp có nhiều trường **Received:** và trường **Return-Path**: phức tạp. Đó là vì thông điệp này có thể được chuyển tiếp (forward) qua nhiều mail server trước khi đến tay người nhận. Ví dụ nếu Bob cấu hình mail server của mình (hamburger.edu) gửi chuyển tiếp tất cả các thư của Bob tới sush.jp, khi đó tất cả các thư của Bob khi lấy từ sush.jp sẽ có những trường sau:

Received: from hamburger. edu by sushi. jp; Oct 98 15:30:01 GMT

Received: from crepes. fr by hamburger. edu; 12 Oct 98 15:27:39 GMT

Những trường này cho phép người nhận theo dõi vết đường đi của thư qua nhiều SMTP server cũng như thời gian thư tới mỗi server.

2.4.4 Giao thức truy nhập mail

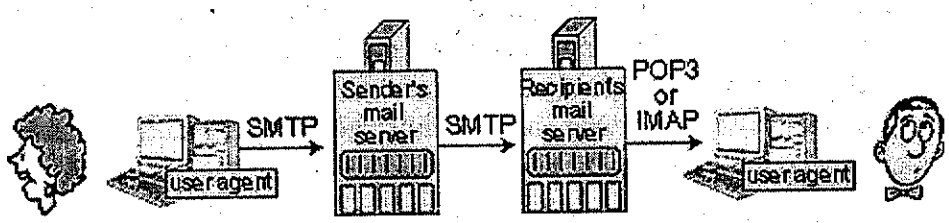
Mỗi khi SMTP gửi thư từ mail server của Alice tới mail server của Bob, thư được đặt trong mail box của Bob. Từ trước tới giờ, chúng ta chấp nhận giả thiết để đọc thư, Bob phải đăng nhập vào mail server và sử dụng một chương trình đọc thư (mail reader) nào đó cài ngay trên mail server. Tới tận đầu những năm 90, mọi người vẫn thực hiện như vậy. Nhưng ngày nay mọi người thường đọc thư qua một user agent chạy trên máy tính cá nhân của mình. Chạy user agent trên máy tính cá nhân, người sử dụng có được nhiều tính năng cao cấp, kể cả việc gửi và nhận những thông điệp đa phương tiện.

Giả sử Bob (người nhận) chạy user agent của mình trên máy tính cá nhân. Có thể cài đặt mail server ngay trên máy tính cá nhân của Bob. Tuy nhiên cách này có nhiều nhược điểm. Mail server quản lý nhiều mailbox, thực hiện cả chức năng client và server của SMTP. Nếu cài mail server trên máy tính cá nhân của Bob thì PC đó lúc nào cũng phải bật và kết nối vào Internet để có thể nhận thư mới (mà thư thì có thể đến bất cứ lúc nào). Điều này không thực tế với đa số người sử dụng Internet. Thông thường, người sử dụng chạy chương trình user agent trên máy tính cá nhân, truy cập vào hộp thư trên một mail server dùng chung (mail server này luôn luôn kết nối tới Internet và được chia sẻ giữa nhiều người dùng khác). Mail server thường được ISP của người dùng (là trường đại học hoặc công ty) quản lý.

Do user agent chạy trên máy tính cá nhân và mail server được quản lý bởi các ISP nên cần có một giao thức cho phép user agent và mail server trao đổi với nhau. Đầu tiên chúng ta xét trường hợp thư được tạo ra tại PC của Alice được chuyển tới mail server của Bob như thế nào. Công việc này có thể được thực hiện một cách đơn giản bằng việc user agent của Alice trao đổi trực tiếp với mail server của Bob bằng giao thức SMTP. User agent của Alice sẽ khởi tạo một kết nối TCP tới mail server của Bob, gửi những lệnh khởi tạo SMTP, tải thư lên bằng lệnh DATA và sau đó đóng kết nối lại. Cách tiếp cận này hoàn toàn có thể thực hiện được nhưng ít khi được dùng vì nó không hỗ trợ trường hợp mail server phía nhận bị trục trặc. Trên thực tế, user agent gửi khởi tạo SMTP để tải thư của Alice tới chính mail server

của Alice (chứ không phải là mail server của người nhận thư). Mail server của Alice sau đó sẽ thiết lập một phiên làm việc SMTP tới mail server của Bob để gửi tiếp thư tới mail server của Bob. Nếu mail server của Bob ngừng làm việc thì mail server của Alice sẽ giữ thư lại và sau đó cố gắng gửi lại. RFC SMTP có những lệnh để gửi tiếp thư qua nhiều SMTP server.

Vậy user agent chạy trên máy tính cá nhân của Bob lấy thông điệp trong hộp thư trên mail server của Bob như thế nào ? Giải pháp là phải có một giao thức lấy thư cho phép chuyển thư từ mail server của Bob tới máy tính cục bộ. Hiện nay có 2 giao thức lấy thư thông dụng là **POP3 (Post Office Protocol - Version 3)** và **IMAP (Internet Mail Access Protocol)**. Dưới đây, chúng ta sẽ trình bày cả hai giao thức này. Lưu ý rằng user agent của Bob không thể sử dụng SMTP để lấy thư bởi vì lấy thư giống như việc "kéo" trong khi SMTP là một giao thức "đẩy". Hình 2.18 minh hoạ về việc gửi và nhận thư. SMTP được dùng để chuyển thư giữa các mail server hay giữa user agent của người gửi và mail server của người gửi. POP3 hay IMAP được dùng để chuyển thư từ mail server tới user agent của người nhận.



Hình 2.18 Giao thức email và các thực thể truyền thông

POP3

POP3 được đặc tả trong RFC 1939 là giao thức lấy thư cực kỳ đơn giản và có rất ít chức năng. POP3 được khởi tạo khi user agent (client) tạo kết nối TCP tới mail server (server) qua cổng 110. Sau khi thiết lập được kết nối TCP, POP3 gồm 3 giai đoạn: kiểm chứng, tiến hành xử lý và cập nhật. Trong giai đoạn kiểm chứng đầu tiên, user agent sử dụng tên và mật khẩu để xác nhận người sử dụng. Trong giai đoạn tiến hành xử lý thứ hai, user agent tiến hành lấy thư. Nó có thể đánh dấu các thư để xoá hay hủy bỏ việc đánh

dấu xoá. Giai đoạn ba - cập nhật, xảy ra sau khi client ra lệnh quit để kết thúc phiên làm việc POP3. Tại thời điểm đó mail server xoá tất cả thư được đánh dấu.

Trong giai đoạn xử lý, user agent gửi lệnh và server trả lời kết quả thực hiện của mỗi lệnh đó. 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.

Giai đoạn kiểm chứng có 2 lệnh là **user <username>** và **pass <password>**. Để minh hoạ hai lệnh này bạn nên Telnet trực tiếp qua một server POP3 sử dụng cổng 110 để thực hành. Giả sử mail server là tên mail server, bạn có thể làm như sau:

```
telnet mail server 110
+OK POP3 server ready
user alice
+OK
pass hungry
+OK user successfully logged on
```

Nếu bạn đánh sai một lệnh thì server POP3 sẽ đáp lại bằng một thông điệp –ERR.

Người sử dụng có thể cấu hình user agent ở một trong hai chế độ "tải và xoá" ("download and delete") hay "tải và giữ" ("download và keep"). Chuỗi lệnh được user agent gửi phụ thuộc vào cấu hình này. Trong chế độ đầu, user agent sẽ phát ra chuỗi lệnh **list, retr** và **dele**. Giả sử người dùng có 2 thông điệp trong hộp thư của mình. Trong đoạn hội thoại dưới đây C: (client) là user agent và S: (server) là mail server. Khi đó giai đoạn xử lý công việc sẽ như sau:

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: (blah blah ...
```

S: .....
S: ..... blah)
S:.
C: dele 1
C: retr 2
S: (blah blah ....
S: .....
S: .....blah)
S:.
C: dele 2
C: quit
S: +OK POP3 server signing off

Đầu tiên, user agent yêu cầu mail server liệt kê kích thước của tất cả thư lưu trữ trong hộp thư. Sau đó, user agent lấy và xoá từng thư trong hộp thư. Lưu ý rằng sau giai đoạn kiểm chứng người dùng chỉ còn 4 câu lệnh là **list, retr, dele** và **quit**. Cú pháp của các lệnh này được đặc tả trong RFC 1939. Sau khi xử lý lệnh quit, server POP3 vào giai đoạn cập nhật và xoá thư 1, 2 trong mailbox.

Trong chế độ này, khi Bob lấy thư từ những địa điểm khác nhau, thư của Bob sẽ nằm rải rác trên nhiều máy. Đặc biệt, nếu Bob đã lấy thư từ máy tính ở nhà thì sau đó sẽ không thể đọc lại thư đó trên máy tính ở cơ quan. Trong chế độ thứ hai "download and keep", user agent vẫn để lại thư trên mail server sau khi đã tải về. Khi đó Bob vẫn có thể đọc thư từ nhiều máy khác nhau.

Trong phiên làm việc POP3 giữa user agent và mail server, server POP3 sẽ ghi nhớ một vài thông tin trạng thái – ví dụ các thư đã bị đánh dấu xoá. Tuy nhiên server POP3 không chuyển thông tin trạng thái giữa các phiên làm việc khác nhau. Ví dụ không có thư nào được đánh dấu xoá ở đầu mỗi phiên làm việc. Điều này làm đơn giản công việc xây dựng một server POP3.

IMAP

Sau khi tải thư về từ máy tính cá nhân, Bob có thể tạo những thư mục chứa thư và chuyển thư vào trong các thư mục đó. Sau đó Bob có thể

xoá, chuyển thư giữa các thư mục hay tìm kiếm thư theo tên người gửi và chủ đề thư. Phương thức như vậy bất tiện với người sử dụng muốn đọc thư từ nhiều nơi vì họ thích duy trì phân cấp thư mục trên mail server để có thể truy cập được từ bất kỳ máy tính nào. POP3 không đáp ứng được yêu cầu này.

IMAP (đặc tả trong RFC 2060) có thể giải quyết vấn đề này. Giống POP3, IMAP cũng là giao thức lấy thư. Nó có nhiều đặc tính phức tạp hơn POP3. IMAP được thiết kế cho phép người dùng thao tác trên những hộp thư ở xa một cách dễ dàng. IMAP cho phép Bob tạo những thư mục thư khác nhau trong mailbox. Bob 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 server IMAP phải duy trì hệ thống thư mục cho mọi người dùng. Những thông tin trạng thái như thế phải được mail server lưu giữ cho tất cả các phiên làm việc. Nên nhớ rằng POP3 server không lưu giữ trạng thái của mỗi người dùng sau khi phiên làm việc kết thúc.

IMAP có một đặc tính quan trọng là có những lệnh cho phép user agent chỉ lấy một số thành phần trong thư. Ví dụ: user agent có thể lấy phần tiêu đề hoặc một phần trong thư có nhiều phần. Điều này rất có ích khi kết nối giữa useragent và mail server chậm, người dùng có thể không cần tải tất cả thư trong hộp thư của mình. Đặc biệt có thể tránh tải những thư chứa nội dung âm thanh hay hình ảnh có kích thước lớn.

Phiên làm việc IMAP gồm 3 giai đoạn: giai đoạn thiết lập kết nối giữa client (user agent) và IMAP server, giai đoạn server chấp nhận kết nối và giai đoạn tương tác client/server. Tương tác client/server của IMAP tương tự nhưng phong phú hơn nhiều tương tác trong POP3. Server luôn ở một trong bốn trạng thái. Trong trạng thái **chưa kiểm chứng (nonauthenticated)** là trạng thái khởi đầu, khi đó người dùng phải đăng nhập hệ thống trước khi thực hiện các lệnh. Trạng thái **đã kiểm chứng (authenticated)**, người dùng phải chọn một thư mục trước khi gửi lệnh. Trong trạng thái **lựa chọn (selected)**, người dùng có thể sử dụng những lệnh có thể tác động tới thông điệp (như lấy, xoá, chuyển thư). Cuối cùng là trạng thái **thoát (logout)**, khi kết thúc phiên làm việc. Bạn có thể đọc tất cả về IMAP tại [IMAP 1999].

HTTP

Ngày nay, nhiều người sử dụng Webmail - dịch vụ có thể truy cập email qua trình duyệt (Hotmail hay Yahoomail). Khi đó, user agent là trình duyệt Web thông thường và mọi người kết nối tới hộp thư của mình trên mail server qua HTTP. Khi Bob muốn đọc thư, thư được gửi từ mail server của Bob tới trình duyệt nhờ giao thức HTTP chứ không phải là giao thức POP3 hay IMAP. Khi người gửi (Alice) có một tài khoản (account) trên mail server muốn gửi thư, thư sẽ được gửi từ trình duyệt của Alice tới mail server nhờ giao thức HTTP chứ không sử dụng SMTP. Tuy nhiên, mail server đó vẫn gửi và nhận thư với những mail server khác bằng giao thức SMTP. Giải pháp truy cập mail kiểu như vậy vô cùng thuận tiện cho người sử dụng hay phải di chuyển. Họ chỉ cần sử dụng một trình duyệt để gửi và nhận thư. Trình duyệt đó có thể ở một quán café Internet, ở nhà của bạn bè, ở khách sạn với Web TV... Giống IMAP người dùng có thể tổ chức thư của mình trong hệ thống thư mục trên mail server. Sự thật là, e-mail dựa trên Web ngày càng thuận tiện và dần dần thay thế POP3 và IMAP trong những năm tới. Nhược điểm chính yếu của nó là chậm, bởi vì server và client ở xa, giao tiếp giữa chúng phải thông qua CGI.

2.5 DỊCH VỤ TÊN MIỀN - DNS

Cá nhân mỗi con người có thể được xác định theo nhiều cách. Ví dụ, chúng ta có thể được nhận biết qua tên trong giấy khai sinh, bằng số chứng minh thư nhân dân. Dù có nhiều cách nhận biết để phân biệt mọi người nhưng lựa chọn phương thức nhận biết phụ thuộc vào hoàn cảnh. Ví dụ công an sử dụng số chứng minh thư nhân dân chứ không sử dụng tên. Bình thường mọi người thích nhớ tên nhau hơn là số chứng minh thư.

Giống con người, máy tính trên Internet cũng có thể được xác định bằng nhiều cách. Tên máy tính (host name) là một cách, ví dụ cnn.com, www.yahoo.com, gais.umass.edu hay surf.eurecom.fr. Những tên đó tương đối dễ nhớ đối với con người. Tuy nhiên tên máy tính cung cấp ít thông tin về vị trí trên Internet của máy tính (Tên máy tính surf.eurecom.fr chỉ cho

chúng ta biết máy tính đó ở nước Pháp vì kết thúc bằng đuôi .fr, ngoài ra không còn thông tin nào khác). Hơn nữa tên máy tính bao gồm nhiều ký tự - cả chữ cái và chữ số - có độ dài thay đổi nên router khó có thể xử lý được. Vì lý do đó, máy tính được xác định thông qua địa chỉ IP. Địa chỉ IP được thảo luận chi tiết trong chương 4. Địa chỉ IP gồm có 4 byte và có cấu trúc phân cấp, giá trị mỗi byte lại được đổi ra số thập phân (0-255) và cách nhau bằng dấu chấm (ví dụ 121.23.45.5). Địa chỉ IP phân cấp vì khi duyệt địa chỉ từ trái qua phải, chúng ta nhận được thêm nhiều thông tin xác định về vị trí của máy tính trên Internet (Vị trí ở trong mạng của các mạng, trong một mạng...). Điều này tương tự khi xét địa chỉ bưu điện từ dưới lên, chúng ta nhận được nhiều thông tin về địa chỉ đó.

2.5.1 Các dịch vụ của DNS

Có hai cách để xác định một máy tính: dựa vào tên máy tính hoặc địa chỉ IP. Con người thích sử dụng tên máy dễ nhớ, trong khi router lại sử dụng địa chỉ IP có cấu trúc phân cấp và độ dài cố định (dễ xử lý). Để dung hoà giữa hai cách, chúng ta cần một dịch vụ chỉ dẫn để chuyển đổi tên máy tính sang địa chỉ IP và đây chính là nhiệm vụ của hệ thống tên miền trên Internet (DNS). DNS là (1) Cơ sở dữ liệu phân tán được đặt trên một hệ thống phân cấp các máy phục vụ tên (nameserver) và (2) Giao thức thuộc tầng ứng dụng cho phép máy tính và máy chủ tên trao đổi thông tin phục vụ mục đích xác định địa chỉ IP. Máy chủ tên miền thường là các máy UNIX c cài đặt phần mềm Berkely Internet Name Domain (BIND). Giao thức DNS chạy trên nền UDP với số hiệu cổng là 53.

Thông thường DNS được các giao thức tầng ứng dụng khác như HTTP, SMTP và FTP sử dụng để xác định địa chỉ IP từ tên máy tính do người dùng đưa vào. Chuyện gì xảy ra khi trình duyệt (HTTP client) trên máy tính của người sử dụng yêu cầu đối tượng có địa chỉ URL là www.someschool.edu/index.html. Để gửi được thông điệp HTTP yêu cầu tới Web server thì máy tính của người sử dụng phải xác định được địa chỉ IP của www.someschool.edu. Điều này được thực hiện như sau: máy tính của người sử dụng chạy phía client của ứng dụng DNS. Trình duyệt sẽ lấy ra tên

máy tính (www.someschool.net) từ địa chỉ URL và chuyển nó cho phần mềm client của DNS. DNS client gửi một truy vấn (query) chứa tên máy tính tới DNS server. DNS client sẽ nhận được một thông điệp trả lời từ DNS server chứa địa chỉ IP cần xác định. Sau đó trình duyệt sẽ mở một kết nối TCP tới tiến trình HTTP server trên máy tính có địa chỉ IP vừa được xác định.

Rõ ràng các ứng dụng Internet sử dụng DNS hoạt động chậm đi. Tuy nhiên, địa chỉ IP đã được xác định thường được ghi tạm (cache) trong một name server DNS ở gần và như vậy làm giảm tải cho hệ thống DNS cũng như độ trễ của ứng dụng.

Bên cạnh dịch vụ xác định địa chỉ IP từ tên máy, DNS cung cấp một số dịch vụ quan trọng sau:

Dịch vụ đặt bí danh cho máy tính (Host aliasing):

Máy tính có tên phức tạp có thể có một hoặc nhiều bí danh (alias). Ví dụ tên máy tính relay1.west-coast.enterprise.com có thể có hai bí danh là www.enterprise.com và enterprise.com. Trong trường hợp này, relay1.west-coast.enterprise.com là tên đầy đủ (canonical name). Tên bí danh thường dễ nhớ hơn tên đầy đủ. Một ứng dụng có thể yêu cầu DNS xác định tên đầy đủ cũng như địa chỉ IP của một tên bí danh.

Dịch vụ đặt bí danh cho mail server (Mail server aliasing):

Hiển nhiên địa chỉ email cần dễ nhớ. Ví dụ nếu Bob có tài khoản trên Hotmail, địa chỉ của Bob có thể chỉ đơn giản là bob@hotmail.com. Tuy nhiên tên máy tính của máy phục vụ thư tại Hotmail phức tạp và vì thế khó nhớ hơn so với hotmail.com (Ví dụ tên đầy đủ có thể là relay1.west-coast.hotmail.com). Ứng dụng có thể sử dụng DNS để xác định tên đầy đủ của một bí danh cũng như địa chỉ IP của máy tính đó. Trên thực tế, DNS cho phép mail server và Webserver của các công ty có tên (bí danh) giống nhau, ví dụ: Webserver và mail server của một công ty có thể cùng là enterprise.com.

Phân tán tải (load distribution):

DNS thực hiện việc phân tán tải cho các server, đặc biệt là các Web server "nhân bản" (replicated) (là các server có nội dung giống hệt nhau).

Những site có nhiều người truy cập như cnn.com được đặt trên nhiều server giống hệt nhau. Mỗi server là một hệ thống đầu cuối (end system) khác nhau, có địa chỉ IP khác nhau. Đối với các server giống hệt nhau như vậy, một nhóm địa chỉ IP sẽ gắn với tên đầy đủ của một máy nào đó. Cơ sở dữ liệu DNS chứa toàn bộ nhóm địa chỉ IP đó. Khi client gửi truy vấn DNS để xác định địa chỉ IP thì server sẽ gửi toàn bộ nhóm địa chỉ IP đó nhưng server thay đổi thứ tự các địa chỉ IP trong nhóm. Thông thường client gửi thông điệp HTTP tới máy tính có địa chỉ IP được liệt kê đầu tiên trong nhóm. Sự hoán chuyển vị trí các địa chỉ IP mà DNS thực hiện đã phân tải cho các server. Việc hoán vị của DNS cũng được áp dụng cho email khi nhiều mail server có chung bí danh.

DNS được đặc tả trong RFC 1034, RFC 1035 và cập nhật trong một số RFC khác. DNS là hệ thống phức tạp và chúng ta chỉ nghiên cứu một vài khía cạnh của nó.

Trong lịch sử

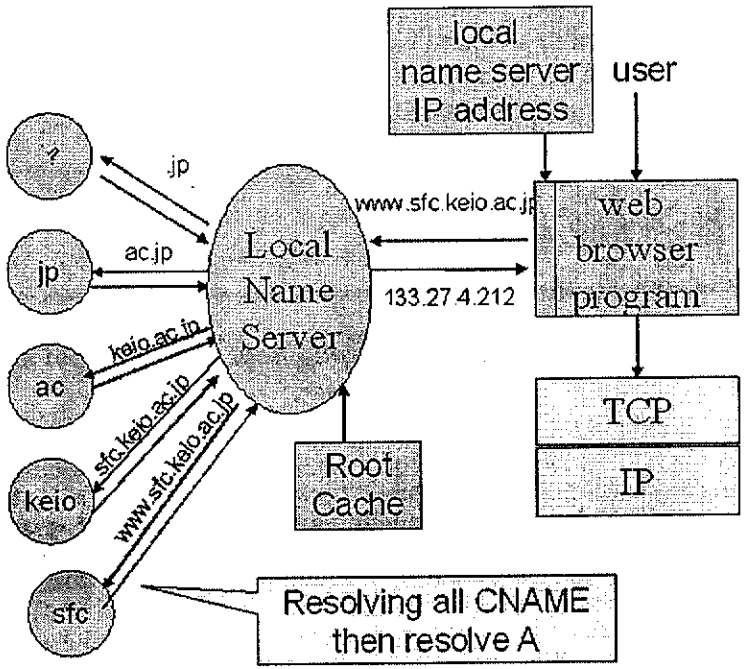
DNS: Một dịch vụ mạng quan trọng hoạt động trên mô hình client/server.

Giống HTTP, FTP hay SMTP, giao thức DNS nằm ở tầng ứng dụng vì (1) nó hoạt động giữa hai thực thể truyền thông đầu cuối sử dụng mô hình client/server và (2) sử dụng một giao thức giao vận end-to-end ở tầng giao vận phía dưới để trao đổi thông điệp DNS giữa hai phía đầu cuối. Tuy nhiên vai trò của DNS khác các ứng dụng Web, FTP hay Email nhiều. DNS không phải ứng dụng được người dùng trực tiếp sử dụng mà DNS chỉ cung cấp một dịch vụ Internet cực kỳ thiết yếu cho các ứng dụng: chuyển đổi tên máy tính sang địa chỉ IP. Trước đây trong mục 1.2 chúng ta thấy rằng sự phức tạp trong kiến trúc của Internet được đặt tại "lớp vỏ" của mạng. DNS được triển khai trên các máy tính đầu cuối là một minh chứng rõ ràng cho nguyên lý thiết kế này.

2.5.2 Cơ chế hoạt động của DNS

Bây giờ, chúng ta trình bày tổng quan cách thức hoạt động của DNS, tập trung vào dịch vụ xác định địa chỉ IP từ tên máy tính. Với client, DNS là

một "hộp đen". Client gửi thông điệp truy vấn DNS vào hộp đen đó, trong thông điệp chứa tên máy cần xác định địa chỉ IP. Với hệ điều hành Unix, gethostname() là một hàm mà ứng dụng có thể gọi để gửi thông điệp truy vấn. Sau một khoảng thời gian nào đó - từ vài phần nghìn giây đến vài chục giây, client nhận được thông điệp trả lời của DNS chứa địa chỉ IP cần xác định. Vì vậy, với client thì DNS là một dịch vụ xác định IP đơn giản và dễ hiểu. Nhưng "hộp đen" triển khai dịch vụ đó thực sự phức tạp, bao gồm nhiều máy chủ tên (nameserver) đặt khắp nơi trên thế giới và một giao thức ở tầng ứng dụng xác định cách thức trao đổi thông tin giữa các nameserver và giữa nameserver với máy tính.



Hình 2.19 Ứng dụng sử dụng dịch vụ của DNS

Để triển khai DNS, người ta có thể đưa ra một kiến trúc đơn giản sau: có một name server chứa tất cả các ánh xạ tên và địa chỉ IP. Theo thiết kế tập trung này, client chỉ cần gửi tất cả các truy vấn tới nameserver duy nhất và nameserver này sẽ trực tiếp trả lời mọi truy vấn. Mặc dù tính đơn giản của thiết kế này rất hấp dẫn nhưng nó hoàn toàn không thích hợp cho

Internet với số lượng lớn và ngày càng nhiều máy tính. Thiết kế tập trung như vậy nảy sinh một số vấn đề sau:

Điểm hỏng duy nhất (A single point of failure) nếu nameserver duy nhất ngừng làm việc cũng có nghĩa là toàn bộ Internet ngừng hoạt động.

Khối lượng công việc (Traffic volume): Một nameserver duy nhất phải xử lý tất cả các truy vấn DNS (cho tất cả các thông điệp yêu cầu từ hàng triệu máy tính trên toàn cầu).

Cơ sở dữ liệu tập trung ở xa (distant centralized database): Nameserver duy nhất không thể gần tất cả các client. Nếu nameserver đặt ở NewYork thì tất cả truy vấn từ Úc phải chuyển tới phía bên kia trái đất và có thể qua một đường kết nối chậm và tắc nghẽn. Hậu quả là các ứng dụng phải chịu độ trễ lớn.

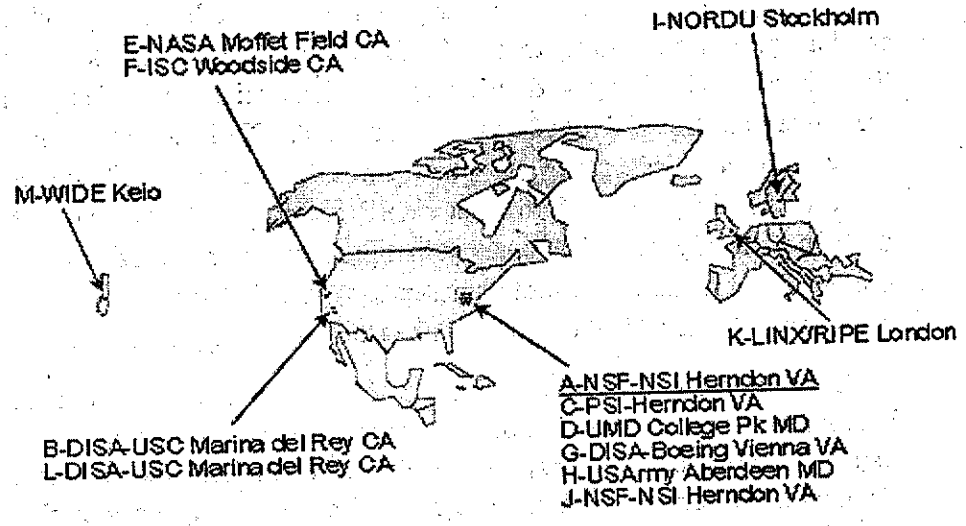
Bảo trì (maintenance): Namserver phải ghi nhớ thông tin về tất cả các máy tính trên Internet. Khi đó cơ sở dữ liệu sẽ cực kỳ lớn và nameserver phải cập nhật thường xuyên thông tin cho mọi máy tính mới cũng như phải giải quyết các vấn đề kiểm chứng và xác nhận khi người dùng sử dụng cơ sở dữ liệu tập trung.

Tóm lại, cơ sở dữ liệu tập trung trên một nameserver duy nhất không phù hợp khi quy mô hệ thống lớn. Do đó, DNS được thiết kế phân tán. Trên thực tế DNS là một ví dụ tuyệt vời về triển khai cơ sở dữ liệu phân tán trên Internet. Để giải quyết vấn đề quy mô, DNS sử dụng nhiều nameserver tổ chức phân cấp và phân tán trên toàn cầu. Không có nameserver nào chứa tất cả tên và địa chỉ IP các máy tính trên Internet, những thông tin này được phân tán trên nhiều nameserver. Có ba loại nameserver: local nameserver, root nameserver và authoritative nameserver. Các nameserver đó trao đổi thông tin với nhau và với các máy tính khác.

Local nameserver: Mỗi ISP như trường đại học, công ty đều có local nameserver (còn được gọi là default name server). Khi máy tính trong cơ quan tạo ra một thông điệp truy vấn DNS thì đầu tiên thông điệp đó được gửi tới local name server của tổ chức. Địa chỉ IP của local name server phải được cấu hình trong máy tính (Trong Win 95/98, bạn có thể tìm thấy địa chỉ IP của local name server bằng cách mở Control Panel, sau đó chọn Network,

chọn TCP/IP, rồi chọn cấu hình DNS). Local name server thường "gần" với client, trong trường hợp tại cơ quan của một tổ chức, nó có thể ở trên cùng mạng LAN với máy tính client. Với ISP phục vụ kết nối từ nhà thì khoảng cách giữa name server và các máy tính client chỉ là vài router. Nếu máy tính yêu cầu xác định địa chỉ IP của một máy tính khác trong cùng một ISP thì local name server có thể ngay lập tức xác định được địa chỉ IP cần thiết. Ví dụ nếu máy tính surf.eurecom.fr yêu cầu địa chỉ IP của baie.eurecom.fr thì local name server ở eurecom ngay lập tức có thể đưa ra địa chỉ IP mà không phải liên hệ với bất kỳ name server nào khác.

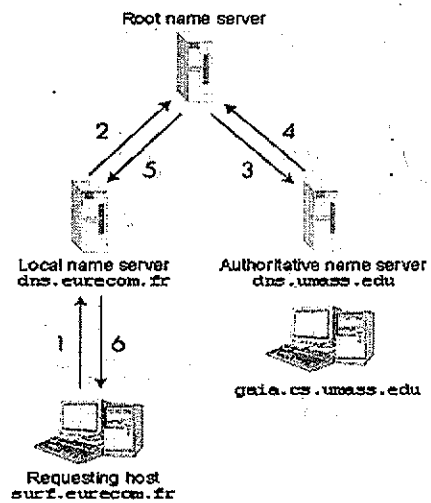
Rootname server: Trên Internet có 13 rootname server, hầu hết đặt tại Bắc Mỹ. Vị trí các root name server vào thời điểm tháng 02/1998 được minh hoạ trên Hình 2.18. Khi local name server không thể trả lời truy vấn của một máy tính (bởi vì nó không có thông tin của máy tính được yêu cầu) thì local name server sẽ đóng vai trò client DNS và gửi câu hỏi truy vấn tới một trong số các root name server. Nếu root name server có thông tin của máy tính được hỏi, nó sẽ gửi một thông điệp DNS hồi âm tới local name server và sau đó thông tin này được local name server gửi trả lời cho máy tính yêu cầu. Nhưng root name server có thể không có thông tin của máy tính đó. Trong trường hợp này, root name server biết được địa chỉ IP của name server quản lý máy tính đó.



Hình 2.20 Các root name server trên thế giới

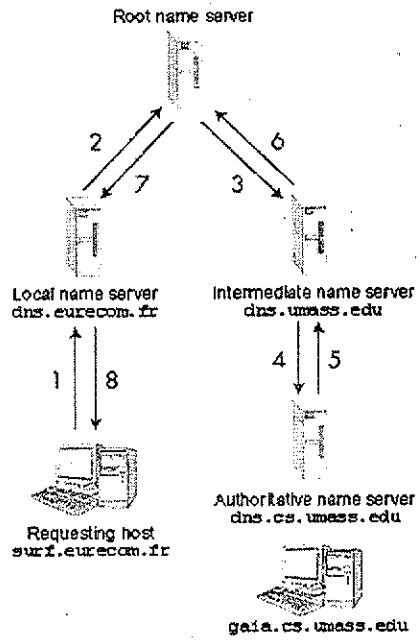
Authoriative name server: Mỗi máy tính phải đăng ký tới một Authoritative name server. Thông thường authoriative name server của một máy tính là name server trong miền ISP của máy tính đó (thực tế mỗi máy tính phải có ít nhất hai authoritative name server, để đề phòng trường hợp một name server bị hỏng). Có thể định nghĩa, Authoriative name server của một máy tính là nameserver luôn lưu trữ bản ghi DNS cho phép xác định địa chỉ IP của máy tính đó từ tên. Khi authoritative name server nhận được truy vấn từ root nameserver, nó sẽ gửi một thông điệp DNS trả lời chứa ánh xạ được yêu cầu. Sau đó, root server gửi ánh xạ đó tới local nameserver và local nameserver lại tiếp tục gửi ánh xạ đó tới máy tính yêu cầu. Nhiều nameserver vừa là local vừa là authoriative nameserver.

Xét ví dụ đơn giản sau. Giả sử máy tính surf.eurecom.fr muốn có địa chỉ IP của máy tính gaia.cs.umas.edu, giả sử nameserver của miền Eurecom là dns.eurecom.com.fr và authoriative nameserver cho gaia.cs.umas.edu là dns.umass.edu. Như đã trình bày trong Hình 2.20, đầu tiên máy tính surf.eurecom.fr gửi một thông điệp truy vấn tới local name server của nó là dns.eurecom.fr. Thông điệp đó chứa tên máy tính cần xác định địa chỉ IP, là gaia.cs.umass.edu. Local name server gửi thông điệp tới root name server. Root nameserver gửi tiếp thông điệp tới nameserver có thể xác định tất cả các máy tính trong miền umass.edu, ví dụ là dns.umass.edu. Sau đó, authoriative name server này gửi kết quả cho surf.eurecom.fr thông qua root name server và local name server. Trong ví dụ này, để xác định được địa chỉ có 6 thông điệp DNS được trao đổi: 3 thông điệp yêu cầu và 3 thông điệp trả lời.



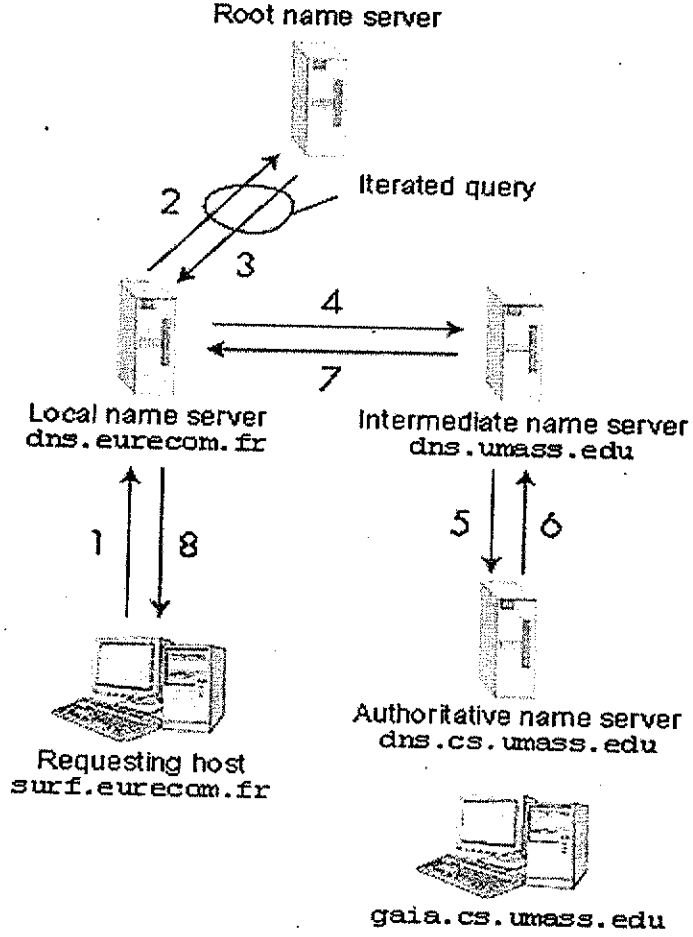
Hình 2.21 Các truy vấn DNS

Giả thiết root name server biết địa chỉ IP của authoriative name server của mọi máy tính như trên có thể không đúng. Với tên một máy tính, root nameserver có thể chỉ biết được địa chỉ IP của một name server trung gian mà chính name server trung gian này mới biết được địa chỉ IP của authoriative nameserver của máy tính đó. Để minh họa điều này vẫn xét ví dụ máy tính surf.eurecom.fr cần xác định địa chỉ IP của gaia.cs.umass.edu. Giả sử trường Đại học Massachusets (Univ of Massachusets) có name server cho toàn bộ trường đại học, đó là dns.umass.edu. Ta cũng giả sử tiếp rằng mỗi khoa trong trường đại học này có name server riêng, quản lý tên cho tất cả các máy tính trong khoa đó. Khi root name server nhận được yêu cầu xác định địa chỉ IP cho một tên máy tính có tận cùng là umass.edu, nó sẽ gửi yêu cầu tới name server dns.umass.edu. Name server này gửi tất cả các yêu cầu có tên máy tính tận cùng là cs.umass.edu cho authoritative name server quản lý tất cả máy tính có tên tận cùng là cs.umass.edu. Authoriative name server này (dns.cs.umass.edu) gửi kết quả tới name server trung gian (dns.umass.edu) và name server này sẽ gửi tiếp kết quả tới root name server. Root name server sẽ gửi tiếp kết quả tới local name server của máy tính yêu cầu. Trong ví dụ này, 8 thông điệp DNS được gửi (Hình 2.21). Thực ra có thể có nhiều hơn 8 thông điệp DNS được trao đổi vì có thể có nhiều name server trung gian ở giữa root name server và authoriative name server.



Hình 2. 22 Các truy vấn DNS

Trong ví dụ trên, tất cả các truy vấn được gọi là đệ quy (recursive query). Khi máy tính hay name server A gửi thông điệp yêu cầu tới name server B, name server B sẽ thay mặt A nhận thông điệp chứa kết quả và sau đó gửi kết quả tới A. Tuy nhiên DNS cho phép các truy vấn tương tác (iterative query) ở bất kì giai đoạn nào trong quá trình từ máy tính yêu cầu đến authoriative name server. Khi name sever A gửi một truy vấn tương tác tới name server B, nếu name server B không có ánh xạ được yêu cầu, nó sẽ gửi cho A thông điệp trả lời chứa địa chỉ IP của name server kế tiếp trên chuỗi, giả sử là name sever C. Sau đó name server A trực tiếp gửi thông điệp yêu cầu tới name server C.



Hình 2. 23 Các truy vấn đệ quy và tương tác

Các truy vấn trong dãy truy vấn liên tiếp có thể là tương tác hoặc đệ quy như minh hoạ trên Hình 2. 23. Thông thường tất cả các truy vấn - ngoại trừ truy vấn từ local name server tới root name server là đệ quy, truy vấn tới root name server thường là tương tác (bởi vì root name server phải xử lý một lượng lớn các yêu cầu nên cần làm giảm số lượng truy vấn tới root name server).

Một đặc tính quan trọng của DNS là lưu trữ tạm thời các bản ghi DNS (**DNS caching**). Trên thực tế, DNS lưu trữ tạm thời (cache) để làm giảm độ trễ cũng như làm giảm số thông điệp DNS trao đổi trên mạng. Ý tưởng này rất đơn giản: Khi nhận được ánh xạ DNS của máy tính nào đó, bên cạnh việc gửi tiếp thông điệp, name server sẽ lưu ánh xạ này vào bộ nhớ cục bộ (ổ đĩa cứng hay RAM). Với ánh xạ tên máy - địa chỉ IP được lưu trữ sẵn, nếu có một truy vấn khác yêu cầu địa chỉ IP của cùng tên máy mà name server vừa lưu trữ, nameserver sẽ xác định ngay được địa chỉ IP mong muốn, kể cả khi nó không là authoritiative name server cho máy tính đó. Để khắc phục tình trạng lưu trữ thông tin cũ, thông tin được lưu trữ tạm thời sẽ bị xoá bỏ sau một khoảng thời gian (thường là hai ngày).

2.5.3 Bản ghi DNS

Name server lưu giữ các bản ghi tài nguyên (resource record) cho các ánh xạ Tên máy / Địa chỉ IP. Mỗi thông điệp trả lời DNS chứa một hay nhiều bản ghi tài nguyên. Trong phần này chúng ta sẽ nói qua về bản ghi tài nguyên và thông điệp DNS. Về chi tiết, bạn có thể xem trong DNS RFC [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. Trong các bản ghi ví dụ dưới đây, chúng ta bỏ qua trường TTL. Ý nghĩa của trường Name và Value phụ thuộc vào trường Type:

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. Ví dụ, (relay1.bar.foo.com, 145.37.93.126, A) là một bản ghi Type A.

Nếu Type = NS thì Name là một miền (như là foo.com) và Value là tên máy của authoriative name server của các máy tính trong miền đó. Bản ghi này thường được sử dụng để gửi tiếp các truy vấn DNS. Ví dụ 1 bản ghi Type NS: (foo.com, dns.foo.com, NS)

Nếu Type = CNAME thì Value là tên đầy đủ của máy có tên bí danh nằm trong Name. Bản ghi kiểu này cho phép xác định tên đầy đủ của một máy tính từ tên bí danh. Ví dụ một bản ghi CNAME: (foo.com, relay1.bar.foo.com, CNAME).

Nếu Type = MX thì Value là tên máy của mail server có tên bí danh nằm trong Name. Ví dụ, bản ghi kiểu MX (foo.com, mail.bar.foo.com, MX). Bản ghi MN cho phép mail server có tên bí danh đơn giản.

Nếu một name server là authoritiative name server cho một máy tính nào đó thì name server sẽ chứa bản ghi kiểu A của máy tính đó (ngay cả nếu name server đó không là authoritiative name server thì có thể nó chứa bản ghi kiểu A trong bộ nhớ cache của nó). Nếu name server không là authoritiative name server của máy tính được hỏi thì nó sẽ chứa một bản ghi kiểu NS cho miền của máy tính này, và nó cũng có một bản ghi kiểu A xác định địa chỉ IP của name server của miền này đặt trong trường Value của bản ghi NS. Ví dụ, root name server không là authoritiative name server cho máy tính gaia.cs.umass.edu, root server sẽ có một bản ghi cho miền chứa umass.edu ví dụ (umass.edu, dns.umass.edu, NS). Root server đó đồng thời cũng có một bản ghi kiểu A cho phép xác định địa chỉ IP của name server dns.umass.edu, chẳng hạn (dns.umass.edu, 128.119.40.111, A).

2.5.4 Thông điệp DNS

Có hai loại thông điệp DNS: thông điệp yêu cầu và thông điệp trả lời. Cả hai kiểu thông điệp này có chung khuôn dạng minh hoạ trên Hình 2.22.

Ý nghĩa các trường trong thông điệp như sau:

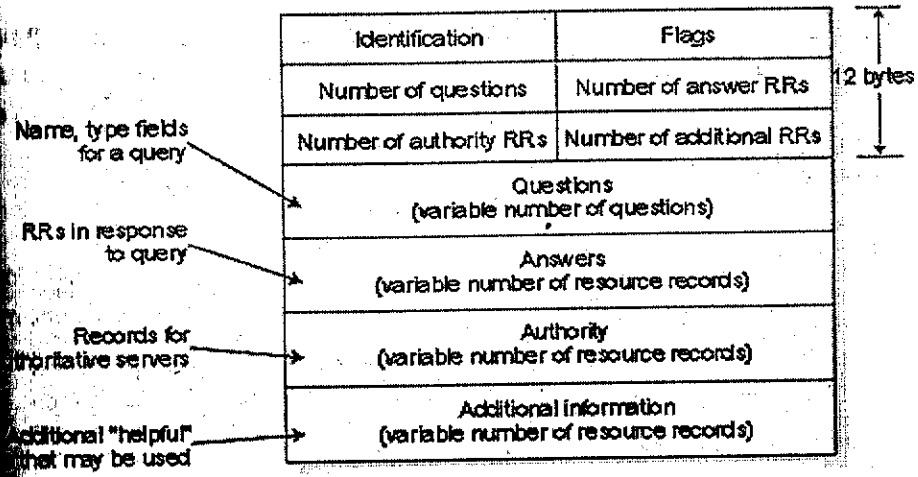
12 byte đầu tiên là phần tiêu đề. Phần tiêu đề có một số trường. Trường đầu tiên là một định danh 16 bit cho mỗi thông điệp yêu cầu. 16 bit định danh này được ghi lại vào thông điệp trả lời, cho phép client xác định được đấy là câu trả lời cho thông điệp yêu cầu nào. Có nhiều cờ trong trường cờ (mỗi cờ ứng với một bit). **Cờ truy vấn (query/reply flag)** xác định thông điệp là yêu cầu (0) hay là trả lời (1). **Cờ authoritiative** được đặt trong thông điệp trả lời khi name server là authoriative name server của tên máy tính cần xác định địa chỉ IP. **Cờ mong muốn đệ quy (recursive-desired query)** được đặt khi client (máy tính hay name server) mong muốn name server thực hiện truy vấn đệ quy khi nó không có bản ghi đó. **Cờ chấp nhận đệ quy (recursion-available flag)** được đặt trong thông điệp trả lời nếu name server đó hỗ trợ đệ quy. Trong phần tiêu đề cũng có 4 trường số lượng, các trường này xác định số lượng các bản ghi trong 4 phần dữ liệu sau phần tiêu đề.

Phần câu hỏi (Question session) chứa thông tin về các câu hỏi được tạo ra. Nó bao gồm (1) trường tên chứa tên đang được hỏi và (2) trường kiểu xác định kiểu câu hỏi cho tên máy tính đó (Kiểu A cho tên máy tính, kiểu MX cho mail server).

Trong thông điệp trả lời từ server name, phần trả lời (answer section) chứa các bản ghi tài nguyên cho tên được yêu cầu trước đó. Chú ý rằng mỗi bản ghi tài nguyên có 4 trường: Type (A, NS, CNAME, MX), Name, Value, TTL. Thông điệp trả lời có thể có nhiều bản ghi tài nguyên vì tên máy tính có thể ứng với nhiều địa chỉ IP.

Mục thẩm quyền (authority section) chứa các bản ghi của các authoritative server.

Mục phụ trợ (additional section) chứa các bản ghi "hữu ích" khác. Ví dụ trường trả lời trong thông điệp trả lời một truy vấn MX sẽ chứa tên đầy đủ của mail server có tên bí danh đặt ở trong Name. Phần phụ trợ có thể chứa một bản ghi kiểu A cung cấp địa chỉ IP cho chính mail server đó.



Hình 2. 24 Khuôn dạng thông điệp DNS

Các phần trên mô tả cách thức lấy dữ liệu trong cơ sở dữ liệu DNS. làm thế nào để đưa được dữ liệu vào cơ sở dữ liệu? Cho tới gần đây, dung của server DNS được cấu hình tĩnh, ví dụ, thông qua file cấu hình người quản trị hệ thống tạo ra. Gần đây, lựa chọn UPDATE được đưa giao thức DNS cho phép dữ liệu được tự động thêm vào hay xoá bỏ cơ sở dữ liệu thông qua thông điệp DNS. RFC 2136 đặc tả quá trình nhật động của DNS.

CÁC ỨNG DỤNG THEO KIẾN TRÚC NGANG HÀNG

Tháng 5/1999, Shawn Fanning và Sean Parker sáng lập ra công ty ster Inc và đã mở ra một cuộc Cách mạng trong ngành công nghệ thông Tại thời điểm đó, Napster là dịch vụ chia sẻ file ngang hàng duy nhất mạng Internet.

Chính nhu cầu chia sẻ các loại phần mềm không bản quyền thúc đẩy ra đời các cộng đồng ảo có tổ chức nhưng lại kết nối lỏng lẻo trên rnet. Napster không chỉ đáp ứng nhu cầu tìm kiếm và chia sẻ các file âm c mà còn hình thành nên một cộng đồng ảo trên đó. Tham gia vào cộng

đồng này, một người có thể dễ dàng nén các đĩa CD âm nhạc ra file MP3 để chia sẻ với những người khác.

Trong lịch sử

Năm 1999, Shawn Fanning công bố chương trình đầu tiên (thực hiện sau vài tháng) đáp ứng được nhu cầu một nhóm nhỏ bạn bè. Nói chung mỗi người đều có nhiều các file nhạc (MP3) trong ổ cứng trên máy tính cá nhân, máy tính cá nhân đều có kết nối Internet. Vấn đề ở đây là làm thế nào tìm được file mà người sử dụng (NSD) muốn kiếm theo một tiêu chí nào đó (Tên bài hát / Ca sỹ / Nhạc sỹ/ Album...). Chương trình mà Fanning viết có thể kiểm soát được những thông tin như vậy.

Ở server trung tâm, Chương trình của Fanning kiểm soát những file mà hệ thống có thể cung cấp và vị trí chính xác của file (tức là địa chỉ máy tính chứa file). Khi người dùng kết nối vào hệ thống (bằng phần mềm của Fanning), danh sách tất cả các file mà người dùng muốn chia sẻ sẽ được cập nhật lên server trung tâm (chú ý là chỉ danh sách các file, còn các file thực sự vẫn nằm trên máy tính đầu cuối – gọi là nút). Khi NSD muốn tìm kiếm file nào đó, từ hệ thống CSDL của mình, server trung tâm sẽ trả về cho NSD danh sách các máy tính có chứa file cần tìm. NSD chỉ việc liên lạc trực tiếp đến máy tính có file để xin tải về.

Mạng Napster – với khả năng hỗ trợ tìm kiếm và tải file MP3 với khối lượng lớn – đã phát triển với tốc độ cực kỳ nhanh, hình thành một cộng đồng 30 triệu NSD chia sẻ 2,8 tỷ file trước khi chấm dứt hoạt động theo phán quyết của Tòa án Hoa Kỳ.

Khởi đầu từ năm 2003, RIAA (Hiệp hội âm nhạc Hoa Kỳ) đã có chiến lược mới để kiểm soát mạng P2P. Sau khi RIAA thua tại Tòa Thượng thẩm Liên bang khi ISP Verizon không cung cấp thông tin về khách hàng thuê bao, RIAA bắt đầu thu thập thông tin về hành vi của người sử dụng (chứ không phải ISP hay nhà cung cấp phần mềm P2P). Tháng 9/2003, RIAA đã đưa ra tòa 261 người chia sẻ các file nhạc bất hợp pháp, ngày nay số lượng vụ kiện kiểu này hàng năm là 13000.

Giờ đây, sau nhiều cố gắng của Hiệp hội công nghiệp thu âm Hoa Kỳ (RIAA), Quốc hội Hoa Kỳ đã đưa ra những điều khoản quy định việc chia sẻ, hạn chế việc tự do chia sẻ các phần mềm hợp pháp trên các mạng ngang hàng.

Người ta thường xem Kỷ nguyên Mạng ngang hàng bắt đầu với Napster, tuy nhiên ta thấy những phần mềm nhắn tin (Yahoo Messenger) đã

manh nha ý tưởng này. Nhắn tin là một hình thức lai tạp – vừa tận dụng tính tức thời của hệ thống điện thoại lại vừa có tính kiểm soát của hệ thống thư tín điện tử.

Nói chung, ứng dụng P2P có các đặc điểm sau:

- **Chia sẻ các tài nguyên và dịch vụ phân tán**. Trong mạng P2P, các nút thành viên vừa đóng vai trò client, vừa đóng vai trò server; nghĩa là vừa là nhà cung cấp nhưng cũng là người tiêu thụ các tài nguyên (Tài nguyên ở đây có thể là thông tin, file, năng lực tính toán...).

- **Không tập trung**. Không có điểm tập trung nào đóng vai trò điều phối chung của toàn bộ hệ thống. Truyền thông giữa các nút diễn ra trực tiếp, không có nút nào kiểm soát được tất cả các nút khác. Ở đây có sự phân biệt giữa mạng P2P thuần túy (các nút hoàn toàn bình đẳng với nhau) và mạng P2P lai tạp (các nút không còn bình đẳng hoàn toàn, có một nhóm nút sẽ đóng vai trò kiểm chứng và tìm kiếm theo chỉ mục). Kiểu lai tạp kết hợp giữa mạng P2P thuần túy và mô hình client/server truyền thống.

- **Tự trị**. Mỗi nút trong mạng P2P tự quyết định khi nào tham gia mạng và mức độ đóng góp chia sẻ cho cộng đồng.

Các ứng dụng P2P có thể phân loại vào 3 nhóm sau đây:

- Nhắn tin tức thì (Instant messaging)

- Chia sẻ file

- Tính toán mạng lưới

Việc phân chia nhóm như vậy cũng chưa rõ ràng và ranh giới giữa các nhóm còn mập mờ, và biến đổi rất nhanh chóng.

2.6.1 Nhắn tin tức thì

Ngày nay Nhắn tin tức thì (IM - Instant Messaging) được sử dụng cực kỳ rộng rãi do có khả năng truyền thông điệp tức thời (giống điện thoại) một cách có kiểm soát (giống email) và khả năng duy trì đồng thời nhiều cuộc hội thoại. Các dịch vụ IM như I-seek-you (ICQ), MSN Messenger của

Microsoft, AOL Instant Messenger® (AIM), Yahoo Messenger (YIM) trở nên cực kỳ thông dụng (đặc biệt với giới trẻ) khi các dịch vụ này tích hợp thêm nhiều tính năng như truyền file, chia sẻ dữ liệu, tương tác đa phương tiện...

Chức năng của IM

Cho đến năm 2000, không có một giao thức chuẩn nào quy định cách thức hoạt động của các hệ thống IM. RFC2778 "A model for Presence and Instant Messaging" và RFC2779 "Instant Messaging/Presence Protocol" đưa ra tập hợp các hướng dẫn hình thành nên cấu hình cơ bản cho một hệ thống IM. Các phần mô tả sau đây chủ yếu dựa trên các khuyến nghị RFC2778 và RFC2779.

Dịch vụ Hiển thị Thông tin (Presence Service) là dịch vụ quan trọng nhất trong hệ thống IM, xác định có thể liên lạc được với một đối tượng hay không và trạng thái của đối tượng đó. Ngoài ra, dịch vụ này còn cung cấp các cơ chế đảm bảo tính riêng tư, chẳng hạn lựa chọn một nhóm đối tượng được phép nhìn thấy trạng thái của mình và được quyền gửi thông điệp tới.

Dịch vụ chuyển phát tin nhắn. Là các tiện ích cho phép chuyển phát các tin nhắn (có thể dưới dạng văn bản hay âm thanh) giữa những người sử dụng trên mạng IM.

Các dịch vụ bổ trợ. Bên cạnh các dịch vụ cơ sở trên, các hệ thống IM bổ sung rất nhiều tính năng mới để thu hút khách hàng. Các dịch vụ bổ trợ được liệt kê trong RFC2778 và RFC2779.

Các thành phần của IM

IM Server. Cung cấp các dịch vụ IM như hiển thị thông tin, cảnh báo, đăng ký tài khoản, và nhắn tin. Người sử dụng (qua phần mềm Client) phải kết nối với Server để có trạng thái trực tuyến (online) trong mạng IM. Mạng IM có thể gồm nhiều server cùng làm việc với nhau để cung cấp khả năng mở rộng trong trường hợp có nhiều client cùng kết nối.

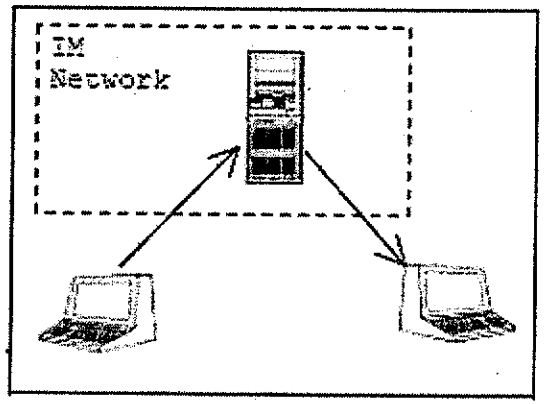
Truyền thông giữa các server thường "trong suốt", vì người sử dụng nói chung không biết được mình kết nối đến những server nào. Server của các mạng IM lớn thường được đặt tại các công ty sở hữu mạng IM.

IM Client. Phần mềm client là "điểm giao tiếp" của người sử dụng với mạng IM. Sau khi cài đặt trên máy tính, NSD có thể sử dụng IM client để kết nối và sử dụng các dịch vụ do IM server cung cấp. Client cũng được sử dụng để lưu trữ các thông tin cục bộ của người sử dụng (chẳng hạn mật khẩu, tên truy cập phục vụ mục đích tự động truy cập hay nhật ký hội thoại).

Mô hình kết nối

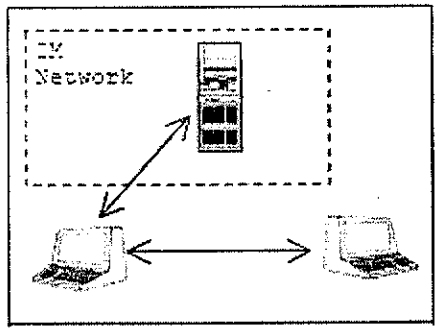
Có hai mô hình cơ sở được áp dụng trên các mạng IM

Client-Server hay kiểu tập trung. Tất cả thông tin trao đổi trên mạng đều phải đi qua server, có nghĩa là hai client không kết nối trực tiếp được với nhau mà kết nối gián tiếp qua server (Xem Hình 2.25)



Hình 2.25 Mô hình kết nối IM kiểu tập trung

Peer-to-Peer hay kiểu phân tán. Dữ liệu được trao đổi trực tiếp giữa các client (không đi qua server). Kiểu kết nối này thường được sử dụng khi truyền file (để làm giảm tải cho server). Server trong mạng kiểu này chỉ làm nhiệm vụ kiểm soát trạng thái các client và giúp các client trao đổi trực tiếp với nhau. Kiểu kết nối này được minh họa trên Hình 2.26.



Hình 2.26 Mô hình IM kiểu phân tán

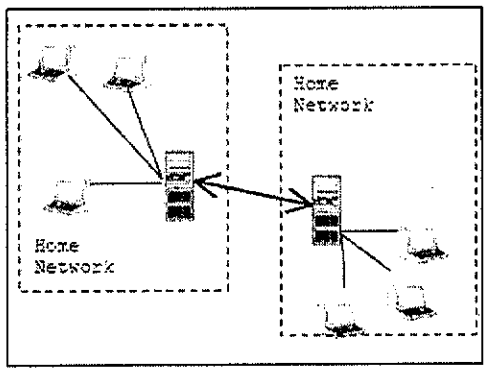
Cấu hình Server

Để tăng khả năng mở rộng, hệ thống IM có thể sử dụng một hay nhiều Server.

Kiến trúc một Server. Mọi dịch vụ IM đều thực hiện trên một server duy nhất (tạo tài khoản, kiểm chứng người sử dụng và các dịch vụ khác). Kiến trúc này dễ bảo trì server và tăng cường khả năng bảo mật, nhưng khó mở rộng hệ thống.

Kiến trúc nhiều Server. Phần lớn các mạng IM lớn đều sử dụng nhiều server. Các server thường được chia thành hai loại: Nhân bản và Dịch vụ phân tán.

Trong kiểu thứ nhất, mỗi server có thể thực hiện được tất cả các dịch vụ IM. Các server giống hệt nhau (nên được gọi là nhân bản) và được kết nối với nhau. Các client "đăng ký" đến một server "nhà" và "nói chuyện" với client thuộc server "nhà" khác. Kiến trúc Jabber theo kiểu này.



Hình 2. 27 Kiến trúc nhiều Server nhân bản

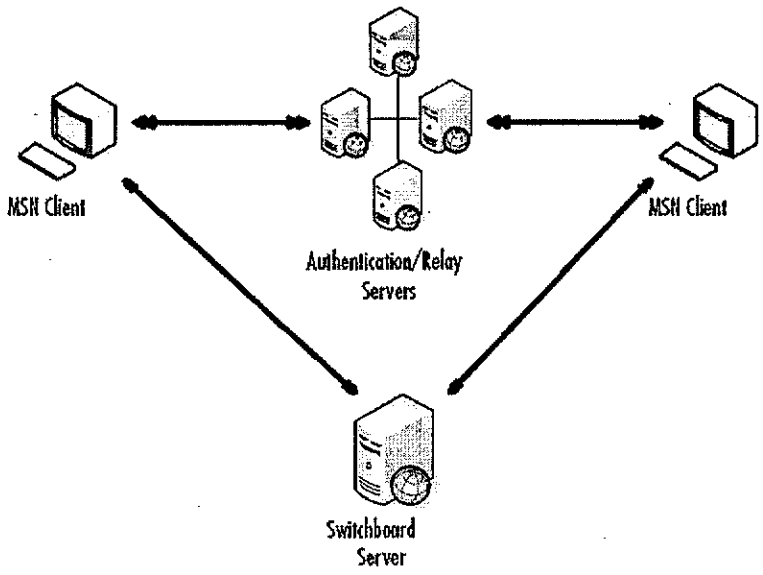
Trong kiểu thứ hai, các dịch vụ khác nhau được cài đặt trên các server khác nhau. Một server có thể cung cấp dịch vụ đăng ký tài khoản và đăng nhập hệ thống, trong khi chức năng truyền tin nhắn và thông báo được cài đặt trên các server khác.

Kiến trúc MSNP cung cấp nhiều server kiểm chứng đặt tại miền messenger.hotmail.com. Sau khi đăng nhập vào miền này, client được chuyển hướng tới một trong các server kiểm chứng. Sau khi kiểm chứng thành công, client có thể được chuyển kết nối sang một trong các server thực hiện nhiệm vụ cảnh báo. Client sẽ kết nối với server nào phụ thuộc vào vị trí địa lý của server hay tải của server.

2.6.6. Kiến trúc Hệ thống MSN

Hai hệ thống IM được sử dụng rộng rãi nhất hiện nay là AIM và MSN trong khi ICQ đứng thứ 5 trong danh sách. AIM và ICQ được quản lý bởi AOL Time Warner và sử dụng giao thức OSCAR (Open System for Communications in Real-time) trong khi MSN được Microsoft quản lý và sử dụng giao thức MSNP (MSN Protocol).

MSN Protocol (MSNP)



Hình 2. 28 Kiến trúc MSN

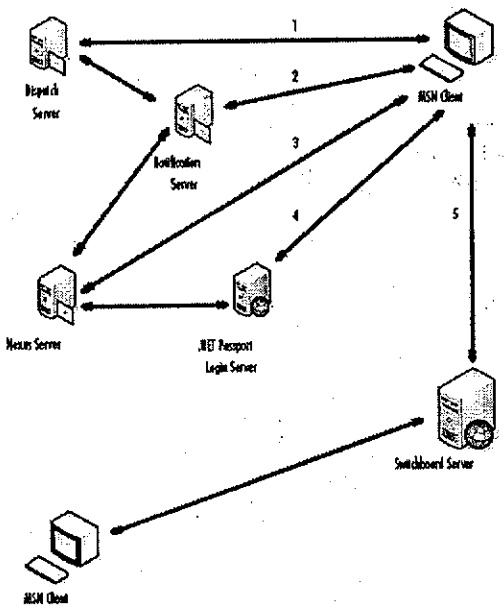
Server và Client trong mạng MSN trao đổi các lệnh được mã hóa theo chuẩn UTF-8 qua TCP socket.

Ở đây có ba kiểu server: Dispatch Server (DS), Notification Server (NS) và Switchboard Server (SS).

Nhiệm vụ chính của DS là thỏa thuận sử dụng phiên bản giao thức nào với client, sau đó sẽ xác định server NS nào kiểm soát người sử dụng. Sau khi xác định, địa chỉ NS được chuyển cho client.

NS là thành phần quan trọng nhất trong mạng IM. Sau khi kiểm chứng để đăng nhập thành công, Client và NS phải đồng bộ với nhau về trạng thái cũng như trao đổi các thông tin dị bộ. Một số sự kiện trao đổi giữa client và NS là khi thay đổi trạng thái (trực tuyến, ngoại tuyến hoặc rảnh rỗi), yêu cầu làm việc với SS (trình bày trong phần sau). Ngoài ra một vài thông báo khác cũng có thể trao đổi giữa NS và client (chẳng hạn thông báo có thư mới).

SS là thành phần trợ giúp client tạo các phiên hội thoại, chuyển tin nhắn giữa các client. Mỗi một cuộc hội thoại sẽ mở một kết nối từ client tới SS. Khi Client 1 muốn nói chuyện với Client 2, Client 1 sẽ gửi yêu cầu tới NS, NS sẽ gán cho Client 1 một SS nào đó (SS 1), đồng thời gửi thông báo tới NS của Client 2, NS này sẽ yêu cầu Client 2 kết nối tới SS 1 để nhận tin nhắn từ Client 1.



Hình 2.29 Minh họa các bước của MSNP

Sau đây chúng ta thử phân tích một số hoạt động của giao thức MSNP.

Trong quá trình hoạt động, người sử dụng MSN thực hiện 2 bước sau: Đăng nhập vào hệ thống (lấy danh sách bạn bè và trạng thái của từng bạn bè) và sau đó Gửi và Nhận tin nhắn.

Đầu tiên client (MSN) mở một kết nối TCP tới địa chỉ 64.4.13.58 ở port 1863. Sau khi thiết lập kết nối, client và server sẽ thỏa thuận với nhau sử dụng phiên bản nào của giao thức. MSN client gửi thông điệp sau chứa danh sách các giao thức client có thể sử dụng được:

VER 0 MSNP7 MSNP6 MSNP5 MSNP4 CVRO

Trong giao thức MSNP, trường "trial id" được khởi tạo là 0 và được gửi đi trong tất cả các thông điệp, cứ sau mỗi thông điệp, giá trị "trial id" được tự động tăng lên 1. Server trả lời kiểu như sau:

VER 0 MSNP7 MSNP6 MSNP5 MSNP4

Khi đó Client và Server thỏa thuận được giao thức sẽ sử dụng. Kế tiếp client hỏi server phương pháp mã hóa khi kiểm chứng.

INF 1

Khác Yahoo, MSN không gửi mật khẩu dưới dạng nguyên không mã hóa. MSN sẽ mã hóa mật khẩu để mật khẩu không bị lộ trên đường truyền. Server trả lời bằng thông điệp sau:

INF 1 MD5

Ở đây MD5 là phương pháp bảo mật mà server hỗ trợ. Kế tiếp client gửi username cho server

USR 2 MD5 I venky_dude@hotmail.com

Server sẽ kiểm tra xem có thông tin gì về người dùng này hay không. Nếu có Server sẽ gửi thông điệp kiểu sau cho client:

XFR 2 NS 64.4.13.55:1863 0

Ở đây server báo cho client biết phải liên lạc với Notification Server (NS) có địa chỉ 64.4.13.55 và cổng 1863. Sau đó Client và server đóng kết nối này, client sẽ mở kết nối với NS mới nhận được (64.4.13.55).

(client) VER 3 MSNP7 MSNP6 MSNP5 MSNP4 CVRO
(server) VER 3 MSNP7 MSNP6 MSNP5 MSNP4
(client) INF 4
(server) INF 4 MD5
(client) USR 5 MD5 I venky_dude@hotmail.com

Sau đó NS server biết được các thông tin về người dùng định đăng nhập. Server sẽ trả lời như sau:

USR 5 MD5 S 989048851.1851137130

Chuỗi "989048851.1851137130" là giá trị Hash theo kiểu MD5. Nó là một hàm băm do server tạo ra và được sử dụng cho mục đích kiểm chứng. Client sẽ gửi server mật khẩu đã được mã hóa theo MD5. Client sẽ gửi hàm băm được tính tạo thành từ chuỗi trên kết hợp với mật khẩu của mình cho server, chẳng hạn: **3b7926d277068ec49576a0c40598ff21**.

USR 6 MD5 S 3b7926d277068ec49576a0c40598ff21

Nếu mật khẩu này đúng, server sẽ trả lời dạng sau:

USR 6 OK venky_dude@hotmail.com venkat

Từ cuối cùng là tên của người sử dụng. Trong SMNP7, server có thể gửi một vài thông tin bổ trợ, chẳng hạn thông tin về người dùng và mã kiểm chứng (tương tự cookie) sử dụng cho các mục đích khác.

MSG Hotmai Hotmail 362
MIME-Version: 1.0
Content-Type: text/x-msmsgspro file; charset=UT
LoginTime: 1011252477
EmailEnabled: 1
MemberIdHigh: 84736
MemberIdLow: - 1434729391
lang _preference: 103
preferredEmai l: venky_dude@hotmail.com
country: IN
PostalCode:
Gender: M
Kid:0
Age: 22
sid: 517
kv: 2
MSPAuth:
2AAAAAAADU0p4uxxxJtDJozJSlUTS0i7YpwnC9PUHRv56YKxxxCTWmg$S

Giờ đây người sử dụng đã đăng nhập hệ thống, nhưng vẫn ở trong trạng thái không trực tuyến, để chuyển trạng thái, client sẽ gửi lệnh sau:

CHG 7 NLN

Người dùng không gửi ngay lệnh này vì có thể người sử dụng muốn đăng nhập hệ thống nhưng dưới chế độ ẩn. Trong thông điệp Server trả lời có liệt kê danh sách các "bạn bè" ở trạng thái khác nhau, chẳng hạn:

CHG 7 NLN
ILN 7 NLN btxxxe@hotmail.com nick
ILN 7 AWY wmpyxxx@msn.com mike
ILN 7 BSY tehpxxpxx@hotmail.com yeaxxx
MSG Hotmail Hotmail 223
MIME-Version: 1.0
Content-Type: text/x-msmsgsinitialemailnotification; charset=UTF-8
Inbox-Unread: 293
Folders-Unread: 0
Inbox-URL: /cgi-bin/HoTMaiL
Folders-URL: /cgi-bin/folders
Post-URL: http://www.hotmail.com

Để có danh sách các bạn bè, client có thể sử dụng lệnh:

LST 9 RL

Nhận được lệnh này, server sẽ gửi cho client danh sách các "bạn bè", ví dụ:

LST 9 RL 69 1 19 venky_dude@hotmail.com venkat
LST 9 RL 69 2 19 puxxxxx@hotmail.com PUJA
LST 9 RL 69 3 19 vancxxxxx@hotmail.com ramachandran
LST 9 RL 69 4 19 moxxxxx@hotmail.com chandramouli
LST 9 RL 69 5 19 v_n_xxxxx@hotmail.com Narayanaswamy
LST 9 RL 69 6 19 dexxxxx@hotmail.com Venkatesh
LST 9 RL 69 7 19 lousydxxxxx@hotmail.com DKV

LST 9 RL 69 8 19 hexxxxxr@hotmail.com Hetchar%20Ramachandran
LST 9 RL 69 9 19 ambxxxxx@hotmail.com Aiyer
LST 9 RL 69 10 19 suxxx@hotmail.com Ganesh
LST 9 RL 69 11 19 deexxxxx@hotmail.com Deepak
LST 9 RL 69 12 19 anilxxxxx@hotmail.com anil
LST 9 RL 69 13 19 dixxxxx@hotmail.com <Diamond>
LST 9 RL 69 14 19 nvxxxx@hotmail.com giri
LST 9 RL 69 15 19 shxxx@hotmail.com Hari
LST 9 RL 69 16 19 radhikashuxxxxx@hotmail.com radhika
LST 9 RL 69 17 19 eskaxxxxx@hotmail.com kannan
LST 9 RL 69 18 19 shaxxxxx@hotmail.com Shankar
LST 9 RL 69 19 19 puneetagarxxxxx@hotmail.com puneet

Cứ mỗi khi có một người bạn chuyển sang trạng thái trực tuyến, NS sẽ gửi thông điệp kiểu sau cho client:

NLN 10 NLN deaxxxx@hotmail.com Venkatesh

Khi một người bạn thoát khỏi mạng (không trực tuyến), NS gửi thông điệp kiểu sau:

FLN 10 FLN deaxxxx@hotmail.com

Như vậy, người sử dụng đã đăng nhập thành công vào mạng MSN, sau đó người sử dụng có thể gửi và nhận các tin nhắn.

NSD chỉ có thể tham gia hội thoại trong hai trường hợp sau: NSD khởi tạo cuộc hội thoại với ai đó hoặc ai đó khởi tạo cuộc hội thoại với NSD.

NSD khởi tạo cuộc hội thoại

Tất cả các cuộc hội thoại đều diễn ra qua SS. Client gửi lệnh sau yêu cầu địa chỉ IP của Switchboard server

XFR 9 SB

NS gửi thông điệp trả lời chứa địa chỉ IP, cổng của SS cùng với mã CKI. CKI là phương pháp bảo mật mà client phải sử dụng khi liên lạc với SS.

XFR 9 SB 64.4.13.88:1863 CKI 989487642.2070896604

Kế tiếp Client sẽ tạo ra một kết nối mới tới SS có địa chỉ vừa nhận được (64.4.13.88). Chú ý rằng client vẫn duy trì kết nối tới NS, kết nối này chỉ được đóng khi NSD thoát khỏi hệ thống. Sau khi kết nối thành công, chúng ta gửi lệnh sau cho SS:

USR 1 venky_dude@hotmail.com 989487642.2070896604

Nếu CKI đúng, SS sẽ gửi cho chúng ta lệnh tương tự sau:

USR 1 OK venky_dude@hotmail.com venkat

Sau đó, NSD có thể "gọi" bạn bè để tham gia hội thoại thông qua lệnh sau

CAL 2 deadxxx@hotmail.com

Thông điệp trả lời từ Server sẽ định danh phiên hội thoại (session id). Định danh này sẽ được chuyển tiếp cho NSD sẽ tham gia hội thoại.

CAL 2 RINGING 11717653

Khi NSD khác trả lời thông điệp và sẵn sàng hội thoại, SS gửi lệnh sau cho client:

JOI deadlee@hotmail.com Venkatesh

Lệnh này chứng tỏ NSD khác đã tham gia hội thoại và sẵn sàng gửi/nhận các tin nhắn.

NSD nhận yêu cầu hội thoại từ NSD khác

Khi NSD 1 được NSD 2 mời "hội thoại" thì NS của NSD 1 gửi thông điệp sau cho client của NSD 1:

RNG 11742066 64.4.13.74:1863 CKI 989495494.750408580 deaxxxx@hotmail.com Venkatesh

Thông điệp này chứa session id, địa chỉ IP và cổng của SwitchBoard, mã hash CKI và tên người dùng khởi tạo cuộc nói chuyện.

NSD kết nối đến SS và gửi lệnh sau:

ANS 1 venky_dude@hotmail.com 989495494.750408580 11742066

Ở đây, NSD gửi tên đăng nhập, mã CKI và session ID mà server đã gửi

Server trả lời:

IRO 1 1 1 deaxxxx@hotmail.com Venkatesh

ANS 1 OK

Bây giờ hai bên có thể tiến hành hội thoại với nhau. Trước khi nhận và gửi thông điệp, chúng ta sẽ xem cách thức client xây dựng thông điệp như thế nào.

Tiêu đề thông điệp có khuôn dạng sau:

MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
X-MMS-IM-Format: FN=Microsoft%20Sans%20Serif; EF=; CO=0; CS=0; PF=22

Khuôn dạng thông điệp được gửi đi:

```
MSG 2 N 137
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
X-MMS-IM-Format: FN=Microsoft%20Sans%20Serif; EF=; CO=0; CS=0; PF=22
hello
```

2 là "trial id", giá trị này được tự động tăng lên 1 sau mỗi thông điệp gửi đi, 137 là độ dài toàn bộ thông điệp Thông điệp trả lời có khuôn dạng tương tự:

```
MSG deaxxxx@hotmail.com Venkatesh 137
MIME-Version: 1.0
Content-Type: text/plain; charset=UTF-8
X-MMS-IM-Format: FN=Microsoft%20Sans%20Serif; EF=; CO=0; CS=0; PF=22
Hello
```

2.6.3 Kiến trúc chia sẻ file ngang hàng Gnutella

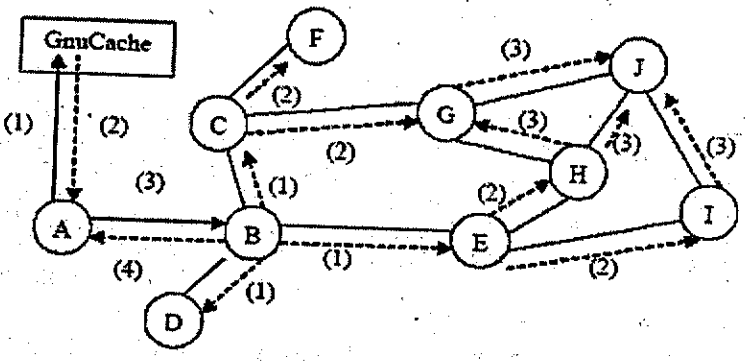
Gnutella lúc đầu được phát triển tại AOL, sau đó tách ra và được nhiều người phát triển. Hiện tại có khá nhiều hệ thống kế thừa kiến trúc này, chẳng hạn LimeWire, BearShear, Gnucleus, XoloX, và Shareaza.

Kiến trúc Gnutella

Gnutella không có một server chỉ mục trung tâm như Napster. Mạng Gnutella gồm các nút ngang hàng nhau (peer), có thể đối một kết nối trực tiếp với nhau. Các nút này còn được gọi là servent, vì vừa đóng vai trò client (khi yêu cầu nút khác) lẫn server (khi đáp ứng yêu cầu từ nút khác). Các nút có bản ghi thông tin chỉ mục về toàn bộ (hoặc một phần) hệ thống.

Muốn gia nhập mạng Gnutella, nút phải xác định được địa chỉ một nút đã nằm trong mạng Gnutella. Điều này có thể thực hiện bằng cách sử dụng GnuCache – lưu giữ địa chỉ một số nút luôn luôn kết nối vào mạng Gnutella. Sau đó, nút muốn gia nhập gửi thông điệp **GNUTELLA CONNECT** tới các nút đã nằm trong mạng Gnutella. Nếu chấp nhận, nút nhận được

thông điệp yêu cầu sẽ gửi thông điệp **GNUTELLA OK** (nếu không chấp nhận có thể gửi thông điệp bất kỳ, có khá nhiều lý do từ chối kết nối, chẳng hạn do quá tải kết nối). Sau khi kết nối vào mạng, nút gửi các thông điệp "**ping**" tới các "hàng xóm" để tìm kiếm thêm các nút khác. Do bản chất động của mạng (các nút có thể thoát khỏi mạng bất kỳ lúc nào) nên một nút thường giữ kết nối với nhiều nút khác (để tránh trường hợp nếu kết nối với một nút duy nhất, nút này thoát khỏi mạng sẽ kéo theo nút mất kết nối với mạng). Khi nhận được thông điệp "**ping**", nút gửi thông điệp "**pong**" trả lời và thông điệp này phải đi theo đúng tuyến đường của thông điệp ping tương ứng.



Hình 2.30 Ví dụ về mạng Gnutella

Thông điệp này chứa thông tin chi tiết của nút, chẳng hạn địa chỉ IP, hiệu port, số lượng file và khối lượng (tính theo byte) được chia sẻ. Mạng utella không có server chỉ mục trung tâm mà mỗi nút duy trì chỉ mục g của mình. Để tìm kiếm file, nút sẽ gửi truy vấn tới tất cả các nút "hàng n". Khi nhận được truy vấn từ hàng xóm, nút sẽ kiểm tra dữ liệu cục bộ mình để xem có đáp ứng được hay không. Trong trường hợp có thể đáp được, nút gửi thông điệp **queryHit** cho nút đang tiến hành tìm kiếm. không đáp ứng được, nút chuyển tiếp thông điệp tới các hàng xóm của h. Chú ý trong trường hợp nút tạo thông điệp **queryHit** nằm đằng sau wall, nút tìm kiếm không thể tạo kết nối TCP tới được, khi đó, nút tìm phải gửi thông điệp **Put** tới nút chứa dữ liệu. Nút này sẽ tạo đường kết TTP tới nút tìm kiếm. Việc truyền file được thực hiện nhờ giao thức TP.

Vì các nút sẽ gửi quảng bá thông điệp, nên để tránh tình trạng làm "tràn ngập" mạng, trong tiêu đề mỗi thông điệp có trường TTL (Time-to-Live), giá trị của trường này sẽ được giảm đi một tại mỗi nút. Tại nút nào mà trường này có giá trị 0, nút sẽ xóa thông điệp tương ứng. Điều này khiến một thông điệp không thể lưu chuyển mãi mãi trên mạng. Mỗi nút phải ghi nhớ định danh thông điệp và trường dữ liệu của mỗi thông điệp nhận được để tránh tình trạng gửi trùng lặp thông điệp. Kế tiếp chúng ta sẽ mô tả giao thức Gnutella.

Trong Lịch sử

Kazaa, một trong những mạng chia sẻ file "lấy lừng" nhất thế giới, vừa chấp nhận trả 100 triệu USD để dàn xếp vụ kiện bản quyền với ngành công nghiệp nhạc số và điện ảnh của Mỹ.

Theo thỏa thuận, chủ sở hữu của Kazaa - Sharman Networks sẽ trả cho 4 hãng đĩa đại gia là Universal Musicm Sony BMG, EMI và Warner Music hơn 100 triệu USD, cùng lời cam kết sẽ "hoạt động hợp pháp". Hiệp hội Điện ảnh Hoa Kỳ cho biết Sharman sẽ tiếp tục duy trì Kazaa và triển khai thêm nhiều công nghệ mới để ngăn chặn việc phát tán trái phép các nội dung có bản quyền trên mạng . Kazaa đã phải tiến hành dàn xếp tại 2 địa điểm cùng một lúc: một là tại Úc, nơi tòa án đã ra phán quyết rằng Sharman Networks vi phạm bản quyền. Vụ kiện kia là tại California, và hai người sáng lập ra Kazaa là Niklas Zennstrom và Janus Friis đều bị buộc tội đồng phạm.

Sau khi bán lại Kazaa cho Sharman Networks vào năm 2002, Zennstrom và Friis đã cùng nhau tạo ra Skype - phần mềm gọi điện Internet thông dụng nhất hiện nay. Sau đấy, họ lại tiếp tục bán Skype cho eBay đổi lấy 2,6 tỷ USD tiền mặt và cổ phiếu.

Biểu tượng là chính

Ngành công nghiệp âm nhạc đã theo đuổi một chính sách pháp lý quyết liệt nhằm tiêu diệt nạn ăn cắp bản quyền qua mạng Internet. RIAA (Hiệp hội ghi âm Hoa Kỳ) đã đệ đơn kiện một loạt mạng chia sẻ file như Kazaa và Grokster cùng hàng chục cá nhân vi phạm. Nỗ lực của họ càng được khuyến khích khi năm ngoái, Tòa án Tối cao Hoa Kỳ tuyên bố các nhà cung cấp nội dung có quyền kiện những hãng công nghệ tiếp tay cho nạn xâm phạm bản quyền. Cùng lúc, những dịch vụ nhạc số hợp pháp như iTunes cũng ăn nên làm ra một cách bất ngờ, khiến cho người dùng nguôi ngoai "nỗi nhớ" mạng P2P. Thật ra, theo giới phân tích, vụ dàn xếp chỉ mang ý nghĩa biểu tượng là chính, bởi Kazaa đã qua thời kỳ đỉnh cao của nó. "Ngày nay, hầu như rất ít người còn dùng Kazaa. Đã xuất hiện nhiều dịch vụ ưu việt hơn, nhưng đây là một chiến thắng pháp lý mang tính biểu tượng cho RIAA", chuyên gia Jonathan Arber nhận định.

Năm ngoái, doanh thu đĩa CD lậu đạt tới 4,5 tỷ USD, tức là cứ 3 đĩa CD được bán thì lại có hơn 1 chiếc là đĩa lậu. Số lượng bản nhạc bị download trái phép là 20 tỷ, tức là bình quân mỗi người trên trái đất download tới 3 bài.

Giả sử NSD A muốn kết nối tới mạng Gnutella để tìm kiếm file (Xem Hình 2.30). A gửi thông điệp truy vấn tới hàng xóm của mình – là B. Đầu tiên B kiểm tra xem có phải đây là lần đầu tiên mình nhận thông điệp này hay không. B nếu có dữ liệu cần tìm sẽ gửi thông điệp **queryHit** tới A. B giảm trường TTL trong tiêu đề thông điệp đi 1, và gửi tiếp thông điệp đến C, D và E. C, D, và E tiếp tục thực hiện các thao tác tương tự và chuyển thông điệp tới F, G, H, và I. 4 NSD này sẽ thực hiện thao tác tương tự để chuyển thông điệp tới J. Giả sử H là nút đầu tiên chuyển tiếp thông điệp tới J, khi đó các thông điệp giống như vậy do G và I chuyển cho J sẽ bị J loại bỏ (vì J đã nhận được thông điệp như thế rồi). Bây giờ, giả sử J có dữ liệu mà A cần, J sẽ gửi thông điệp **queryHit** tới A (thông điệp này đi theo đúng tuyến đường cũ nhưng theo chiều ngược lại, từ J, qua H, qua E, B rồi đến A). Sau đó A có thể kết nối trực tiếp đến J để tải dữ liệu về theo giao thức HTTP.

Khuôn dạng thông điệp của giao thức Gnutella

Các thông điệp của giao thức Gnutella được chia thành 5 kiểu: **Ping, Pong, Query, QueryHit,** và **Push**.

Ping: Được sử dụng để phát hiện các nút trên mạng. Một nút khi nhận được một thông điệp Ping có thể trả lời bằng một hoặc nhiều thông điệp Pong.

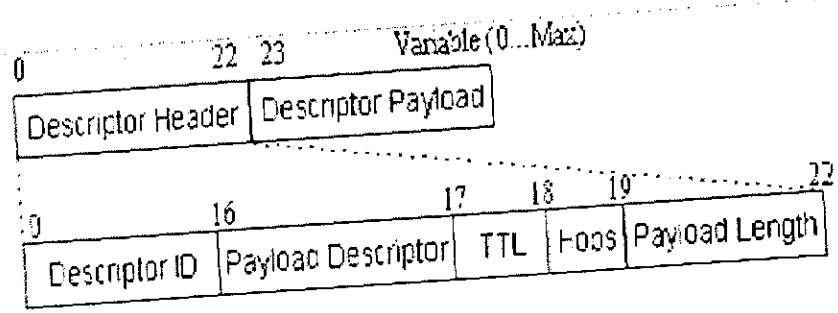
Pong. Thông điệp trả lời cho Ping. Chứa địa chỉ và các thông tin của nút (chẳng hạn danh sách các file chia sẻ).

Query. Là thông điệp phục vụ mục đích tìm kiếm trên mạng. Nếu nhận được truy vấn mà có dữ liệu đáp ứng được, nút gửi thông điệp QueryHit trả lời.

QueryHit. Là thông điệp trả lời thông điệp Query. Thông điệp này cung cấp các thông tin cần thiết để phía tìm kiếm có thể tải file.

Push. Được các nút đằng sau firewall sử dụng khi muốn chuyển dữ liệu cho nút khác.

Các thông tin tiêu đề



Hình 2.31 Khuôn dạng tổng quát của thông điệp GNUTELLA

Hình 2.31 minh họa tiêu đề gồm 5 trường của một thông điệp Gnutella tổng quát:

- **Descriptor ID (Định danh thông điệp)** xác định duy nhất một thông điệp trên mạng.
- **Payload Descriptor (Dữ liệu)** Xác định kiểu thông điệp (0x00 cho Ping, 0x01 cho Pong, 0x40 cho Push, 0x80 cho Query, and 0x81 cho QueryHit).
- **TTL** Số chặng mà thông điệp có thể lan tỏa trên mạng.
- **Hops**: Số chặng mà thông điệp đã đi qua (tổng của giá trị này với giá trị TTL là giá trị TTL khởi đầu).
- **Payload Length**: Độ dài của trường dữ liệu.

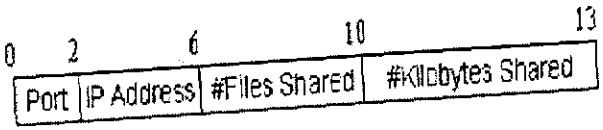
Mô tả trường dữ liệu

Có năm kiểu trường dữ liệu khác nhau (ứng với 5 kiểu thông điệp).

1. Ping

Thông điệp Ping không có trường dữ liệu.

2. Pong



Hình 2.32 Khuôn dạng thông điệp PONG

Dữ liệu trong thông điệp Pong có bốn phần (Hình 2.32).Thông điệp Pong được gửi để trả lời thông điệp Ping. Một thông điệp Ping có thể có nhiều thông điệp Pong trả lời.

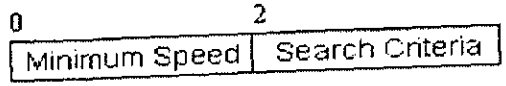
Port: số hiệu cổng host chấp nhận kết nối.

IP Address: địa chỉ IP của nút.

#Files Shared: Số lượng file mà host chia sẻ.

#Kilobytes Shared: Số lượng KB host đã chia sẻ.

3. Query

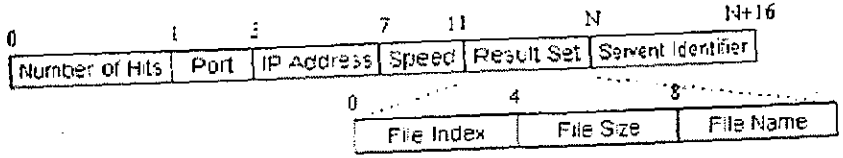


Hình 2.33 Khuôn dạng thông điệp Query

Là thông điệp truy vấn, gồm hai phần: Minimum Speed và Search Criteria.

Minimum Speed: Tốc độ cực tiểu (đơn vị KB/s) mà nút có thể đáp ứng với thông điệp.

Search Criteria: Tiêu chí tìm kiếm, độ dài của trường này bị giới hạn bởi giá trị trường Payload Length trong tiêu đề thông điệp.



Hình 2.34 Khuôn dạng thông điệp QueryHit

4. Thông điệp QueryHit trả lời cho thông điệp Query. Thông điệp này được nút có thể đáp ứng yêu cầu gửi cho nút có yêu cầu. Định danh thông điệp trong tiêu đề thông điệp QueryHit trùng với định danh thông điệp của thông điệp Query tương ứng. Điều này cho phép nút nhận có thể xác định kết quả trả lời ứng với câu hỏi truy vấn nào.

Number of Hits: Số lượng các câu trả lời trong Result Set.

Port: Số hiệu cổng mà nút có thể chấp nhận các kết nối.

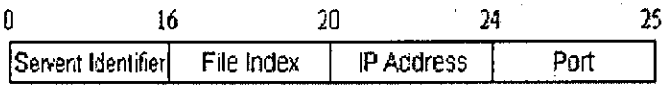
IP Address: địa chỉ IP của nút tương ứng

Speed: tốc độ (KB/s) của nút.

Result Set: Tập hợp các kết quả thỏa mãn câu truy vấn. Tập hợp này bao gồm các bản ghi liên tục nhau, mỗi bản ghi gồm tên file, kích thước file và đường dẫn đến file.

Servent Identifier: là chuỗi 16-byte xác định nút duy nhất trên mạng.

5. Push (0x40)



Hình 2.35 Khuôn dạng thông điệp PUSH

Thông điệp Push được sử dụng bởi nút yêu cầu đề nghị nút có dữ liệu nhưng ở phía sau firewall khởi tạo kết nối trước.

Servent Identifier: chuỗi 16-byte xác định nút được đề nghị chuyển file có đường dẫn đặt ở File_Index. Định danh này phải giống với định danh trong thông điệp QueryHit tương ứng.

File Index: xác định file được đề nghị gửi.

Port: Số hiệu cổng mà nút có thể chấp nhận các kết nối.

IP Address: địa chỉ IP của nút tương ứng.

2.7 LẬP TRÌNH SOCKET

Phần này giới thiệu quá trình phát triển ứng dụng mạng. Trong các phần trước, chúng ta đã biết phần lớn các ứng dụng mạng gồm hai phía: Client và Server. Hai phía trao đổi với nhau bằng cách gửi và nhận các thông điệp qua Socket. Phần này sẽ giới thiệu các bước chính trong quá trình

phát triển các ứng dụng mạng. Sau đó chúng ta sẽ giới thiệu một chương trình Web server rất đơn giản.

2.7.1 Các hàm thao tác trên Socket

Đầu tiên, chương trình phía server phải chạy và lắng nghe trên một cổng nào đó để đợi phía client kết nối tới. Nếu mọi việc diễn ra bình thường (kết nối thành công), cả hai phía đều có hai thể hiện của lớp Socket và dữ liệu sẽ được truyền qua hai socket này.

Để mở socket MyClient, phía client sẽ thực hiện khai báo và sau đó sử dụng phương thức tạo mới Socket như sau:

```
Socket MyClient;
try {
        MyClient = new Socket("Machine name", PortNumber);

catch (IOException e) {
        System.out.println(e);
```

Trong đó **Machine name** là tên (hoặc địa chỉ IP) của máy tính server, và **PortNumber** là số hiệu cổng mà tiến trình server đang chạy "lắng nghe" (đợi kết nối từ client). Chú ý các port từ 0-1023 được sử dụng cho các ứng dụng đặc biệt (HTTP, FTP, SMTP) nên bạn hãy lựa chọn cổng có số hiệu lớn hơn 1023.

Phía server sẽ mở socket bằng cách sau:

```
ServerSocket MyService;
try {
        MyServerice = new ServerSocket(PortNumber);

catch (IOException e) {
        System.out.println(e);
```

Phía server cần phải khởi tạo đối tượng của lớp **ServerSocket** với mục đích lắng nghe và chấp nhận kết nối đến từ client.

```
Socket clientSocket = null;
```

```
try {
        serviceSocket = MyService.accept();
}
catch (IOException e) {
        System.out.println(e);
}
```

Tạo đối tượng Input Stream ở phía client

Phía client phải sử dụng lớp DataInputStream tạo input với mục đích nhận thông điệp trả lời từ server:

```
DataInputStream input;
try {
        input = new DataInputStream(MyClient.getInputStream());
}
catch (IOException e) {
        System.out.println(e);
}
```

Lớp **DataInputStream** cho phép bạn đọc từng dòng văn bản hoặc các đối tượng thuộc kiểu cơ bản trong ngôn ngữ lập trình Java (chẳng hạn read, readChar, readInt, readDouble, và readLine).

Ở phía server, cũng sử dụng lớp **DataInputStream** để nhận thông điệp từ client:

```
DataInputStream input;
try {
    input = new DataInputStream(serviceSocket.getInputStream());
}
catch (IOException e) {
  System.out.println(e);
}
```

Tạo output stream

Phía client có thể sử dụng lớp **PrintStream** hoặc **DataOutputStream** của java.io để gửi thông điệp cho server.

```
PrintStream output;
try {
        output = new PrintStream(MyClient.getOutputStream());
```



```
}
catch (IOException e) {
        System.out.println(e);
}
```

Lớp **PrintStream** có phương thức để hiển thị các kiểu dữ liệu cơ bản của Java dưới dạng văn bản (Chẳng hạn các phương thức Write hoặc Println), còn lớp **DataOutputStream** có thể được sử dụng như sau:

```
DataOutputStream output;
try {
        output = new DataOutputStream(MyClient.getOutputStream());
}
catch (IOException e) {
        System.out.println(e);
}
```

Lớp **DataOutputStream** cũng cho phép viết các kiểu dữ liệu cơ bản trong Java (ví dụ phương thức writeBytes).

Lớp **PrintStream** được phía server sử dụng để gửi thông điệp cho client.

```
PrintStream output;
try {
        output = new PrintStream(serviceSocket.getOutputStream());

catch (IOException e) {
        System.out.println(e);
```

Đóng kết nối

Phải đóng output và input stream trước khi đóng socket.

Ở phía client:

```
try {
        output.close();
        input.close();
        MyClient.close();

atch (IOException e) {
        System.out.println(e);
```

```
}
```

Ở phía server:

```
try {
        output.close();
        input.close();
        serviceSocket.close();
        MyService.close();
}
catch (IOException e) {
        System.out.println(e);
}
```

2.7.2 Ví dụ một chương trình client/server đơn giản

Ví dụ đơn giản này minh họa một ứng dụng server/client đơn giản. Client sẽ gửi cho Server một chuỗi, Server sẽ gửi phản hồi chuỗi này cho Client. Trong ví dụ này sử dụng địa chỉ máy tính là **localhost** (ứng với địa chỉ IP **127.0.0.1**), khi đó cả server và client chạy trên cùng một máy tính.

Đây là mã nguồn của Server.java:

```
import java.lang.*;
import java.io.*;
import java.net.*;
class     Server {
          public static void main(String args[]) {
                    String data = "Toobie ornaught toobie";
                    try {
                         ServerSocket srvr = new ServerSocket(1234);
                         Socket skt = srvr.accept();
                         System.out.print("Server has connected!\n");
                              PrintWriter out =
                              new PrintWriter(skt.getOutputStream(),true);
                              System.out.print("Sending string: '" + data +
"'\n");
                              out.print(data);
                              out.close();
```



```
                              skt.close();
                              srvr.close();
                    }
                    catch(Exception e) {
                              System.out.print("Whoops!     It     didn't
work!\n");
                    }
          }
}
```

Khối chương trình chính nằm trong khối **try{}**. Đối tượng **ServerSocket** được khởi tạo để lắng nghe cổng 1234. Sau khi khởi tạo, đối tượng Socket sử dụng phương thức **accept()** của lớp **ServerSocket** để đợi client kết nối đến. Phương thức này trả lại một đối tượng của lớp **Socket (skt)** – đối tượng này đóng vai trò "đầu mút" truyền thông ở phía server. **skt.getOutputStream()** trả lại output stream qua đó server có thể gửi thông điệp tới client, và **skt.getInputStream()** trả lại input stream qua đó server có thể nhận thông điệp từ client. Ví dụ này tạo ra đối tượng **PrintWriter** sử dụng output stream để đơn giản hóa việc kết xuất và gửi dữ liệu (đặt trong biến data) tới client (**out.print(data)**).

Đây là mã chương trình **Client.java**:

```
import java.lang.*;
import java.io.*;
import java.net.*;
class Client {
          public static void main(String args[]) {
          try {
                    Socket skt = new Socket("localhost", 1234);
                    BufferedReader in = new BufferedReader(new
                    InputStreamReader(skt.getInputStream()));
                    System.out.print("Received string: '");
                    while (!in.ready()) {}
                              System.out.println(in.readLine());
                              System.out.print("'\n");
                              in.close();
                    }
          catch(Exception e) {
                    System.out.print("Whoops! It didn't work!\n");
          }
          }
}
```

Chúng ta thấy khối chương trình chính của client cũng nằm trong khối **try{}**. Một đối tượng của lớp Socket được tạo ra để kết nối tới server. Ở đây, client sẽ kết nối tới server có địa chỉ localhost ở cổng 1234 – cổng mà server đang lắng nghe. Sau khi được tạo ra, socket cũng hoạt động tương tự như đối tượng lớp ServerSocket trong file Server.java. Tiếp theo, input stream được tạo ra và là dữ liệu để khởi tạo một đối tượng thuộc lớp BufferedReader. Dữ liệu được đọc từ lớp này và hiển thị lên màn hình.

2.7.3 Web server đơn giản

Trong phần này chúng ta sẽ phân tích và phát triển ứng dụng Web server rất đơn giản. Chương trình của chúng ta khi thực thi sẽ đợi yêu cầu từ trình duyệt và nếu có đối tượng được yêu cầu sẽ gửi đối tượng cho trình duyệt. Đối với ứng dụng Web, trình duyệt (client) luôn là phía thiết lập kết nối trước và sau đó gửi yêu cầu HTTP. Phía server không có nhiệm vụ kết nối với client, tuy nhiên cả hai phía đều có khả năng đóng kết nối. Chẳng hạn, khi NSD nhấn nút Stop trên trình duyệt thì có thể khiến trình duyệt kết thúc việc tải file về và sau đó đóng kết nối TCP lại.

HTTP Request

Như chúng ta đã nói trong phần 2.2, yêu cầu HTTP có ba thành phần chính

- Phương thức / Phiên bản giao thức
- Tiêu đề của thông điệp yêu cầu
- Thân yêu cầu

Mỗi yêu cầu HTTP có thể sử dụng một trong các phương thức sau: GET, POST, HEAD, OPTIONS, PUT, DELETE, TRACE, trong đó GET và POST là thông dụng nhất. Sau phần phương thức là URL – địa chỉ của tài nguyên, địa chỉ này thường là địa chỉ tương đối so với thư mục root của Web server. Phần tiêu đề thường chứa các thông tin về môi trường của trình duyệt (tên, phiên bản trình duyệt; hệ điều hành của máy client...). Các tiêu đề cách nhau bằng ký tự xuống dòng (CR-LF).

HTTP Response

Thông điệp trả lời của HTTP gồm 3 phần sau

- Trạng thái giao thức / Mô tả trạng thái
- Tiêu đề của thông điệp
- Thân thông điệp

Ứng dụng Web server trình bày ở đây gồm ba lớp chính:

- **HttpServer**
- **Request**
- **Response**

Chương trình chính được cài đặt trong lớp **HttpServer**. Phương thức main() của lớp này sẽ tạo ra một đối tượng **HttpServer** và sau đó gọi phương thức await để đợi một yêu cầu từ client (trên một cổng nào đó), xử lý yêu cầu và sau đó gửi kết quả cho client. Đối tượng **HtttpServer** sẽ đợi cho đến khi nhận được lệnh **Shutdown**.

Web server này rất đơn giản, chỉ gửi các đối tượng tĩnh (các file HTML, file ảnh). Nó không chấp nhận tiêu đề (chẳng hạn ngày tháng, cookie) từ trình duyệt.

Lớp HttpServer

Lớp **HttpServer** biểu diễn đối tượng Web server và có thể phục vụ các yêu cầu đối tượng tĩnh là các file nằm trong thư mục được xác định bởi biến tĩnh **WEB_ROOT** và các thư mục con của thư mục này. **WEB_ROOT** được khởi tạo như sau:

```
public static final String WEB_ROOT =
        System.getProperty("user.dir") + File.separator  + "Webroot";
```

Để yêu cầu đối tượng tĩnh, đánh địa chỉ kiểu như sau trong thanh địa chỉ của trình duyệt:

```
http://machineName:port/staticResource
```

Nếu yêu cầu này được gửi từ một máy tính khác máy tính chạy chương trình Web server, thì machimeName là địa chỉ IP của máy tính chạy Web server; còn nếu hai chương trình chạy trên cùng một máy tính thì machineName có thể là **localhost** hoặc **127.0.0.1**. Port là **8080** và

staticResource là tên file được yêu cầu (file này phải nằm trong thư mục WEB_ROOT).

Ví dụ, nếu hai chương trình chạy trên cùng một máy tính và trình duyệt muốn yêu cầu file **index.html** nằm trong thư mục **WEB_ROOT** thì gõ địa chỉ sau:

http://localhost:8080/index.html

Để ngừng hoạt động của server, có thể từ trình duyệt gửi lệnh **SHUTDOWN** cho server

http://localhost:8080/SHUTDOWN

Lệnh **SHUTDOWN** được cài đặt bằng một biến tĩnh bên trong lớp **HttpServer**:

```
private static final String SHUTDOWN_COMMAND = "/SHUTDOWN";
```

Chúng ta xét phương thức await sau đây:

```
public void await() {
        ServerSocket serverSocket = null;
        int port = 8080;
        try {
                serverSocket =  new ServerSocket(port, 1,
                InetAddress.getByName("127.0.0.1"));
        }
        catch (IOException e) {
                e.printStackTrace();
                System.exit(1);
        }

  // Vòng lặp, được sử dụng để đợi kết nối từ server.
        while (!shutdown) {
                Socket socket = null;
                InputStream input = null;
                OutputStream output = null;
                try {
                        socket = serverSocket.accept();
                        input = socket.getInputStream();
                        output = socket.getOutputStream();

                        // Tạo và sau đó phân tích đối tượng Request
                        Request request = new Request(input);
```



```
                        request.parse();

                        // Tạo đối tượng Response
                        Response response = new
Response(output);
                        response.setRequest(request);
                        response.sendStaticResource();

                        // Đóng kết nối
                        socket.close();

                        //Kiểm tra xem yêu cầu đến từ client có là
                        lệnh shutdown hay không

        request.getUri().equals(SHUTDOWN_COMMAND);
                }
                catch (Exception e) {
                        e.printStackTrace();
                        continue;
                }
        }
```

Phương thức **await** bắt đầu bằng cách tạo một đối tượng thuộc lớp **ServerSocket** và đặt nó trong vòng lặp **while**.

```
verSocket= new ServerSocket(port, 1, InetAddress.getByName("127.0.0.1"));

ợi kết nối
ile (!shutdown) {
```

Đoạn mã bên trong vòng lặp while bắt đầu bằng phương thức **ept** của lớp **ServerSocket**, trả lại kết quả là yêu cầu HTTP nhận được ổng 8080:

```
ket = serverSocket.accept();
```

Khi nhận được yêu cầu, phương thức **await** tạo ra đối tượng io.InputStream và java.io.OutputStream từ đối tượng Socket i.

```
ut = socket.getInputStream();
put = socket.getOutputStream();
```

Phương thức **await** cũng tạo ra đối tượng **Request** và gọi phương thức **parse** để phân tích yêu cầu HTTP.

```
Request request = new Request(input);
request.parse();
```

Kế tiếp, tạo ra đối tượng **Response** và khởi tạo cho đối tượng này nhờ đối tượng Request và sau đó sử dụng phương thức **sendStaticResource** để gửi đối tượng.

```
Response response = new Response(output);
response.setRequest(request);
response.sendStaticResource();
```

Cuối cùng, đóng Socket và sử dụng phương thức **getUri** của lớp **Request** để kiểm tra xem yêu cầu có phải là lệnh kết thúc **SHUTDOWN** hay không.

```
// Đóng socket
socket.close();
//Kiểm tra có phải là lệnh SHUTDOWN không
shutdown = request.getUri().equals(SHUTDOWN_COMMAND);
```

Phân tích lớp Request

Lớp **Request** thực hiện việc biểu diễn và xử lý yêu cầu HTTP. Đối tượng của lớp này được khởi tạo từ đối tượng của lớp Socket để thực hiện việc truyền thông với Client.

Lớp **Request** có hai phương thức chính: **parse** và **getUri**. Phương thức **parse** phân tích dữ liệu thô trong yêu cầu HTTP. Nó sử dụng phương thức **parseUri** để lấy ra URL. Phương thức **parseUri** đặt giá trị URL vào biến uri. Phương thức **getUri** trả lại URL của thông điệp yêu cầu HTTP.

Chú ý rằng thông điệp HTTP yêu cầu gồm ba phần: dòng yêu cầu, các tiêu đề và thân thông điệp. Tuy nhiên trong ví dụ này, chúng ta chỉ quan tâm đến dòng yêu cầu (dòng này gồm ba phần: từ khóa là tên phương thức yêu cầu, tên đối tượng và cuối cùng là phiên bản giao thức HTTP; các phần cách nhau bằng dấu cách). Ví dụ

```
GET /index.html HTTP/1.1
```

là yêu cầu file **index.html** bằng phương thức **GET**.

Phương thức **parse** đọc toàn bộ luồng byte từ socket InputStream, sau đó chuyển cho đối tượng và sau đó lưu giữ luồng byte này trong một bộ đệm. Bộ đệm này sau đó được dùng để khởi tạo cho đối tượng **request** của

lớp **StringBuffer**. Đối tượng này sau đó được chuyển sang kiểu **String** để phương thức **parseUri** có thể phân tích

```
public void parse() {
        // Đọc các ký tự từ socket
        StringBuffer request = new StringBuffer(2048);
        int i;
        byte[] buffer = new byte[2048];
        try {
                i = input.read(buffer);
        }
        catch (IOException e) {
                e.printStackTrace();
                i = -1;
        }
        for (int j=0; j<i; j++) {
                request.append((char) buffer[j]);
        }

        System.out.print(request.toString());
        uri  = parseUri(request.toString());
```

Sau đó phương thức **parseUri** sẽ lấy địa chỉ URL từ dòng yêu cầu bằng cách trích chuỗi nằm giữa hai dấu cách đầu tiên.

```
rivate String parseUri(String requestString) {
        int index1, index2;
        index1 = requestString.indexOf(' ');
        if (index1 != -1) {
                index2 = requestString.indexOf(' ', index1 + 1);
                 if (index2 > index1)
                        return  requestString.substring(index1  +  1,
ndex2);
        }
        return null;
```

hân tích lớp Response

Lớp **Response** biểu diễn thông điệp HTTP trả lời, có thể sử dụng ối tượng thuộc lớp **OutputStream** để khởi tạo như sau:

```
ublic Response(OutputStream output) {
 this.output = output;
```

Đối tượng **Response** sẽ được lớp **HttpServer** khởi tạo bằng việc gán đối tượng **OutputStream** được tạo ra từ socket.

Lớp **Response** có hai phương thức: **setRequest** và **sendStaticResource**. Phương thức **setRequest** sẽ gán đối tượng **Request** cho đối tượng **Response**.

```
public void setRequest(Request request) {
  this.request = request;
}
```

Nhiệm vụ của phương thức **sendStaticResource** là gửi đối tượng tĩnh, chẳng hạn file HTML

```
public void sendStaticResource() throws IOException {
        byte[] bytes      = new byte[BUFFER_SIZE];
        FileInputStream fis = null;
        try {
                File file      =    new    File(HttpServer.WEB_ROOT,
request.getUri());
                if (file.exists()) {
                        fis   = new FileInputStream(file);
                        int ch = fis.read(bytes, 0, BUFFER_SIZE);
                        while (ch != -1) {
                        output.write(bytes, 0, ch);
                        ch = fis.read(bytes, 0, BUFFER_SIZE);
                        }
                }
                else { Không tìm thấy file
                        String errorMessage = "HTTP/1.1 404 File
                        Not Found\r\n"      +      "Content-Type:
                        text/html\r\n" +
                        "Content-Length: 23\r\n" + "\r\n" +
                        "<h1>File Not Found</h1>";
                        output.write(errorMessage.getBytes());
                }
        }
        catch (Exception e) {
                System.out.println(e.toString() );
        }
        finally {
                if (fis != null)
                fis.close();
        }
}
```


Phương thức **sendStaticResource** rất đơn giản, sử dụng tham số là đường dẫn đầy đủ của thư mục gốc (**WEB_ROOT**) và tên file yêu cầu khởi tạo đối tượng file thuộc lớp **java.io.File**.

```
File file = new File(HttpServer.WEB_ROOT, request.getUri());
```

Đầu tiên sẽ kiểm tra xem file như vậy có tồn tại không. Nếu tồn tại, phương thức **sendStaticResource** sẽ tạo ra một đối tượng thuộc lớp java.io.FileInputStream từ đối tượng **File**. Kế tiếp sẽ sử dụng phương thức **read** của lớp **FileInputStream** để ghi luồng byte lên **OutputStream**. Nội dung của đối tượng tĩnh được gửi cho trình duyệt dưới dạng dữ liệu thô.

```
If (file.exists()) {
        fis   = new FileInputStream(file);
        int ch = fis.read(bytes, 0, BUFFER_SIZE);
        while (ch != -1) {
                output.write(bytes, 0, ch);
                ch = fis.read(bytes, 0, BUFFER_SIZE);
        }
}
```

Nếu file không tồn tại, phương thức **sendStaticResource** gửi thông báo lỗi cho trình duyệt

```
String errorMessage = "HTTP/1.1 404 File Not Found\r\n" +
 "Content-Type: text/html\r\n" +
 "Content-Length: 23\r\n" +
 "\r\n" +
 "<h1>File Not Found</h1>";

output.write(errorMessage.getBytes());
```

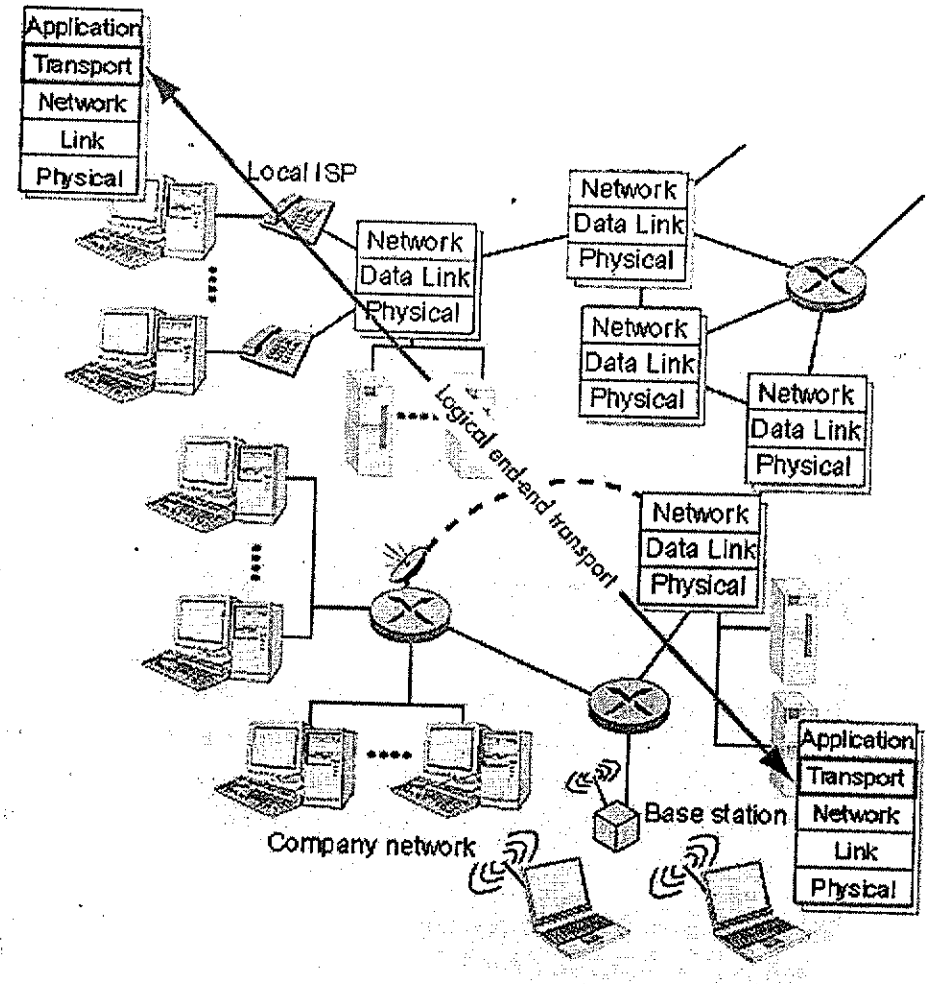
Chương 3
TẦNG GIAO VẬN

3.1 DỊCH VỤ VÀ NGUYÊN TẮC CỦA TẦNG GIAO VẬN

Nằm giữa tầng ứng dụng và tầng mạng, tầng giao vận 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. Chương này nghiên cứu tất cả dịch vụ của tầng giao vận cũng như các nguyên tắc cơ bản thực hiện điều này theo nhiều cách tiếp cận khác nhau. Chúng ta sẽ xem cách thức dịch vụ này được cài đặt trong các giao thức. Tầng giao vận của Internet có hai giao thức quan trọng là TCP và UDP.

Hai chương trước đã nói về vai trò và những dịch vụ mà tầng giao vận cung cấp, vậy cho đến bây giờ, chúng ta đã biết gì về tầng giao vận?

Giao thức tầng giao vận cung cấp một kênh truyền logic (ảo) giữa các tiến trình ứng dụng chạy trên máy tính khác nhau. Gọi là logic vì không tồn tại một đường truyền vật lý thực sự giữa hai tiến trình. Các tiến trình ứng dụng sẽ sử dụng đường truyền ảo này để trao đổi thông điệp mà không phải bận tâm về cơ sở hạ tầng của môi trường vật lý thực sự. Hình 3.1 minh hoạ điều này.



Hình 3.1 Tầng giao vận cung cấp dịch vụ truyền thông logic cho các tiến trình ứng dụng

Trên Hình 3.1, tầng giao vận nằm trên các thiết bị đầu cuối chứ không phải ở các router. Router hoạt động ở tầng mạng.

Ở phía gửi, thực thể giao vận chèn thông điệp mà nó nhận được từ tiến trình ứng dụng vào các 4-PDU (là đơn vị dữ liệu của giao thức tầng giao vận – Protocol Data Unit). Công việc được thực hiện bằng cách chia thông điệp thành nhiều đoạn nhỏ, bổ sung vào đầu mỗi đoạn tiêu đề của tầng giao vận để tạo ra gói dữ liệu của tầng giao vận (4-PDU). Sau đó tầng giao vận truyền gói dữ liệu (4-PDU) xuống tầng mạng, tại đây mỗi gói này được đặt

trong gói dữ liệu của tầng mạng (3-PDU). Ở phía nhận, tầng giao vận nhận gói dữ liệu từ tầng mạng, loại bỏ phần tiêu đề của gói dữ liệu 4-PDU, ghép chúng lại thành một thông điệp hoàn chỉnh và chuyển cho tiến trình ứng dụng nhận.

Trên mạng máy tính có thể có nhiều giao thức giao vận khác nhau, cung cấp cho ứng dụng các dịch vụ với chất lượng khác nhau.

Tất cả giao thức tầng giao vận đều cung cấp dịch vụ **dồn kênh (multiplex)** và **phân kênh (demultiplex)**, điều này sẽ nói cụ thể trong các phần sau. Như đã nói trong phần 2.1, ngoài dịch vụ dồn kênh/ phân kênh, tầng giao vận còn có thể cung cấp các dịch vụ khác cho tiến trình ứng dụng như truyền dữ liệu tin cậy, đảm bảo băng thông hay giới hạn độ trễ.

3.1.1 Quan hệ giữa tầng giao vận và tầng mạng

Tầng giao vận nằm ở trên tầng mạng. Nếu giao thức tầng giao vận cung cấp đường truyền logic giữa tiến trình chạy trên các máy tính khác nhau, thì giao thức tầng mạng cung cấp đường truyền giữa các máy tính. Điểm khác biệt nhỏ này tuy khó nhận biết nhưng rất quan trọng, xét ví dụ dưới đây.

Giả sử có hai nhà: một ở Hà Nội, một ở Huế trong mỗi nhà có 12 đứa trẻ là anh em họ với nhau. Hàng tuần chúng trao đổi thư cho nhau, mỗi thư được đặt trong một phong bì riêng và được dịch vụ bưu chính gửi đi theo địa chỉ ghi trên phong bì. Hàng tuần mỗi nhà sẽ nhận 144 lá thư từ nhà bên kia (bọn trẻ có thể tiết kiệm được tiền nếu chúng sử dụng email). Ở mỗi nhà có một đứa trẻ chịu trách nhiệm thu thập và phân phát thư - An trong nhà phía tây, Bình trong nhà phía đông. Mỗi tuần, An lấy thư từ bọn trẻ trong nhà mình và chuyển cho nhân viên bưu cục - người thường xuyên ghé qua nhà để lấy và chuyển thư. Khi nhận thư từ nhân viên bưu tá, An chuyển tiếp thư cho người nhận. Bình cũng sẽ thực hiện công việc tương tự.

Trong ví dụ trên, dịch vụ bưu chính cung cấp đường truyền logic giữa hai nhà - chuyển thư từ nhà này đến nhà kia, chứ không phải từ người

này đến người kia. Còn An và Bình cung cấp đường truyền logic giữa từng người trong hai nhà. Đối với lũ trẻ, An và Bình là dịch vụ chuyển thư mặc dù An và Bình chỉ là một phần (phần đầu mút) của cả hệ thống chuyển thư. Qua ví dụ này ta hiểu được quan hệ giữa tầng giao vận và tầng mạng:

Máy tính (hay thiết bị đầu cuối) = Ngôi nhà.

Tiến trình = Từng người trong ngôi nhà.

Thông điệp ứng dụng = Thư trong phong bì.

Giao thức tầng mạng = Dịch vụ bưu chính (gồm nhân viên bưu chính).

Giao thức tầng giao vận = An và Bình.

Trong ví dụ trên, An và Bình thực hiện công việc phân phát thư tại chính ngôi nhà của chúng, nhưng không thực hiện những việc như sắp xếp thư tại các bưu cục (là các trạm trung chuyển trên đường đi) hay gửi thư từ bưu cục này tới bưu cục khác. Tương tự, giao thức tầng giao vận chỉ hoạt động ở các thiết bị đầu cuối. Tại thiết bị đầu cuối, giao thức tầng giao vận chuyển dữ liệu từ tiến trình ứng dụng xuống tầng mạng và ngược lại nhưng không biết thông điệp được truyền đi như thế nào trong tầng mạng. Trên hình 3.1, các router không xử lý bất kỳ thông tin tiêu đề nào mà tầng giao vận chèn vào bên cạnh thông điệp ứng dụng.

Giả sử An và Bình đi vắng, Hạnh và Phúc làm thay. Nhưng thật tiếc hai đứa trẻ này còn quá nhỏ, không làm việc được cẩn thận như An và Bình. Chúng làm mất thư. Tương tự như vậy, mạng máy tính có thể có nhiều giao thức giao vận, mỗi giao thức cung cấp các dịch vụ với chất lượng khác nhau cho chương trình ứng dụng.

Dịch vụ mà An và Bình cung cấp phụ thuộc vào dịch vụ của bưu điện. Ví dụ nếu bưu điện không đảm bảo thời gian chuyển thư giữa hai nhà thì An và Bình cũng sẽ không đảm bảo được thời gian chuyển thư giữa những người trong hai nhà. Tương tự, dịch vụ của giao thức tầng giao vận cung cấp sẽ phụ thuộc vào dịch vụ của tầng mạng bên dưới. Nếu giao thức tầng mạng không đảm bảo thời gian trễ hay đảm bảo về băng thông cho gói dữ liệu 4-PDU trong quá trình gửi giữa các máy tính, thì giao thức tầng giao vận cũng không thể cung cấp những dịch vụ này khi gửi thông điệp giữa các tiến trình ứng dụng.

Tuy nhiên, tầng giao vận vẫn có thể cung cấp những dịch vụ mà tầng mạng không cung cấp. Những dịch vụ như thế được nghiên cứu ngay trong chương này, ví dụ giao thức tầng giao vận cung cấp dịch vụ truyền dữ liệu tin cậy cho tầng ứng dụng ngay cả khi tầng mạng không đáng tin cậy - làm mất, gửi lỗi hay gửi trùng lặp dữ liệu. Một dịch vụ khác sẽ nghiên cứu trong chương 7 (An ninh mạng) là khả năng mã hoá thông điệp của tầng giao vận để đảm bảo thông điệp không bị đọc trộm, trong khi tầng mạng không thực hiện được điều này.

3.1.2 Tổng quan về tầng giao vận trong Internet

Trong mạng Internet hay mạng TCP/IP có hai giao thức ở tầng giao vận: UDP và TCP. **UDP** (User Datagram Protocol) cung cấp dịch vụ truyền không tin cậy, không hướng nối và **TCP** (Transmission Control Protocol) cung cấp dịch vụ tin cậy, hướng nối cho ứng dụng. Người viết ứng dụng phải lựa chọn một trong hai giao thức này cho ứng dụng của mình.

Để đơn giản, trong mô hình Internet ta coi 4-PDU là một segment. Tuy vậy, nhưng trong các khuyến nghị RFC thì 4-PDU được coi là **segment** đối với TCP và **datagram** đối với UDP. Nói chung thuật ngữ **datagram** thường sử dụng cho PDU ở tầng mạng nhưng trong một quyển sách nhập môn như thế này, nói chung ít xảy ra nhầm lẫn khi sử dụng thuật ngữ segment cho cả TCP PDU và UDP PDU.

Trước khi tiếp tục, chúng ta nói qua về tầng mạng của Internet (Tầng mạng sẽ được nghiên cứu chi tiết trong chương 4). Giao thức của tầng mạng là IP (Internet Protocol). IP cung cấp đường truyền logic giữa các máy tính và mô hình dịch vụ của nó theo kiểu **cố gắng tối đa (best effort delivery service)**. Nghĩa là IP cố gắng gửi các segment giữa các máy tính – hay thiết bị đầu cuối khác nhau - với hết khả năng của mình nhưng không đảm bảo điều này. Nói cụ thể hơn, IP không đảm bảo về thứ tự truyền, về tính toàn vẹn của dữ liệu trong segment. Chính vì thế người ta xem IP là dịch vụ không tin cậy. Mỗi máy tính có một địa chỉ IP xác định. Trong chương này ta chỉ cần biết mỗi máy tính cần có một địa chỉ IP xác định duy nhất.

Nhiệm vụ chính của UDP và TCP là mở rộng dịch vụ IP - truyền dữ liệu giữa hai thiết bị đầu cuối - thành dịch vụ truyền dữ liệu giữa hai tiến trình chạy trên thiết bị đầu cuối. Việc mở rộng từ truyền dữ liệu giữa các máy tính (host-to-host) đến truyền dữ liệu giữa các tiến trình (process-to-process) được gọi là quá trình dồn kênh (multiplex) và phân kênh (demultiplex). Vấn đề này sẽ nghiên cứu ở phần sau. UDP và TCP kiểm soát tính toàn vẹn (hay tính đúng đắn) của dữ liệu nhờ trường phát hiện lỗi đặt trong tiêu đề gói dữ liệu. UDP chỉ cung cấp dịch vụ phân phối dữ liệu giữa hai tiến trình và kiểm tra lỗi. Tương tự IP, UDP là dịch vụ không tin cậy, không đảm bảo dữ liệu được truyền đi một cách đúng đắn giữa các tiến trình. UDP được trình bày kỹ trong phần 3.1.

Ngoài phân kênh, dồn kênh, TCP còn cung cấp một số dịch vụ khác cho ứng dụng. Dịch vụ đầu tiên và quan trọng nhất là truyền dữ liệu tin cậy (reliable data transfer). Các cơ chế điều khiển lưu lượng, đánh số thứ tự, số thứ tự biên nhận, bộ định thời sẽ giúp TCP đảm bảo dữ liệu được truyền từ tiến trình gửi đến tiến trình nhận chính xác và đúng thứ tự. Như vậy giao thức TCP đã biến dịch vụ truyền không tin cậy giữa các thiết bị đầu cuối (IP) thành dịch vụ truyền dữ liệu tin cậy giữa các tiến trình.

Giao thức cung cấp dịch vụ truyền dữ liệu tin cậy và kiểm soát tắc nghẽn rất phức tạp. Các phần từ 3.4 đến 3.8 trình bày nguyên tắc chung của các dịch vụ trên và giao thức TCP. Cách tiếp cận của chương này là giới thiệu xen kẽ các nguyên lý cơ bản với giao thức TCP. Ví dụ chúng ta nói tổng quan cách thức cung cấp dịch vụ truyền dữ liệu tin cậy sau đó mới nghiên cứu TCP thực hiện điều này như thế nào. Chúng ta sẽ bắt đầu bằng công việc dồn kênh/phân kênh với dữ liệu từ tầng ứng dụng.

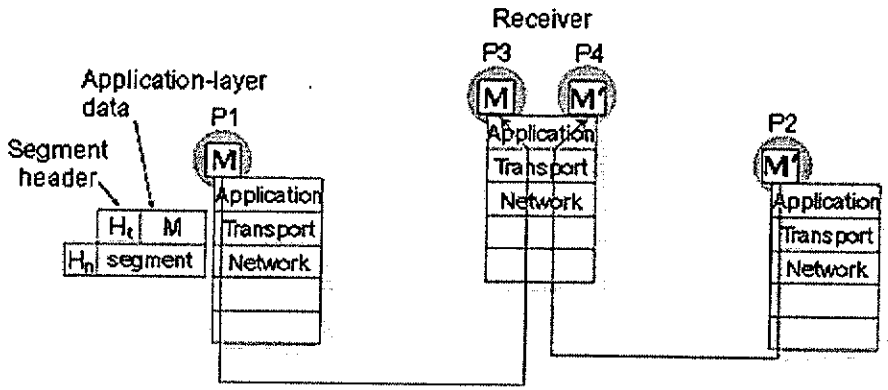
3.2 DỊCH VỤ DỒN KÊNH, PHÂN KÊNH

Phần này chúng ta sẽ nghiên cứu về công việc dồn kênh và phân kênh trong mạng. Để đơn giản, chúng ta chỉ nói đến các dịch vụ của tầng giao vận trong mô hình Internet. Tuy nhiên cần nhấn mạnh rằng - đây là dịch vụ cần thiết đối với tất cả các mô hình kết nối mạng.

Mặc dù dồn kênh/ phân kênh không phải là một trong những dịch vụ quan trọng nhất của tầng giao vận, nhưng nó cực kỳ cần thiết. Để hiểu tại sao như vậy, ta thấy rằng IP truyền dữ liệu giữa hai thiết bị đầu cuối, mỗi thiết bị có một địa chỉ IP nhất định. IP không truyền dữ liệu giữa các tiến trình ứng dụng chạy trên các máy tính. Mở rộng việc gửi - từ máy tính đến máy tính - tới từ tiến trình đến tiến trình là công việc dồn kênh và phân kênh.

Tại máy tính nhận, tầng giao vận nhận gói dữ liệu (hay còn gọi là segment) từ tầng mạng ngay phía dưới và có trách nhiệm gửi dữ liệu bên trong segment này tới tiến trình ứng dụng thích hợp trên máy tính. Giả sử lúc nào đó máy tính của bạn đang tải trang Web xuống, chạy một phiên FTP và hai phiên Telnet cùng một lúc. Như vậy bạn đang chạy 4 tiến trình ứng dụng: 2 tiến trình Telnet, 1 tiến trình FTP, và 1 tiến trình HTTP. Khi tầng giao vận trong máy tính của bạn nhận được dữ liệu từ tầng mạng chuyển lên, nó phải gửi dữ liệu trong đó tới 1 trong 4 tiến trình trên. Việc đó diễn ra như thế nào?

Mỗi segment của tầng giao vận có trường xác định tiến trình nhận dữ liệu. Tầng giao vận bên nhận sẽ sử dụng trường này để xác định rõ tiến trình nhận và gửi dữ liệu trong segment tới tiến trình đó. Công việc chuyển dữ liệu trong segment tới đúng tiến trình ứng dụng được gọi là phân kênh. Tại thiết bị gửi, tầng giao vận nhận dữ liệu từ nhiều tiến trình ứng dụng khác nhau, tạo segment chứa dữ liệu cùng với một số thông tin tiêu đề và cuối cùng chuyển segment xuống tầng mạng. Quá trình trên được gọi là dồn kênh. Hình 3.2 minh hoạ cả hai quá trình này.



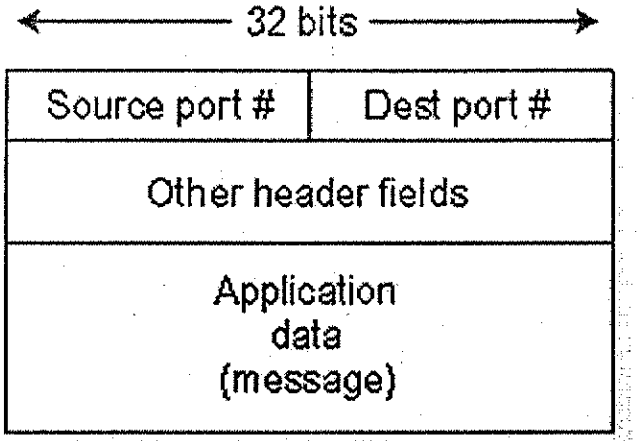
Hình 3.2 Dịch vụ dồn kênh, phân kênh

Để hiểu rõ hơn về dịch vụ dồn kênh, ta quay lại ví dụ trước. Mỗi đứa trẻ được xác định qua tên. Khi Bình nhận được thư từ người đưa thư, cậu bé sẽ thực hiện quá trình phân kênh bằng cách đọc tên trên phong bì thư để chuyển cho đúng người nhận. Còn An thực hiện quá trình dồn kênh khi thu thập thư từ mọi người và chuyển cho người đưa thư.

UDP và TCP thực hiện việc dồn kênh và phân kênh nhờ hai trường đặc biệt ở đầu segment: trường định danh cổng tiến trình gửi (nguồn – source port number) và trường định danh cổng tiến trình nhận (đích – destination port number). Hai trường này được minh họa trên Hình 3.3. Chúng xác định một tiến trình ứng dụng duy nhất chạy trên máy tính. Tất nhiên UDP và TCP còn có nhiều trường khác mà chúng ta sẽ nghiên cứu sau.

Khái niệm số hiệu cổng đã được giới thiệu qua trong chương 2. Nó là một con số 16 bit, nhận giá trị từ 0 tới 65535. Giá trị từ 0 đến 1023 là các giá trị đặc biệt và được sử dụng rất hạn chế, chỉ dành cho các ứng dụng thông dụng như HTTP, FTP sử dụng. HTTP sử dụng cổng 80, FTP sử dụng cổng 21. Danh sách các cổng thông dụng có thể tham khảo trong RFC 1700. Khi xây dựng một ứng dụng mới, phải xác định số hiệu cổng cho ứng dụng này.

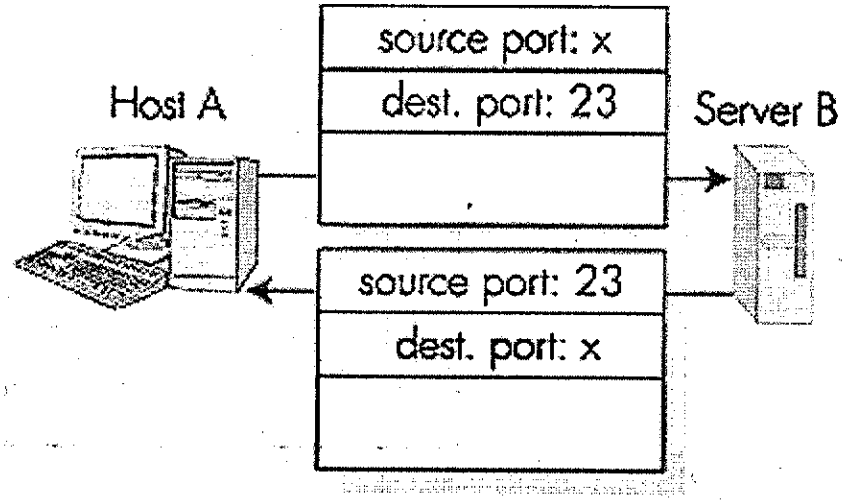
Mỗi ứng dụng chạy trên thiết bị đầu cuối có số hiệu cổng nhất định. Với vậy vấn đề đặt ra là tại sao mỗi segment ở tầng giao vận đều có trường số hiệu cổng nguồn và đích. Một thiết bị đầu cuối có thể chạy đồng thời hai tiến trình cùng kiểu, như vậy số hiệu cổng đích chưa đủ để phân biệt các tiến trình. Giả sử Web server chạy tiến trình HTTP xử lý các thông điệp yêu cầu; khi Web server phục vụ nhiều yêu cầu cùng một lúc (điều này hết sức thông thường) thì server sẽ chạy nhiều tiến trình trên cổng 80. Để gửi dữ liệu đến tiến trình nhận, phải xác định số hiệu cổng của phía gửi (cổng nguồn).



Hình 3.3 Trường địa chỉ tiến trình gửi, tiến trình nhận trong gói dữ liệu segment

Cổng nguồn được tạo ra như thế nào? Nhận giá trị bao nhiêu? Để trả lời câu hỏi này hãy nhớ lại rằng ứng dụng mạng sử dụng kiến trúc khách hàng/ người phục vụ. Thông thường máy tính nào khởi đầu trước đóng vai trò client, máy tính kia đóng vai trò server. Xét ví dụ một tiến trình ứng dụng có số hiệu cổng là 23 (số hiệu cổng của ứng dụng Telnet server). Hãy quan sát segment ở tầng giao vận khi rời client (là máy tính chạy chương trình Telnet client) chuyển tới server. Số hiệu cổng nguồn và đích của segment này là bao nhiêu? Số hiệu cổng đích chính là số hiệu cổng tiến trình nhận - 23. Còn số hiệu cổng nguồn - ở phía client – là một giá trị chưa được sử dụng bởi tiến trình nào, được phần mềm giao vận chạy trên máy tính client xác định tự động. Giả sử phía client chọn số hiệu cổng là x thì mỗi segment được gửi tới ứng dụng Telnet có cổng nguồn là x, cổng đích là 23. Khi segment tới, server căn cứ vào số hiệu cổng để chuyển dữ liệu trong segment tới đúng tiến trình ứng dụng nhận. Cổng đích 23 xác định tiến trình Telnet, cổng nguồn x để xác định một tiến trình gửi cụ thể.

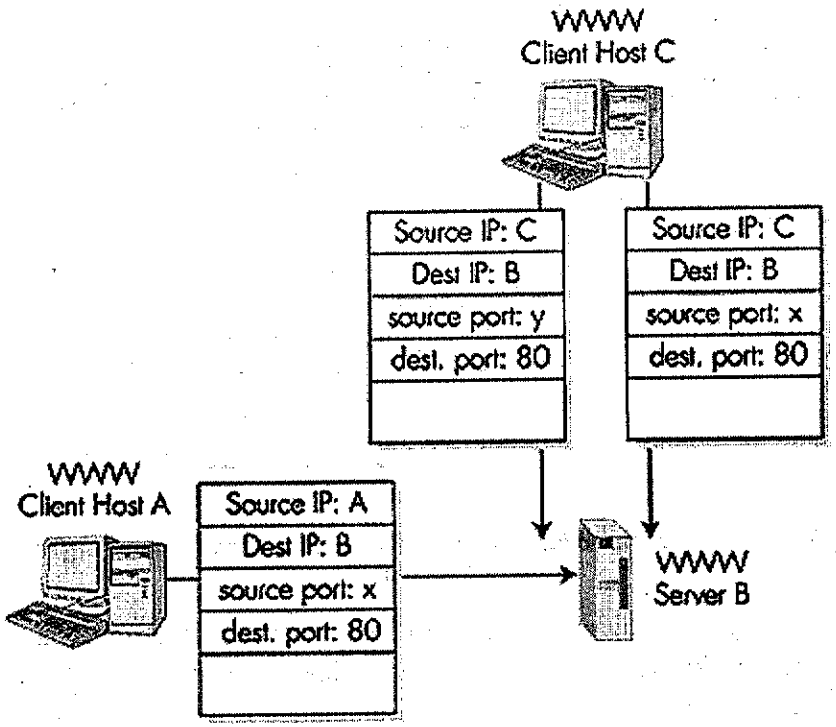
Segment truyền từ server tới client sẽ ngược lại. Cổng nguồn bây giờ sẽ là cổng của ứng dụng có giá trị 23, còn cổng đích sẽ là x (là số hiệu cổng nguồn trong segment gửi từ client tới server). Khi segment tới, client cũng sẽ căn cứ vào số hiệu cổng nguồn và đích để gửi dữ liệu trong segment tới đúng tiến trình ứng dụng. Hình 3.4 minh hoạ quá trình trên.



Hình 3.4 Sử dụng số hiệu cổng nguồn và cổng đích trong trình ứng dụng khách/chủ

Chuyện gì xảy ra nếu có hai client khác nhau cùng thiết lập phiên làm việc tới một server và mỗi client đều chọn cổng nguồn là x? Điều này rất dễ xảy ra với những Web server có nhiều người truy cập, phải phục vụ nhiều yêu cầu. Bên server phải phân kênh segment như thế nào khi hai phiên làm việc có cùng cặp số hiệu cổng? Khi đó server phải sử dụng địa chỉ IP trong gói dữ liệu IP (datagram) chứa segment. Trên Hình 3.5, máy C có hai phiên làm việc và máy A có một phiên làm việc HTTP tới cùng server B. Cả ba máy A,B,C đều có địa chỉ IP phân biệt lần lượt là A, B, C. Máy C sử dụng hai cổng nguồn (x,y) khác nhau cho hai kết nối HTTP tới B. A chọn số hiệu cổng nguồn độc lập với C nên nó có thể gán cổng nguồn x cho kết nối HTTP của mình. Tuy nhiên, máy chủ B vẫn có thể thực hiện phân kênh hai đoạn segement có cặp cổng giống nhau do địa chỉ IP nguồn khác nhau. Tóm lại, bên nhận sử dụng cả ba giá trị (địa chỉ IP nguồn, số hiệu cổng nguồn, số hiệu cổng đích) để xác định tiến trình ứng dụng nhận.

Sau khi xét tầng giao vận thực hiện việc dồn kênh và phân kênh các tiến trình ứng dụng như thế nào, chúng ta sẽ nghiên cứu một trong các giao thức giao vận của Internet - UDP. Trong phần này chúng ta sẽ thấy ngoài hai dịch vụ dồn kênh và phân kênh, UDP gần như không cung cấp dịch vụ nào khác.



Hình 3.5 Hai client cùng số hiệu cổng đích truyền thông với cùng một server

3.3 UDP – GIAO THỨC KHÔNG HƯỚNG NỐI

Trong phần này ta sẽ nghiên cứu cơ chế hoạt động của UDP. Độc giả cần nhớ lại khái quát về dịch vụ UDP trình bày trong phần 2.1.

Bạn sẽ làm gì nếu muốn xây dựng một giao thức giao vận cực kỳ đơn giản - một giao thức giao vận "rỗng"? Khi đó, thực thể giao vận phía gửi nhận thông điệp từ tiến trình ứng dụng và chuyển xuống tầng mạng; thực thể giao vận phía nhận chuyển thông điệp tầng mạng đưa lên tới chương trình ứng dụng tương ứng. Tầng giao vận chỉ cung cấp dịch vụ dồn kênh/phân kênh bằng cách chuyển dữ liệu đến từ tầng mạng tới đúng tiến trình ứng dụng nhận.

UDP đặc tả trong RFC 768 là giao thức giao vận cực kỳ đơn giản. Như vậy, bên cạnh chức năng dồn kênh/phân kênh, UDP có thêm cơ chế phát hiện lỗi đơn giản. Có thể nói nếu sử dụng UDP thì gần như ứng dụng làm việc trực

tiếp với tầng mạng IP. UDP lấy thông điệp từ tiến trình ứng dụng, chèn thêm một số trường tiêu đề, trong đó có hai trường địa chỉ cổng nguồn và đích cho dịch vụ dồn kênh/phân kênh để tạo nên gói dữ liệu segment. Gói segment sau khi tạo ra được chuyển xuống tầng mạng. Tầng mạng đặt segment này trong gói dữ liệu IP datagram và ***cố gắng*** gửi gói IP datagram tới máy tính nhận. Nếu segment tới đích, UDP sử dụng số hiệu cổng và địa chỉ IP của tiến trình nhận để truyền dữ liệu trong segment tới đúng tiến trình ứng dụng nhận. Chú ý UDP không đòi hỏi thực thể bên gửi và bên nhận phải liên kết trước khi trao đổi dữ liệu. Vì thế UDP được xem là dịch vụ **không hướng nối hay không liên kết trước (connectionless)**.

DNS là một giao thức tầng ứng dụng chạy trên nền UDP. Khi muốn truy vấn, DNS tạo thông điệp truy vấn DNS, chuyển thông điệp tới socket. UDP bổ sung một số trường vào đầu mỗi thông điệp để tạo ra UDP segment rồi gửi segment này xuống tầng mạng. Tầng mạng sẽ đóng gói UDP segment này trong IP datagram và gửi datagram tới đích (name server). Sau đó, DNS bên gửi đợi trả lời. Nếu không nhận được câu trả lời (điều này có thể xảy ra khi các tầng dưới làm mất thông điệp yêu cầu hay thông điệp trả lời) thì DNS gửi lại yêu cầu hoặc báo cho ứng dụng biết là không nhận được câu trả lời. Các đặc tả DNS cho phép DNS chạy trên nền TCP nhưng trong thực tế DNS thường chạy trên UDP.

So với UDP, TCP có vẻ có nhiều ưu điểm hơn: TCP cung cấp dịch vụ truyền dữ liệu tin cậy trong khi UDP không làm được. Tuy nhiên trên thực tế nhiều ứng dụng lại sử dụng UDP với các lý do sau đây:

Không có giai đoạn thiết lập kết nối: TCP sử dụng cơ chế "bắt tay" ba bước trước khi bắt đầu truyền dữ liệu thực sự. UDP không cần cơ chế này trước khi truyền dữ liệu. Vì thế UDP sẽ không phải chịu thời gian trễ để thiết lập đường truyền. Đây chính là nguyên nhân DNS chạy trên UDP chứ không phải là TCP. DNS sẽ chạy chậm nếu sử dụng TCP. HTTP sử dụng TCP vì các đối tượng Web cần được tải về chính xác – do đó yêu cầu một đường truyền tin cậy. Nhưng như đã trình bày trong phần 2.2, giai đoạn thiết lập đường truyền trong TCP gây nên một thời gian trễ cho ứng dụng HTTP (tình trạng "world wide wait").

Không duy trì trạng thái kết nối. TCP ghi nhớ trạng thái kết nối trên hệ thống đầu cuối. Trạng thái kết nối bao gồm vùng đệm (buffer) của bên nhận và bên gửi, các tham số kiểm soát tắc nghẽn, số tuần tự phát và số

biên nhận. Nó giúp TCP triển khai dịch vụ truyền tin tin cậy và cơ chế kiểm soát tắc nghẽn. Trong phần 3.5 ta sẽ hiểu ý nghĩa các trạng thái này. UDP không phải lưu giữ những thông tin như vậy. Do đó nếu phía server sử dụng UDP thì có khả năng phục vụ đồng thời nhiều client hơn.

Tiêu đề gói dữ liệu nhỏ.: Tiêu đề của TCP segment là 20 byte trong khi UDP chỉ có 8 bytes.

Không kiểm soát tốc độ gửi. TCP có cơ chế kiểm soát tắc nghẽn, điều chỉnh tốc độ gửi khi xảy ra tắc nghẽn. Cơ chế điều chỉnh này có thể ảnh hưởng tới những ứng dụng thời gian thực – là những ứng dụng chấp nhận mất mát dữ liệu (trong phạm vi nào đó) nhưng lại đòi hỏi phải có một tốc độ truyền tối thiểu. Tốc độ truyền dữ liệu của UDP chỉ bị giới hạn bởi tốc độ sinh dữ liệu của ứng dụng, khả năng máy tính nguồn (CPU, tốc độ đồng hồ), và tốc độ truy cập mạng. Chú ý rằng bên nhận không nhất thiết phải nhận toàn bộ dữ liệu. Khi mạng bị tắc nghẽn, một phần dữ liệu có thể bị mất do tràn vùng đệm ở router. Tốc độ nhận có thể bị giới hạn do tắc nghẽn ngay cả khi tốc độ gửi không bị giới hạn.

Ứng dụng	Giao thức tầng ứng dụng	Tầng giao vận tương ứng
Thư điện tử	SMTP	TCP
Truy cập từ xa	Telnet	TCP
Web	HTTP	TCP
Truyền file	FTP	TCP
File server	NFS	thường là UDP
Đa phương tiện	Phụ thuộc vào hãng sản xuất	thường là UDP
Điện thoại qua Internet	Phụ thuộc vào hãng sản xuất	thường là UDP
Quản lý mạng	SNMP	thường là UDP
Định tuyến	RIP	thường là UDP
Tên miền	DNS	thường là UDP

Hình 3.6 Các ứng dụng thông dụng và giao thức giao vận tương ứng

Hình 3.6 liệt kê một số ứng dụng phổ biến và giao thức giao vận của chúng. Email, truy cập từ xa, Web và truyền file 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. Giao thức cập nhật bảng định tuyến RIP (sẽ học trong chương 4) sử dụng UDP, bởi vì việc cập nhật được thực hiện định kỳ (thường khoảng 5 phút một lần), cho nên dù cập nhật bị mất nhưng sẽ có cập nhật mới sau một khoảng thời gian ngắn. UDP được sử dụng để gửi dữ liệu quản trị mạng (SNMP). Trong trường hợp này UDP thích hợp hơn TCP vì các tiến trình quản trị mạng thường hoạt động khi mạng có sự cố không thể truyền dữ liệu chính xác hay cơ chế kiểm soát tắc nghẽn không làm việc. DNS sử dụng UDP, do đó có thể tránh được thời gian trễ trong giai đoạn thiết lập kết nối.

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 sử dụng. Các ứng dụng như thế có thể chấp nhận mất mát, lỗi trên một phần dữ liệu, vì thế truyền dữ liệu tin cậy không phải là tiêu chí quan trọng nhất đánh giá sự thành công của ứng dụng. Hơn nữa các ứng dụng thời gian thực không thích ứng được với cơ chế kiểm soát tắc nghẽn của TCP. Do đó chúng thường lựa chọn UDP ở tầng giao vận.

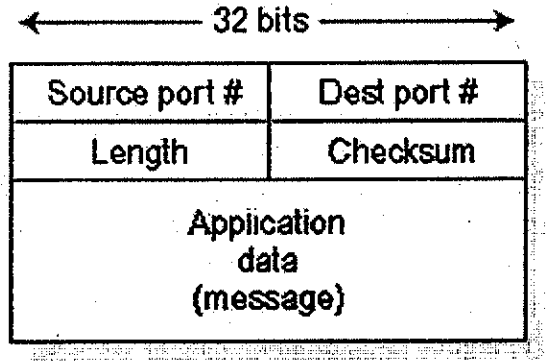
Hiện nay mặc dù đã triển khai trong thực tế, song việc các ứng dụng đa phương tiện sử dụng UDP gây ra nhiều tranh cãi. Như đã nói ở trên, UDP không kiểm soát được tắc nghẽn nên mạng rất dễ bị tắc nghẽn – khi đó chỉ rất ít thông tin được chuyển. Nếu tất cả mọi người đều xem phim trực tuyến thì các gói tin sẽ bị tràn bộ đệm ở các router – và khi đó thì chẳng ai xem được gì cả. Thiếu cơ chế kiểm soát tắc nghẽn có thể sẽ là một vấn đề nghiêm trọng đối với UDP. Người ta đã đưa ra nhiều cơ chế đòi hỏi tất cả các thực thể - kể cả UDP - thực hiện một cơ chế kiểm soát lưu lượng thích nghi.

Trước khi trình bày về cấu trúc UDP, cần chú ý rằng tuy sử dụng UDP nhưng ứng dụng vẫn có thể có một đường truyền tin cậy. Điều này được thực hiện bằng cách đảm bảo tính tin cậy ngay trong bản thân ứng dụng (bằng các cơ chế đánh số thứ tự, truyền lại). Công việc này sẽ làm ứng dụng cồng kềnh và phức tạp. Tuy nhiên ưu điểm là ứng dụng có thể truyền thông tin cậy với tốc độ không bị cơ chế kiểm soát tắc nghẽn của TCP khống chế. Ngày nay một số phần mềm chuyên dụng đa phương tiện sử

dụng cơ chế đánh số thứ tự và truyền lại ngay trong chương trình ứng dụng để giảm bớt việc mất dữ liệu.

3.3.1 Cấu trúc UDP segment

Cấu trúc UDP segment, đặc tả trong RFC 768 được minh hoạ trên Hình 3.7. Dữ liệu của ứng dụng nằm trong trường dữ liệu của UDP datagram. Ví dụ đối với DNS, trường dữ liệu chứa thông điệp yêu cầu hay thông điệp trả lời. Tiêu đề UDP có bốn trường, độ lớn mỗi trường là hai byte. Như đã nói ở phần trước, số hiệu cổng cho phép thiết bị gửi chuyển dữ liệu tới đúng tiến trình chạy trên thiết bị nhận (chức năng phân kênh). Trường Checksum được bên nhận sử dụng để kiểm tra trong segment có lỗi hay không. Trên thực tế, kể cả tiêu đề của gói dữ liệu IP cũng được tính checksum. Nguyên tắc cơ bản của cơ chế phát hiện và sửa lỗi được trình bày trong phần 5.1. Trường độ dài (Length) cho biết độ dài (tính theo byte) của toàn bộ gói dữ liệu UDP segment - kể cả phần tiêu đề.



Hình 3.7 Cấu trúc gói UDP datagram

3.3.2 UDP checksum

UDP checksum được sử dụng để phát hiện lỗi. Checksum được tính như sau: tính giá trị bù một của tổng các từ 16 bit trong segment, giá trị nhận được được đặt vào trường checksum trong gói dữ liệu UDP segment. Có thể

tìm hiểu phương thức triển khai trong RFC 1071 và hiệu quả trên dữ liệu thực trong [Stone 1998 và Stone 2000]. Giả sử có ba từ 16 bit sau đây:

0110011001100110

0101010101010101

0000111100001111

Tổng hai từ đầu là:

0110011001100110

0101010101010101

1011101110111011

Cộng từ thứ ba vào, ta có:

1011101110111011

0000111100001111

1100101011001010

Cách lấy bù một là đảo 0 thành 1 và 1 thành 0. Vì vậy kết quả phép lấy bù một của 1100101011001010 là 0011010100110101 và đó chính là giá trị checksum. Tại phía nhận, tất cả bốn từ (kể cả checksum) được cộng lại. Nếu dữ liệu không có lỗi thì tổng nhận được là 1111111111111111. Nếu có một bit nào đó bằng 0 thì ta biết dữ liệu nhận được có lỗi.

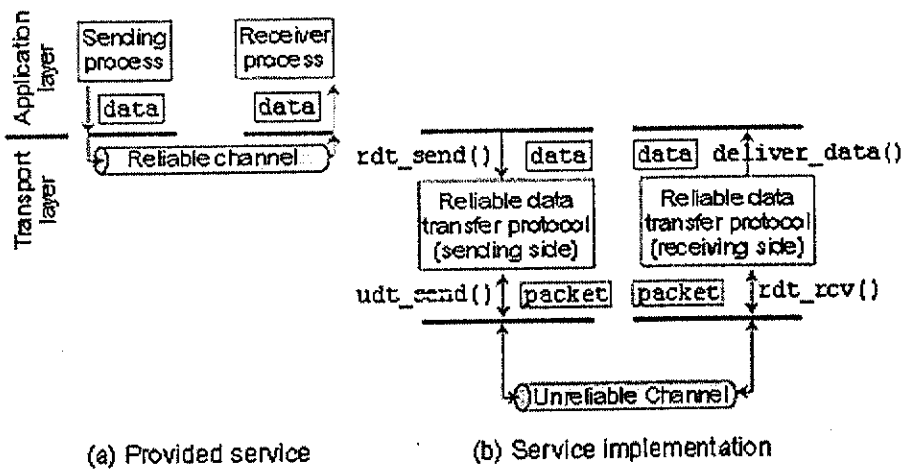
Bạn có thể hỏi tại sao UDP tính checksum – trong khi một vài giao thức tầng liên kết dữ liệu (kể cả giao thức Ethernet thông dụng) cũng có cơ chế kiểm tra lỗi. Lí do là chưa chắc tất cả các kết nối (link - đường truyền vật lý thực sự) giữa thiết bị gửi và thiết bị nhận đều có cơ chế kiểm tra lỗi – có thể một trong các kết nối đó sử dụng giao thức không cung cấp việc kiểm tra lỗi. Mặc dù UDP có thể phát hiện được lỗi nhưng nó không làm gì khi phát hiện ra lỗi. Có thể nó sẽ loại bỏ segment bị lỗi, có thể nó sẽ chuyển segment bị lỗi cho ứng dụng nhận cùng với một thông điệp cảnh báo.

TCP cung cấp đường truyền tin cậy – do đó hiển nhiên triển khai nó phức tạp hơn UDP rất nhiều. Trước khi tìm hiểu về TCP, trong phần sau chúng ta sẽ trình bày nguyên tắc chung để xây dựng một đường truyền tin cậy. TCP sẽ áp dụng đúng những nguyên tắc này khi triển khai.

3.4 CÁC NGUYÊN TẮC TRUYỀN DỮ LIỆU TIN CẬY

Phần này trình bày tổng quan dịch vụ truyền dữ liệu tin cậy. Dịch vụ này không chỉ nằm ở tầng giao vận mà còn có thể nằm ở tầng liên kết dữ liệu hay tầng ứng dụng. Có thể nói truyền dữ liệu tin cậy là một trong những vấn đề quan trọng nhất của mạng. Trong phần kế tiếp về TCP, chúng ta sẽ nghiên cứu cách thức TCP áp dụng các nguyên tắc chung được trình bày ở đây như thế nào.

Hình 3.8 là sơ đồ cấu trúc của quá trình truyền dữ liệu tin cậy. Tầng dưới cung cấp dịch vụ truyền tin cậy cho các thực thể ở tầng trên. Trên đường truyền tin cậy này, dữ liệu không bị lỗi (bit 0 biến thành bit 1 hoặc ngược lại), không bị mất và được nhận theo đúng thứ tự gửi. Đây chính là dịch vụ mà TCP cung cấp cho các ứng dụng Internet.



Hình 3.8 Dịch vụ truyền dữ liệu tin cậy: Mô hình và Triển khai

Để thực hiện công việc này, người ta cần đến những giao thức truyền dữ liệu tin cậy. Nguyên nhân là tầng phía dưới của giao thức tin cậy là không tin cậy. Ví dụ TCP là giao thức truyền dữ liệu tin cậy nằm ở phía trên giao thức truyền không tin cậy (IP) giữa hai thiết bị đầu cuối trên mạng.

Để đơn giản trong trường hợp này chúng ta coi tầng phía dưới là một đường truyền điểm nối điểm (point-to-point) không tin cậy.

Trong phần này, chúng ta sẽ xây dựng dần giao thức truyền dữ liệu tin cậy giữa phía gửi và phía nhận theo độ phức tạp tăng dần của kênh truyền bên dưới. Hình 3.8b minh hoạ điều này. Thực thể gửi sẽ nhận dữ liệu từ phía trên chuyển xuống qua hàm **rdt_send()** (Ở đây rdt là viết tắt của "reliable data transfer" và **_send** chỉ rõ đây là phía gửi của giao thức rdt. Bước đầu tiên khi xây dựng một giao thức nào đó là chọn cho nó một cái tên dễ nhớ!). Phía nhận sử dụng hàm **rdt_rcv()** để lấy gói dữ liệu từ đường truyền. Để chuyển dữ liệu lên tầng trên, phía nhận sử dụng hàm **deliver_data()**. Trong phần này, chúng ta sử dụng thuật ngữ **"packet"** thay thế "segment" với ý nghĩa là đơn vị dữ liệu giao thức - PDU. Ý tưởng trình bày trong phần này không chỉ áp dụng cho tầng giao vận mà còn áp dụng chung cho toàn mạng máy tính, vì thế sử dụng thuật ngữ **"packet"** thích hợp hơn.

Trong phần này chỉ nghiên cứu trường hợp dữ liệu truyền theo một hướng từ nơi gửi đến nơi nhận. Trường hợp dữ liệu truyền theo hai hướng là một vấn đề không khó về mặt lý thuyết nhưng triển khai cụ thể tương đối phức tạp. Mặc dù dữ liệu chỉ được truyền theo một hướng nhưng các bên truyền thông trong giao thức rdt cần truyền dữ liệu theo cả hai hướng (xem Hình 3.8) bởi vì ngoài các gói dữ liệu thực sự, chúng còn phải trao đổi các gói dữ liệu chứa thông tin điều khiển. Cả bên gửi và bên nhận đều sử dụng hàm **udt_send()** để gửi dữ liệu đến phía bên kia (udt là viết tắt của unreliable data transfer).

3.4.1 Xây dựng giao thức truyền dữ liệu tin cậy

Bây giờ chúng ta sẽ từng bước nghiên cứu các giao thức với độ phức tạp tăng dần để cuối cùng đi đến giao thức truyền dữ liệu không lỗi. Chúng ta sẽ mô tả trạng thái của phía nhận và phía gửi bằng kỹ thuật máy hữu hạn trạng thái (**finite state machine – FSM**)

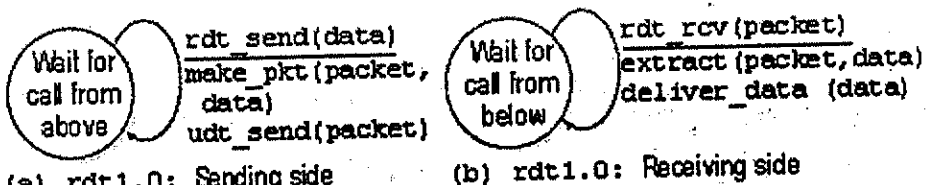
Truyền dữ liệu tin cậy trên kênh truyền tin cậy hoàn toàn: giao thức rdt 1.0.

Giao thức đầu tiên, đơn giản nhất được đưa ra - rdt 1.0 sử dụng kênh truyền tin cậy ở phía dưới. Giao thức rdt 1.0 cực kỳ đơn giản, FSM của bên gửi và bên nhận đều chỉ có một trạng thái (xem Hình 3.9). Mũi tên trong sơ đồ chỉ sự chuyển trạng thái của giao thức (mặc dù mỗi FSM trong Hình 3.9 chỉ có một trạng thái, vẫn cần đến sự chuyển trạng thái để quay về chính trạng thái cũ). Sự kiện kích hoạt việc chuyển trạng thái được đặt phía trên đường kẻ nằm ngang, đó là nhãn sự kiện. Phía bên dưới đường kẻ nằm ngang là những hành động mà thực thể phải thực hiện ngay khi sự kiện đó xảy ra (thực hiện trước khi thực thể chuyển sang trạng thái mới).

Với rdt 1.0, việc gửi đơn giản chỉ là nhận dữ liệu từ tầng trên thông qua sự kiện **rdt_send(data)**, tạo ra gói dữ liệu (bằng hành động **make_data (packet,data)**) và gửi gói dữ liệu (packet) lên kênh truyền. Trên thực tế, sự kiện **rdt_send(data)** là kết quả của một thủ tục (ví dụ khi ứng dụng phía trên sử dụng hàm rdt_send()) .

Ở phía nhận, rdt nhận gói dữ liệu **(packet)** từ kênh truyền bằng sự kiện **rdt_rcv(packet)**, lấy dữ liệu ra khỏi gói dữ liệu (bằng hành động **extract (packet,data)**) và đưa dữ liệu lên tầng trên. Trên thực tế, sự kiện **rdt_rcv(packet)** là kết quả của một thủ tục (ví dụ khi ứng dụng phía trên sử dụng hàm **rdt_rcv()**).

Trong giao thức đơn giản này, không có sự khác biệt giữa dữ liệu **(data)** với gói dữ liệu **(packet)**. Như vậy, tất cả packet đều được truyền từ phía gửi cho phía nhận. Với kênh truyền tin cậy, phía nhận không cần thiết phải phản hồi cho phía gửi vì nó chắc rằng không có chuyện gì xảy ra. Chú ý rằng, chúng ta đã giả thiết phía nhận có thể nhận dữ liệu với tốc độ phía gửi gửi. Vì vậy, phía nhận không cần yêu cầu phía gửi gửi chậm lại.



Hình 3.9 Giao thức cho kênh truyền tin cậy hoàn toàn

Truyền dữ liệu tin cậy trên kênh truyền có lỗi bit: giao thức rdt 2.0.

Một dạng kênh truyền thực tế hơn là gói dữ liệu trên kênh truyền có thể bị lỗi. Thường bit bị lỗi trên đường truyền vật lý của mạng. Tuy nhiên, chúng ta giả thiết rằng tất cả các gói dữ liệu truyền đi đều đến được đích và theo đúng thứ tự gửi mặc dù các bit trong gói dữ liệu có thể bị lỗi.

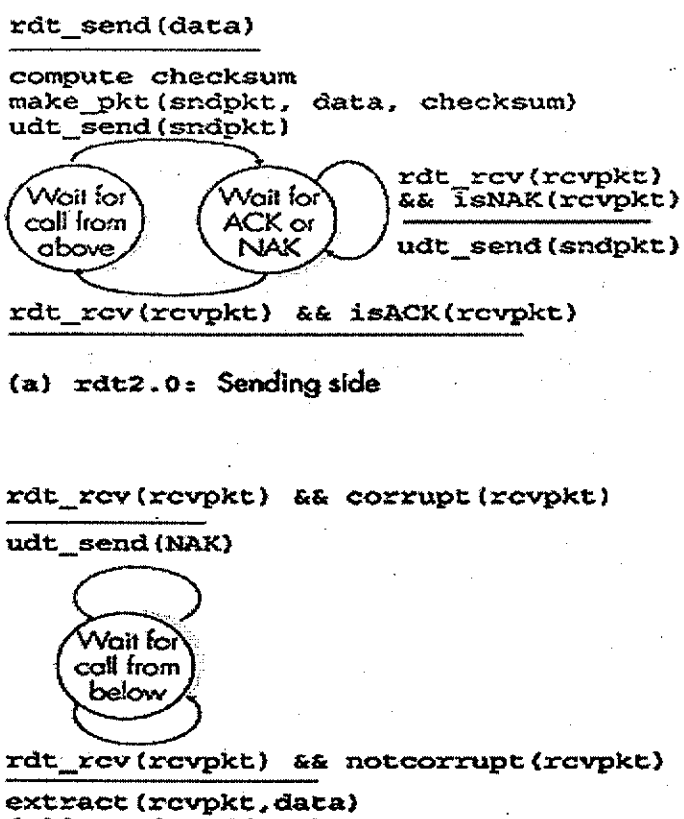
Trước khi tiếp tục, xét ví dụ sau. Giả sử bạn đọc một bài chính tả cho ai đó qua điện thoại. Thông thường, người chép sẽ nói "Xong rồi" sau khi đã nghe, hiểu và ghi lại một câu chính tả. Nếu câu nói của bạn bị nhiễu, người kia nghe không rõ thì họ sẽ yêu cầu bạn nhắc lại. Giao thức truyền tin này sử dụng **phản hồi tích cực** (positive acknowledgement) ("Xong rồi") **hay phản hồi tiêu cực** (negative acknowledgement) ("Gì cơ ?"). Những thông điệp điều khiển này cho phép bên nhận báo cho bên gửi biết dữ liệu nào nhận đúng, dữ liệu nào bị lỗi và yêu cầu truyền lại dữ liệu bị lỗi. Trong mạng máy tính, giao thức truyền tin cậy dựa trên cơ chế truyền lại như vậy được gọi là các giao thức **ARQ (Automatic Repeat reQuest)**.

Các giao thức ARQ cần phải có ba khả năng sau để xử lý trong trường hợp dữ liệu có lỗi:

Phát hiện lỗi (error detection): là cơ chế cho phép bên nhận phát hiện được khi nào trong gói dữ liệu có bit bị lỗi. Trong phần trước, ta thấy UDP sử dụng trường Internet checksum cho mục đích này. Trong chương V, chúng ta sẽ xem xét chi tiết một số kỹ thuật phát hiện và thậm chí có thể sửa được lỗi. Còn bây giờ chúng ta chỉ cần biết rằng những kỹ thuật như vậy yêu cầu ngoài việc gửi dữ liệu gốc, bên gửi còn phải tạo ra và gửi kèm một lượng dữ liệu dư thừa (nhưng phụ thuộc vào dữ liệu gốc). Các bit dư thừa này được đặt trong trường checksum của gói dữ liệu rdt 2.0.

Phản hồi từ phía nhận (receiver feedback): Khi phía gửi và phía nhận nằm trên các thiết bị đầu cuối khác nhau - có thể cách nhau hàng nghìn km, cách duy nhất để phía gửi biết được kết quả gửi là phía nhận gửi thông tin phản hồi thông báo tình trạng nhận cho phía gửi. Báo nhận đúng (đôi khi gọi là báo nhận tích cực) ACK và báo nhận sai NAK trong ví dụ trên chính là các thông tin phản hồi. Giao thức rdt 2.0 yêu cầu phía nhận gửi phản hồi các thông điệp ACK hay NAK cho phía gửi. Gói dữ liệu phản hồi chỉ cần sử dụng một bit, ví dụ giá trị 0 ứng với NAK và giá trị 1 ứng với ACK.

Truyền lại (retransmission): gói dữ liệu bị lỗi sẽ được bên gửi truyền lại.



Hình 3.10 Giao thức cho kênh truyền có lỗi bit

Hình 3.10 là FSM của phía gửi và nhận trong giao thức rdt 2.0 với cơ chế phát hiện lỗi, phản hồi (ACK, NAK) và truyền lại.

Trong giao thức rdt 2.0, phía gửi có hai trạng thái. Ở trạng thái thứ nhất, phía gửi đợi dữ liệu từ tầng trên. Trong trạng thái thứ hai, phía gửi đợi phản hồi ACK hoặc NAK từ phía nhận. Nếu nhận được ACK (rdt_rcv(rcvpkt) & isACK(rcvpkt) trong Hình 3.10 tương ứng với sự kiện này), phía gửi biết được gói dữ liệu chuyển đến đích an toàn, vì vậy nó trở về trạng thái đợi dữ liệu từ tầng trên để chuyển tiếp. Nếu nhận được NAK,

phía gửi gửi lại gói dữ liệu rồi quay lại trạng thái đợi phản hồi ACK hoặc NAK cho gói dữ liệu vừa gửi lại. Chú ý rằng khi phía gửi ở trong trạng thái chờ phản hồi (ACK hoặc NAK), nó không thể nhận thêm dữ liệu từ tầng trên đưa xuống. Nó chỉ chấp nhận dữ liệu mới khi nhận được ACK và chuyển trạng thái. Phía gửi sẽ không gửi dữ liệu mới cho đến khi nó chắc chắn rằng phía nhận đã nhận đúng gói dữ liệu đã gửi. Giao thức rdt 2.0 với hành vi như vậy thuộc kiểu **dừng và chờ (stop and wait)**.

FSM bên nhận trong giao thức rdt 2.0 chỉ có một trạng thái duy nhất. Khi nhận được gói dữ liệu (packet), phía nhận gửi thông điệp phản hồi ACK hoặc NAK, phụ thuộc vào gói dữ liệu đã nhận có lỗi hay không. Trong Hình 3.10, rdt_rcv (rcvpkt) && corrups(rcvpkt) tương ứng với sự kiện gói dữ liệu nhận được bị lỗi.

Giao thức rdt 2.0 vẫn còn nhược điểm: chúng ta chưa tính đến khả năng chính gói ACK hoặc NAK có lỗi. (Trước khi tiếp tục bạn hãy thử nghĩ cách cải tiến giao thức này). Chúng ta cần tạo checksum cho chính gói phản hồi (ACK hoặc NAK) để bên gửi (lúc này đóng vai trò bên nhận) có khả năng phát hiện lỗi trong chính gói phản hồi. Vấn đề ở đây là khi nhận được một gói phản hồi bị lỗi – phía gửi không thể xác định nó là ACK nay NAK, do đó không xác định được gói dữ liệu nó gửi tới đích có bị lỗi hay không. Trong trường hợp này, bên gửi sẽ phải làm gì ?

Có ba giải pháp xử lý ACK hoặc NAK bị lỗi:

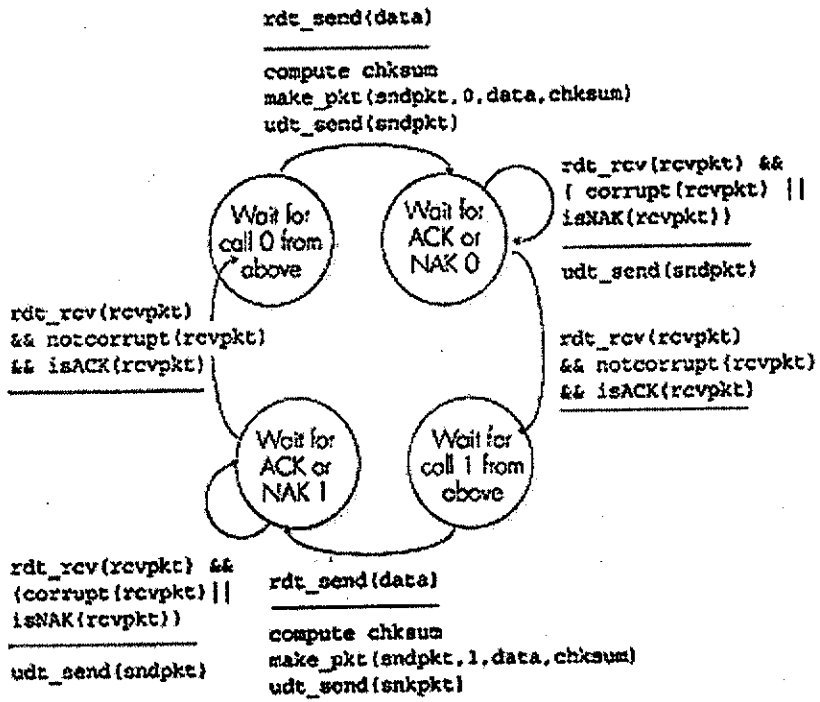
Giải pháp thứ nhất, người đọc trong ví dụ đọc chính tả lúc nãy sẽ làm gì trong trường hợp này? Nếu không hiểu câu phản hồi "Xong rồi" hay "Gì cơ" thì họ có thể hỏi "Bạn nói gì?" (một dạng thông điệp điều khiển khác). Nếu nghe được, người bên kia sẽ lặp lại câu phản hồi. Nhưng chuyện gì xảy ra nếu chính câu "Bạn nói gì?" có lỗi? Khi đó phía nhận - do không xác định được câu có lỗi đó là một phần trong bài chính tả hay là yêu cầu nhắc lại câu phản hồi - nên có thể phản hồi lại bằng câu "Bạn nói gì?". Dĩ nhiên, câu trả lời này cũng có thể bị lỗi. Rõ ràng giải pháp này đã đi vào ngõ cụt.

Giải pháp thứ hai là thêm vào trường checksum một số bit để không những cho phép phía nhận phát hiện mà còn sửa được các bit lỗi. Đây hoàn toàn có thể là giải pháp trung gian cho những kênh truyền có lỗi – nhưng không xử lý được trường hợp toàn bộ gói dữ liệu (packet) bị mất.

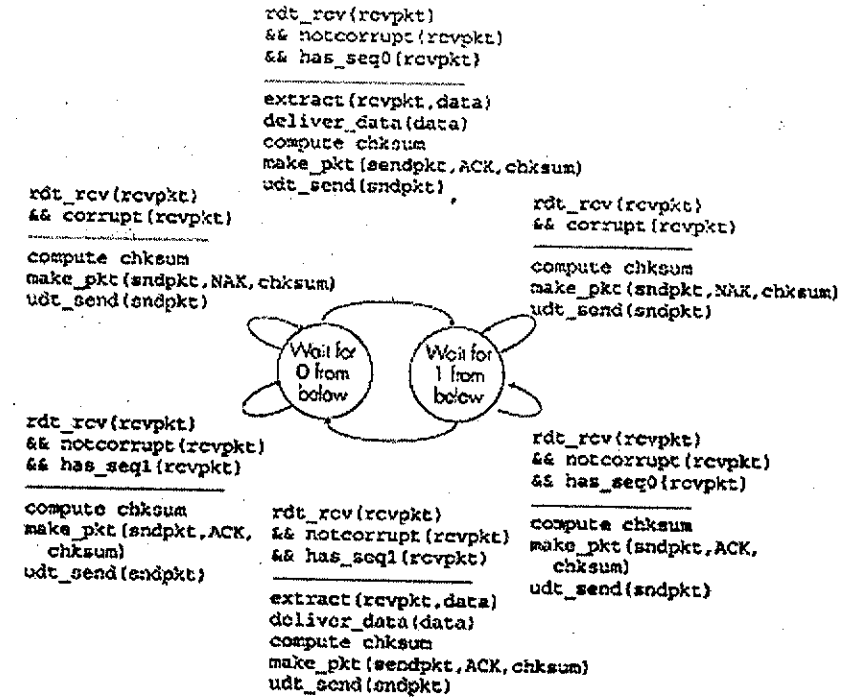
Giải pháp thứ ba, phía gửi truyền lại gói dữ liệu nếu phát hiện lỗi trong gói phản hồi (ACK hoặc NAK). Tuy nhiên, phương pháp này có thể

dẫn đến **sự trùng lặp dữ liệu (duplicate packet)**. Phía nhận không biết được ACK/NAK mà nó gửi phản hồi có bị lỗi trên đường truyền không. Vì thế nó không xác định được gói dữ liệu vừa nhận được là gói dữ liệu mới hay gói cũ (sẽ bị trùng lặp).

Giải pháp đơn giản nhất cho vấn đề này (sẽ được áp dụng cho nhiều giao thức, kể cả TCP) là thêm trường số thứ tự cho gói dữ liệu (packet), phía gửi đánh số thứ tự các gói dữ liệu và đặt giá trị này vào trường số thứ tự (sequence number). Phía nhận chỉ cần kiểm tra số thứ tự để xác định gói dữ liệu nhận được là gói mới hay gói truyền lại. Với giao thức **stop and wait** đơn giản, chỉ cần một bit số thứ tự. Bên nhận có thể xác định bên gửi truyền lại gói dữ liệu đã gửi lần trước (số thứ tự của gói dữ liệu nhận được trùng với số thứ tự với gói dữ liệu nhận được lần trước) hay gói dữ liệu mới (có số thứ tự khác nhau, tăng lên theo module 2). Vì chúng ta vẫn giả định toàn bộ gói dữ liệu (packet) không bị mất trên kênh truyền, nên trong gói phản hồi (ACK/NAK) không cần chỉ ra số thứ tự của gói dữ liệu mà chúng biên nhận. Phía gửi biết rằng gói ACK/NAK (có thể bị lỗi hoặc không) là biên nhận cho gói dữ liệu gần nhất nó gửi.



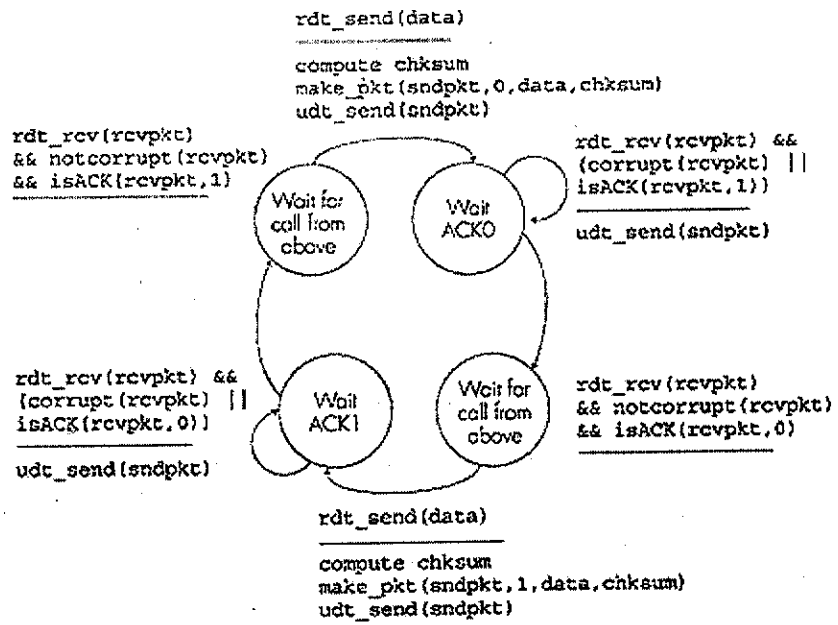
Hình 3.11 FSM của phía gửi trong rdt 2.1



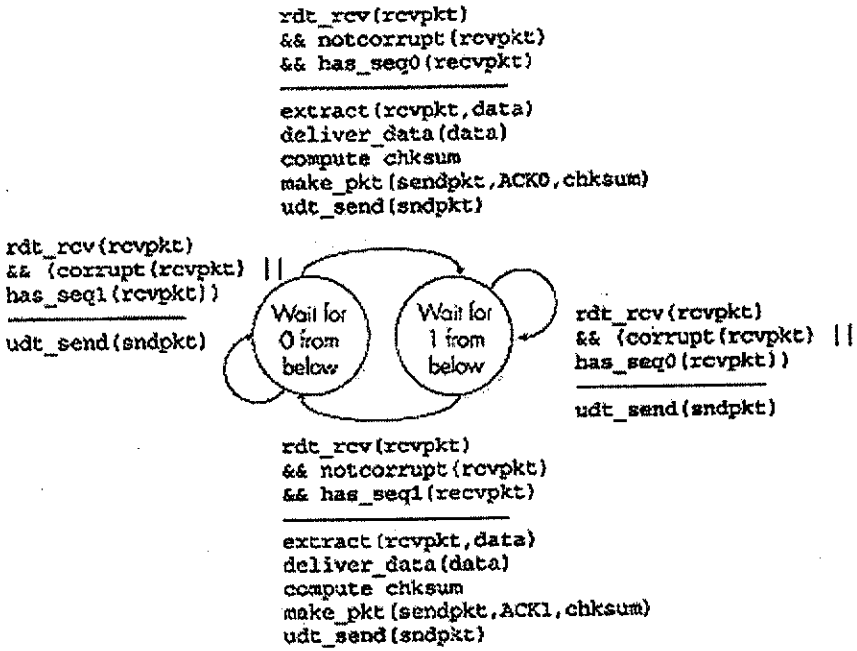
Hình 3.12 FSM của phía nhận trong rdt 2.1

Hình 3.11 và Hình 3.12 là FSM của bên gửi và nhận trong giao thức rdt 2.1 – phiên bản mới của rdt 2.0. Trong rdt 2.1, FSM của bên gửi và nhận đều có số trạng thái tăng gấp đôi. Đó là vì trạng thái giao thức phải biểu diễn gói dữ liệu được gửi (bởi bên gửi) và gói dữ liệu được đợi (tại bên nhận) có số thứ tự là 0 hay 1. Chú ý rằng các hành động trong trạng thái gói dữ liệu có số thứ tự 0 được gửi (phía gửi) hoặc được mong đợi (phía nhận) ngược với trạng thái gói dữ liệu có số thứ tự 1 được gửi hay được đợi.

Giao thức rdt 2.1 sử dụng cả biên nhận đúng (ACK) và biên nhận sai (NAK). NAK được gửi khi nhận được gói dữ liệu bị lỗi hay không đúng số thứ tự. Chúng ta có thể không cần sử dụng NAK: thay vì việc gửi NAK, chúng ta gửi ACK cho gói dữ liệu cuối cùng đã được nhận đúng. Nếu nhận hai ACK cho cùng một gói dữ liệu (hiện tượng **trùng ACK – duplicate ACK**) bên gửi xác định được bên nhận không nhận đúng gói dữ liệu sau gói dữ liệu đã biên nhận ACK hai lần. TCP sử dụng sự kiện "3 lần nhận được ACK trùng nhau" ("triple duplicate ACKs") để kích hoạt việc gửi lại. rdt 2.2 là giao thức truyền dữ liệu tin cậy trên kênh truyền có bit lỗi không sử dụng NAK, minh hoạ trên Hình 3.13 và Hình 3.14.



Hình 3.13 FSM của phía gửi trong rdt 2.2



Hình 3.14 FSM của phía nhận trong rdt 2.1

Truyền dữ liệu tin cậy trên kênh truyền mà dữ liệu bị mất, lỗi: rdt 3.0.

Dữ liệu trên kênh truyền không những bị lỗi mà còn có thể bị mất, đây là tình huống khá phổ biến trong mạng máy tính ngày nay, kể cả Internet. Lúc này giao thức cần phải giải quyết hai vấn đề: làm thế nào để phát hiện gói dữ liệu bị mất và làm gì khi mất gói dữ liệu. Sử dụng cơ chế phát hiện lỗi nhờ checksum, số thứ tự, biên nhận ACK và truyền lại gói dữ liệu - đã được phát triển trong giao thức rdt 2.2 - cho phép chúng ta giải quyết được vấn đề thứ hai. Để giải quyết vấn đề thứ nhất, chúng ta cần một cơ chế mới.

Có nhiều giải pháp xử lý việc mất mát dữ liệu. Ở đây chúng ta trình bày giải pháp lựa chọn bên gửi là nơi phát hiện và xử lý việc mất dữ liệu. Giả sử phía gửi gửi đi gói dữ liệu nhưng chính gói dữ liệu đó hoặc biên nhận ACK cho nó bị mất trên đường truyền. Trong cả hai trường hợp, bên gửi đều không nhận được biên nhận cho gói dữ liệu đã gửi. Giải pháp được đưa ra là sau khi gửi một khoảng thời gian nào đó mà không nhận được biên nhận ACK (có thể gói dữ liệu bị mất) thì bên gửi sẽ truyền lại.

Nhưng phía gửi phải đợi trong bao lâu để chắc chắn rằng gói dữ liệu đã bị mất? Ít nhất phía gửi phải đợi trong khoảng thời gian để gói tin đi đến được phía nhận, phía nhận xử lý gói tin và thông tin biên nhận quay lại. Trong nhiều mạng, rất khó dự đoán và ước lược thời gian này. Lý tưởng là phải xử lý việc mất gói tin ngay khi có thể, đợi một khoảng thời gian dài đồng nghĩa với việc chậm trễ khi xử lý gói tin bị mất. Trên thực tế, phía gửi sẽ chọn một khoảng thời gian đợi nào đó, mặc dù không đảm bảo chắc chắn là gói tin có bị mất hay không. Nếu không nhận được ACK trong khoảng thời gian này, bên gửi sẽ gửi lại gói dữ liệu. Chú ý rằng, nếu gói dữ liệu đến trễ, phía gửi sẽ gửi lại gói dữ liệu - ngay cả khi gói dữ liệu đó và cả ACK đều không bị mất. Điều này gây ra trùng lặp dữ liệu tại phía nhận. Tuy nhiên, giao thức rdt 2.2 đã có đủ khả năng (nhờ số thứ tự) để ngăn chặn sự trùng lặp dữ liệu.

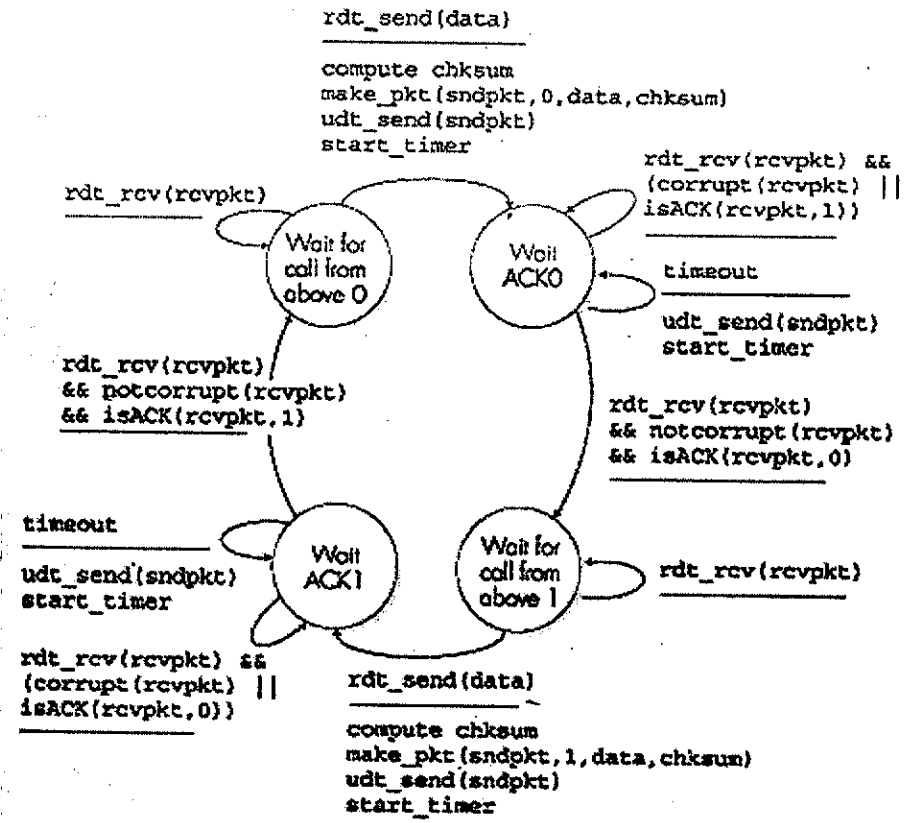
Đối với phía gửi, truyền lại là giải pháp "vạn năng". Phía gửi không xác định được gói dữ liệu bị mất, gói biên nhận ACK bị mất hay chỉ đơn giản là chúng bị trễ. Trong mọi trường hợp, hành động của nó là giống nhau:

truyền lại. Để thực hiện cơ chế truyền lại theo thời gian, một bộ định thời đếm ngược (countdown timer) được sử dụng để nhắc phía gửi thời gian đợi đã hết. Do vậy, phía gửi phải có khả năng (1) khởi tạo timer mỗi khi gửi gói dữ liệu (gói dữ liệu gửi lần đầu hay gói dữ liệu được truyền lại), (2) phản ứng với ngắt của timer (đưa ra những hành động thích hợp) và (3) dừng timer.

Sự trùng lặp các gói dữ liệu do phía gửi tạo ra, sự mất mát các gói dữ liệu (cả gói dữ liệu lẫn gói biên nhận) gây khó khăn cho phía gửi khi xử lý các gói biên nhận ACK. Nếu nhận được ACK, làm thế nào để phía gửi biết được ACK đó là biên nhận cho gói dữ liệu gửi đi gần đây nhất, hay là ACK biên nhận cho gói dữ liệu nào đó đã gửi từ trước nhưng đến trễ? Giải pháp là ta thêm vào gói ACK trường **số thứ tự biên nhận (acknowledge number)**. Giá trị của trường này – do phía nhận tạo ra - là số thứ tự của chính gói dữ liệu cần được biên nhận. Bằng cách kiểm tra giá trị trường biên nhận, phía gửi có thể xác định được số thứ tự của gói dữ liệu được biên nhận.

Hình 3.15 là FSM của bên gửi trong giao thức rdt 3.0 – giao thức truyền dữ liệu trên kênh truyền có thể có lỗi hoặc bị mất dữ liệu. Hình 3.16 minh họa sự vận hành của giao thức trong một số trường hợp. Thời gian dịch chuyển theo chiều từ trên xuống. Thời điểm nhận gói dữ liệu chậm hơn thời điểm gửi gói dữ liệu vì tính đến thời gian gói dữ liệu lan tỏa trên đường truyền. Trong Hình 3.16b-d, ngoặc vuông xác định thời điểm timer được thiết lập và thời điểm "timeout". Vì số thứ tự của gói dữ liệu thay đổi lần lượt giữa 0 và 1 nên đôi khi giao thức rdt 3.0 được gọi là giao thức một bit luân chuyển (alternate bit protocol).

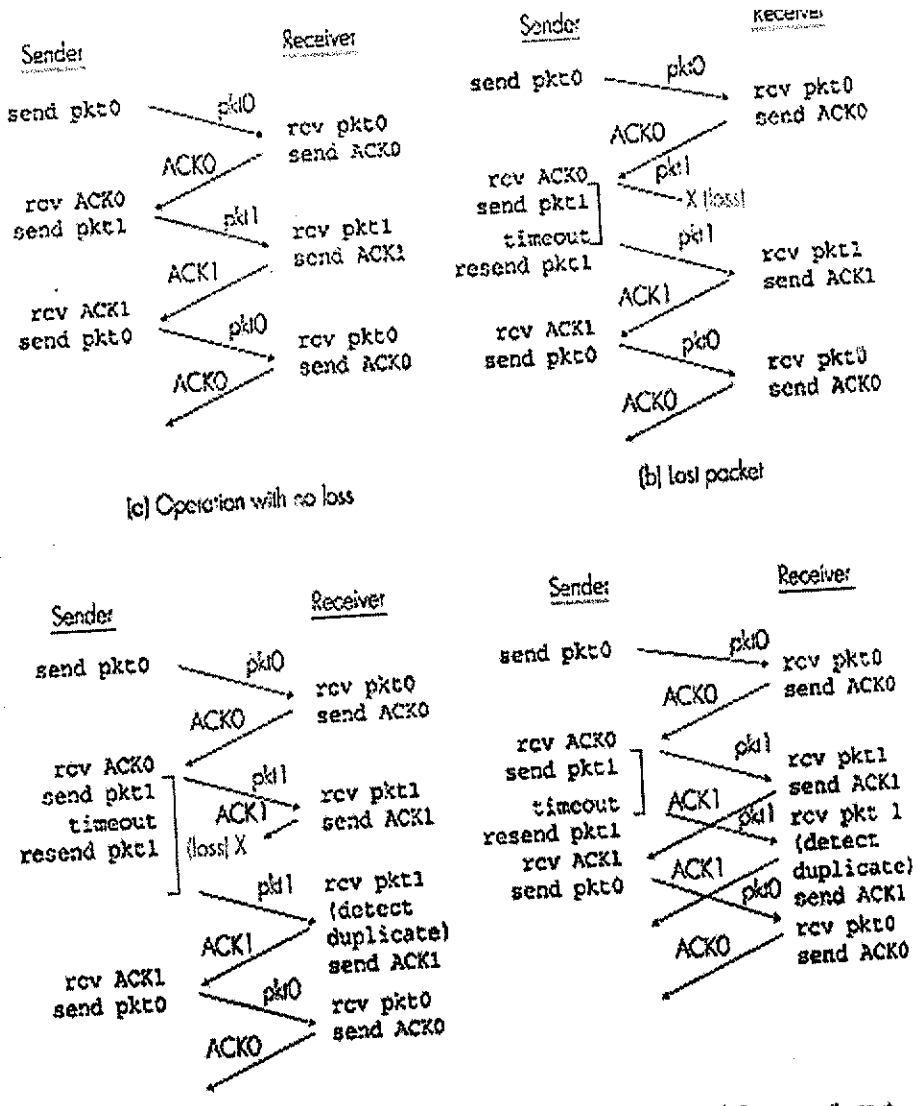
Chúng ta đã điểm qua các thành phần chính cho một giao thức truyền số liệu. Checksum, số thứ tự phát, bộ định thời (timer), các gói biên nhận ACK và NAK đều cực kỳ cần thiết và đóng vai trò quan trọng trong quá trình hoạt động của giao thức. Đến bây giờ chúng ta đã có một giao thức truyền dữ liệu tin cậy thực sự hoạt động được.



Hình 3.15 FSM của bên gửi trong rdt 3.0

.4.2 Giao thức truyền dữ liệu tin cậy liên tục (Pipeline)

Mặc dù hoạt động đúng nhưng không phải ai cũng vừa lòng với hiệu uất của rdt3.0, đặc biệt trong các mạng cao tốc ngày nay. Cốt lõi vấn đề ệu suất của giao thức rdt 3.0 chính là hành vi dừng và chờ (stop and wait). guyên tắc của giao thức kiểu "Dừng và Chờ" như sau: sau khi phát một gói liệu, thiết bị phát dừng phát (stop) để chờ nhận phản hồi từ thiết bị nhận kết quả nhận số liệu (wait). Nếu kết quả nhận tốt (biên nhận ACK), bên át được quyền phát tiếp. Nếu kết quả nhận sai (biên nhận NAK), bên phát ải gửi lại gói dữ liệu.



Hình 3.16 Ví dụ hoạt động của giao thức rdt 3.0

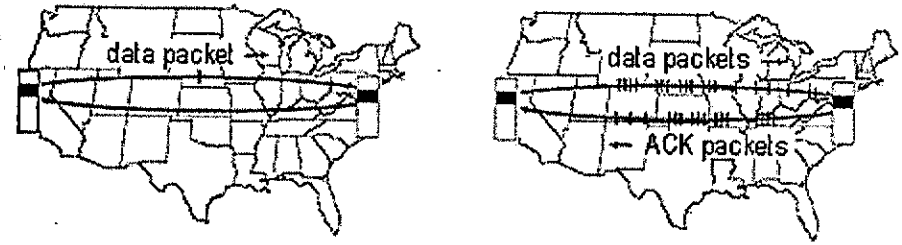
Để ước lượng hiệu suất của giao thức **stop and wait**, xét trường hợp lý tưởng với hai thiết bị đầu cuối, một ở bờ biển phía đông, một ở bờ biển phía tây nước Mỹ. Thời gian trễ giữa hai thiết bị (dù tín hiệu lan truyền với tốc độ ánh sáng) là P_{prop} xấp xỉ 15 ms. Giả sử rằng hai thiết bị được kết nối bằng đường truyền tốc độ C (1 gigabit/s). Kích thước của gói dữ liệu SP là

1Kbyte/packet, thời gian cần thiết để truyền toàn bộ gói dữ liệu trên kênh truyền tốc độ 1 Gbps được tính bởi công thức:

$$T_{trans} = \frac{SP}{C} = \frac{8Kbit/packet}{1Mbit/sec} = 8\ ms$$

Với giao thức **stop and wait**, nếu phía gửi bắt đầu gửi gói 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 gói dữ liệu đi từ phía gửi sang phía nhận (xem Hình 3.17a) 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 gói dữ liệu và phía nhận gửi ngay gói ACK khi nhận được bit cuối cùng của gói dữ liệu. Như vậy bit cuối cùng của gói ACK được truyền tới đích tại thời điểm t = 30.016 ms. Trong khoảng thời gian 30.016ms, phía gửi chỉ hoạt động (gửi hoặc nhận) trong 0.016 ms. Nếu định nghĩa **Hiệu suất (utilization)** của phía gửi (hay kênh truyền) là tỷ lệ thời gian phía gửi hoạt động (gửi dữ liệu trên kênh truyền), chúng ta có hiệu suất U_{sender} cực thấp:

$$U_{sender} = \frac{.008}{30.016} = 0.00015$$



(a) A stop-and-wait protocol in operation (b) A pipelined protocol in operation

Hình 3.17

Điều đó có nghĩa là phía gửi chỉ hoạt động trong khoảng 0.15 phần trăm thời gian. Theo cách tính khác, phía gửi gửi 1 Kbyte trong 30,016 millisecond tương đương với tốc độ truyền là 33 Kbyte/s thấp hơn nhiều so với tốc độ có thể là 1 Gigabit/s. Người quản trị mạng "bất hạnh" này phải trả một số tiền khổng lồ để thuê đường truyền 1 Gigabit/s nhưng cuối cùng chỉ nhận được một đường truyền có tốc độ 33 Kbyte/s. Đây là một ví dụ sống động minh hoạ việc phần mềm có thể giới hạn các khả năng của phần cứng bên dưới. Trong trường hợp này chúng ta đã bỏ qua thời gian xử lý của các

giao thức tầng dưới ở cả phía gửi và phía nhận cũng như thời gian xử lý và thời gian trễ của gói tin tại các router trung gian. Nếu tính cả những yếu tố này, hiệu suất hoạt động thực sự sẽ còn thấp hơn nữa.

Giải pháp cho vấn đề hiệu suất sẽ là cho phép phía gửi gửi đồng thời nhiều gói dữ liệu mà không cần phải đợi ACK (xem Hình 3.17b). Có thể hình dung các gói dữ liệu nối tiếp nhau trên đường truyền từ phía gửi đến phía nhận giống như nước chảy trong một đường ống. Vì thế kỹ thuật gửi liên tiếp này được gọi là **kỹ thuật đường ống (pipeline)**. Kỹ thuật nầy 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, bởi vì mỗi gói dữ liệu được truyền đi (không tính các gói dữ liệu truyền lại) phải có một số thứ tự duy nhất. Trên đường truyền có thể có đồng thời nhiều gói dữ liệu đã gửi nhưng chưa được biên nhận.

Phía gửi và phía nhận có thể phải có bộ đệm (buffer) cho nhiều gói dữ liệu. Ít nhất phía gửi có vùng đệm cho các gói dữ liệu đã được truyền đi nhưng chưa được biên nhận. Phía nhận cũng có thể cần vùng đệm cho cả các gói dữ liệu đã nhận đúng, như sẽ trình bày dưới đây.

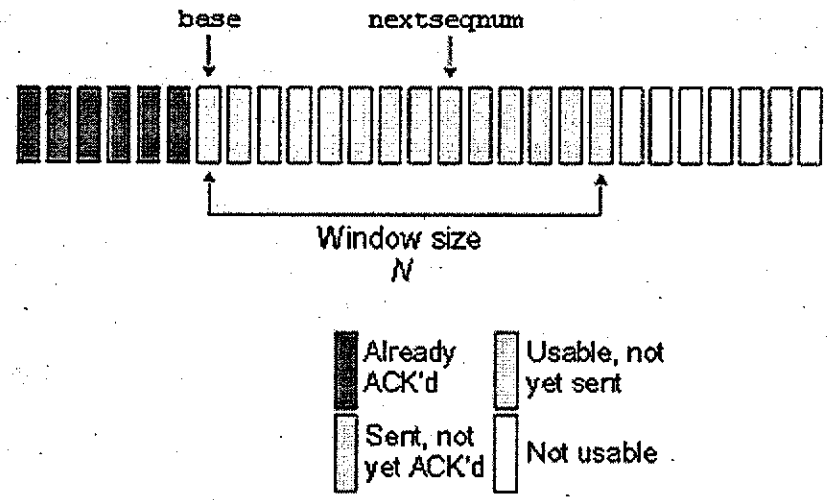
Yêu cầu về khoảng số thứ tự cần thiết cũng như về vùng đệm phụ thuộc vào cách giao thức xử lý việc mất dữ liệu, dữ liệu bị lỗi, bị trễ. Có hai cách tiếp cận chính được trình bày ở đây: **Quay lại N (Go-back-N)** và **Lặp lại có lựa chọn (SelectiveRepeat)**.

3.4.3 Go-back-N (GBN)

Trong giao thức **Go-back-N**, phía gửi được phép truyền đi đồng thời nhiều gói dữ liệu mà không phải đợi biên nhận. Tuy nhiên tổng số gói dữ liệu bị giới hạn bởi giá trị N - tổng số gói dữ liệu tối đa chưa được biên nhận trong đường ống. Hình 3.18 là khoảng số thứ tự trong giao thức **Go-back-N**. Định nghĩa **base** là số thứ tự của gói dữ liệu đã được truyền đi lâu nhất chưa được biên nhận và **nextseqnum** là số thứ tự nhỏ nhất chưa được sử dụng (là số thứ tự của gói tiếp theo sẽ gửi). Có bốn khoảng số thứ tự như sau: Khoảng [0,base-1] ứng với số thứ tự của các gói dữ liệu đã được truyền đi

và đã được biên nhận. Khoảng [base, nextseqnum-1] ứng với các gói dữ liệu đã được gửi đi nhưng chưa được biên nhận. Khoảng [nextseqnum, base +N-1] có thể được sử dụng làm số thứ tự cho các gói 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 các gói tin đợi biên nhận được biên nhận.

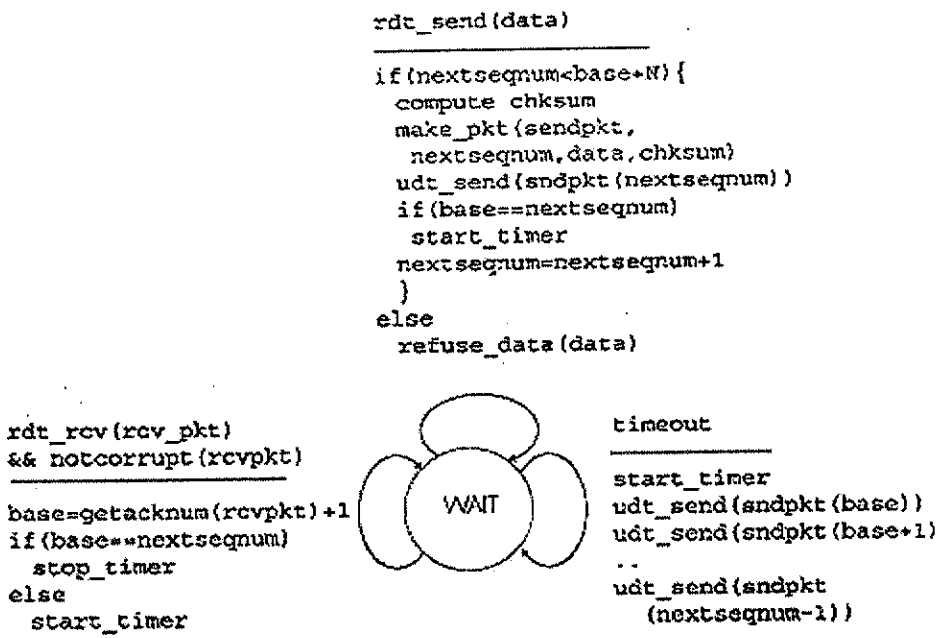
Trong Hình 3.18, khoảng số thứ tự cho phép của những gói dữ liệu đã được gửi nhưng chưa được biên nhận có thể xem là một "cửa sổ" kích thước N nằm trong phạm vi số thứ tự. Khi giao thức vận hành, cửa sổ này có thể "trượt" trên toàn bộ khoảng số thứ tự. Vì vậy, N thường được xem là **độ lớn cửa sổ (window size)** và giao thức GBN là giao thức **cửa sổ trượt (sliding-window)**. Tại sao ngay từ đầu chúng ta phải giới hạn số lượng tối đa các gói dữ liệu được gửi mà chưa cần biên nhận bởi giá trị N. Tại sao không để giá trị N này là vô hạn. Chúng ta sẽ thấy trong phần 3.5, kiểm soát lưu lượng là một trong những lý do bắt buộc ta phải đặt giới hạn phía gửi.



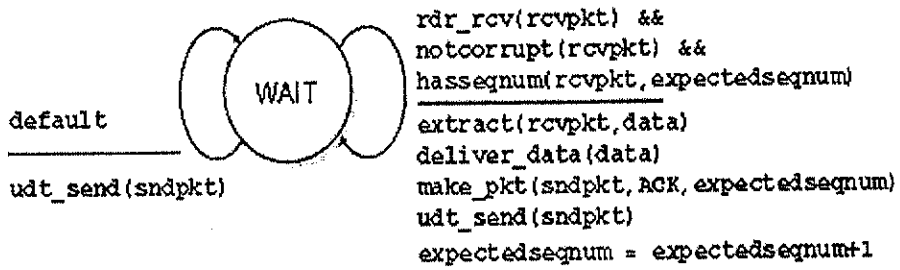
Hình 3.18 Khoảng số thứ tự của bên gửi trong giao thức Go-Back-N

Trên thực tế, số thứ tự được đặt trong một trường có độ dài cố định trong tiêu đề của gói dữ liệu Nếu k là độ lớn trường số thứ tự (tính theo bit) của gói dữ liệu thì khoảng số thứ tự sẽ là $[0,2^k-1]$. Vì khoảng số thứ tự bị giới hạn, nên tất cả các thao tác trên số thứ tự sẽ được thực hiện theo module 2^k (khoảng số thứ tự có thể xem là một vòng tròn với 2^k giá trị, sau giá trị 2^k-1 là giá trị 0). Giao thức rdt 3.0 chỉ sử dụng 1 bit làm số thứ tự nên khoảng số

thứ tự là [0,1]. Trong phần 3.5 chúng ta sẽ thấy trường số thứ tự của TCP là 32 bit, và TCP đánh số thứ tự đến từng byte - chứ không phải cho các gói.



Hình 3.19 FSM mở rộng của bên gửi trong GBN



Hình 3.20 FSM mở rộng của bên nhận trong GBN

Hình 3.19 và Hình 3.20 là FSM mở rộng của phía gửi và phía nhận trong giao thức GBN chỉ sử dụng ACK, không sử dụng NAK. Gọi là **FSM mở rộng (extended FSM)** vì chúng ta thêm vào các biến (base và nextseqnum - giống như biến trong ngôn ngữ lập trình), các lệnh và hành động có điều kiện liên quan đến các biến này.

Trong giao thức GBN, phía gửi phải đáp ứng ba sự kiện sau:

Có dữ liệu từ trên chuyển xuống: khi rdt_send() được phía trên sử dụng để chuyển dữ liệu xuống, phía gửi phải kiểm tra xem cửa sổ đã đầy chưa (tức là đã có N gói dữ liệu gửi đi chưa được biên nhận không). Nếu cửa sổ chưa đầy, phía gửi tạo ra và sau đó gửi gói dữ liệu đồng thời cập nhật các biến. Nếu cửa sổ đầy, phía gửi không chấp nhận dữ liệu từ tầng trên và thông báo cửa sổ đã đầy. Khi đó, tầng trên sẽ phải gửi lại. Trên thực tế, phía gửi sẽ đưa dữ liệu vào vùng đệm (nhưng chưa gửi ngay) hoặc có cơ chế đồng bộ (sử dụng semaphore hay cờ) chỉ cho phép tầng ứng dụng sử dụng rdt_send() khi cửa sổ chưa đầy.

Nhận được một ACK: trong giao thức GBN, giá trị biên nhận mang tính tích lũy, nghĩa là nếu biên nhận gói tin có số thứ tự n thì toàn bộ gói dữ liệu có số thứ tự nhỏ hơn hoặc bằng n đều đã được phía nhận nhận đúng. Chúng ta sẽ quay lại vấn đề này khi xem xét phía nhận trong giao thức GBN.

Hết thời gian đợi (timeout): tên giao thức - "Go-Back-N" bắt nguồn từ hành vi của phía gửi khi dữ liệu bị mất hay bị trễ. Giống như trong giao thức stop and wait, timer được sử dụng để xử lý việc mất gói dữ liệu hay gói phản hồi. Khi hết thời gian đợi (timeout), phía gửi sẽ gửi lại tất cả các gói dữ liệu đã được gửi đi trước đó nhưng chưa được biên nhận. Trong Hình 3.19, phía gửi chỉ sử dụng duy nhất một timer, có thể xem là timer của gói dữ liệu đã được truyền đi lâu nhất nhưng chưa được biên nhận. Nếu ACK nào đó được nhận nhưng vẫn còn gói dữ liệu gửi đi chưa được biên nhận thì timer sẽ được khởi động lại. Nếu tất cả các gói dữ liệu đã gửi đều được biên nhận thì có thể ngừng timer.

Các hành động của phía nhận trong giao thức GBN đơn giản. Nếu nhận được đúng gói dữ liệu và gói này đúng thứ tự thì phía nhận gửi ACK cho gói nhận được và chuyển dữ liệu trong gói dữ liệu này lên trên. Trong tất cả các trường hợp còn lại, phía nhận loại bỏ gói dữ liệu và gửi lại ACK cho gói dữ liệu đúng thứ tự cuối cùng nó nhận được. Chú ý rằng gói dữ liệu được chuyển lên tầng trên một lần duy nhất nên nếu gói dữ liệu thứ k được nhận và chuyển lên trên thì nghĩa là tất cả các gói dữ liệu có số thứ tự nhỏ hơn k cũng đã được chuyển lên. Sử dụng ACK tích lũy là sự lựa chọn tuyệt vời cho giao thức GBN.

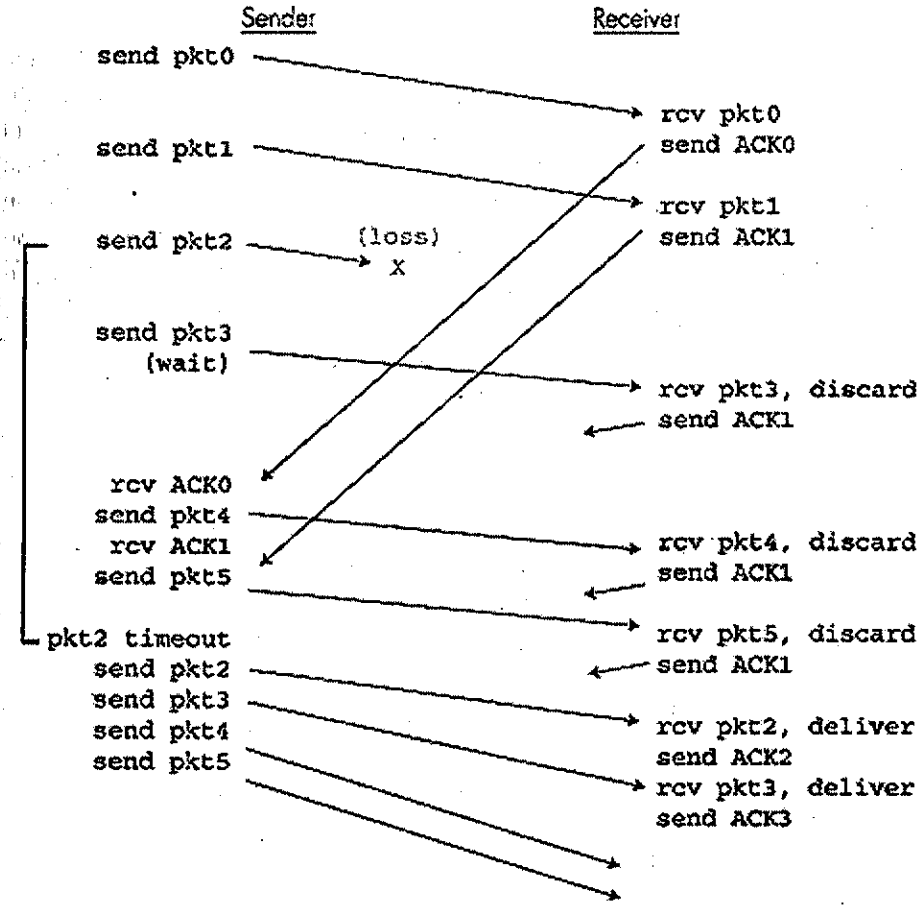
Trong giao thức GBN, bên nhận loại bỏ gói tin không theo thứ tự. Dường như lãng phí khi loại bỏ gói tin đã nhận đúng nhưng không đúng thứ tự, nhưng có vài nguyên nhân cho hành vi trên. Bên nhận phải chuyển dữ liệu lên tầng trên theo đúng thứ tự. Giả sử gói tin N đang được đợi nhận nhưng gói tin thứ (N+1) lại đến trước. Trong trường hợp ấy, để dữ liệu chuyển lên hợp lệ, bên nhận có thể lưu tạm gói tin (N+1) và chỉ chuyển gói tin này lên tầng trên sau khi đã nhận đúng gói tin thứ N. Tuy nhiên theo quy tắc truyền lại của bên gửi, nếu gói tin thứ N bị mất thì gói tin này và cả gói tin N+1 sẽ được truyền lại. Như vậy, bên nhận có thể loại bỏ gói tin N+1. Ưu điểm của giải pháp này là bên nhận triển khai vùng đệm (buffer) đơn giản bởi không cần lưu lại các gói tin không đúng thứ tự. Nếu bên gửi phải ghi nhớ các cận của cửa sổ (base, base+N) và vị trí **nextseqnum** trong cửa sổ, thì bên nhận chỉ phải nhớ số thứ tự của gói tin hợp lệ tiếp theo. Giá trị này được giữ trong biến **expectedseqnum** (số thứ tự được mong đợi) (xem Hình 3.20). Tất nhiên, nhược điểm của việc loại bỏ gói tin đã nhận đúng (nhưng không theo thứ tự) là khi truyền lại gói tin có thể bị mất hay lỗi, do đó phải truyền đi truyền lại nhiều lần.

Hình 3.21 là một ví dụ hoạt động của giao thức GBN trong trường hợp cửa sổ có độ lớn bốn gói tin. Với độ lớn này, bên gửi sẽ chỉ được gửi các gói tin từ 0 đến 3 nhưng sau đó phải đợi biên nhận cho các gói tin này trước khi tiếp tục gửi tiếp. Khi nhận được các ACK liên tiếp nhau (ví dụ ACK0 và ACK1), cửa sổ sẽ trượt về phía trước, bên gửi có thể truyền gói tin mới (lần lượt là pkt4 và pkt5). Ở phía bên nhận, gói tin số 2 bị mất, do đó gói tin 3,4,5 gửi đến không theo đúng thứ tự và bị loại bỏ.

Với GBN, có một chú ý quan trọng là triển khai GBN tương tự FSM mở rộng (Hình 3.19). Hình thức triển khai bao gồm nhiều thủ tục khác nhau, mỗi thủ tục thực hiện một nhóm các hành động nào đó đáp lại các sự kiện khác nhau có thể xảy ra. Với lập trình hướng sự kiện (**event-based programming**), các thủ tục sẽ được gọi khi sự kiện tương ứng xuất hiện. Ở phía bên gửi, sự kiện có thể là: (1) thực thể tầng trên truyền dữ liệu xuống qua thủ tục rdt_send(), (2) ngắt khi thời gian đợi hết và (3) tầng dưới chuyển dữ liệu lên qua hàm rdt_rcv( ).

Chú ý rằng giao thức GBN kết hợp hầu hết các kỹ thuật mà chúng ta sẽ gặp khi nghiên cứu TCP trong mục 3.5: số thứ tự, số biên nhận tích luỹ, checksum, timeout và việc truyền lại. Trong thực tế, TCP là giao thức "tựa" GBN. Tuy nhiên có sự khác biệt giữa GBN và TCP. Nhiều phiên bản TCP

lưu lại các segment không theo thứ tự nhận đúng. Trong phương án nâng cấp TCP, sử dụng biên nhận có lựa chọn [RFC 258] cho phép bên nhận có thể biên nhận tuỳ ý một gói tin không theo thứ tự (chứ không sử dụng giá trị biên nhận tích luỹ). Biên nhận có lựa chọn chính là lớp giao thức gửi liên tiếp thứ hai mà chúng ta sẽ nghiên cứu dưới đây: **lặp lại có lựa chọn (selective repeat - SR)** . Có thể xem TCP là sự kết hợp của cả hai giao thức GBN và SR.

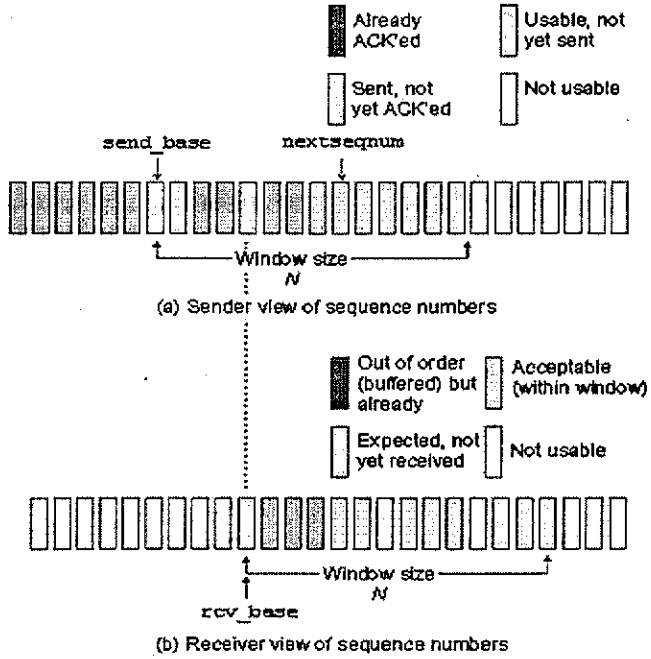


Hình 3.21 Giao thức Go-Back N trong quá trình hoạt động

3.4.4 Giao thức lặp lại có lựa chọn (Selective Repeat)

Giao thức GBN cho phép bên gửi "đổ tràn đường truyền" bằng các gói tin như trong Hình 3.17 và do đó khắc phục được hiệu suất thấp của giao thức **stop and wait**. Tuy nhiên trong một vài tình huống, chính hiệu suất của giao thức GBN cũng cực thấp. Ví dụ khi kích thước cửa sổ và thời gian truyền một gói tin lớn, có thể có nhiều gói tin ở trên đường truyền. Một gói tin bị lỗi có thể khiến GBN phải truyền lại nhiều gói tin mà trong một số trường hợp là không cần thiết. Nếu trong ví dụ đọc chính tả của chúng ta, nếu mỗi từ bị lỗi phải đọc lại khoảng 1000 từ đứng trước (kích thước cửa sổ là 1000 từ) thì tốc độ đọc sẽ rất chậm.

Giao thức lặp lại có lựa chọn (**SR - Selective Repeat**) tránh việc truyền lại không cần thiết bằng cách bên gửi chỉ truyền lại các gói tin mà nó cho là có lỗi (hoặc mất). Để truyền lại từng gói tin cần thiết, bên nhận cần biên nhận cho từng gói tin nhận đúng. Giao thức này vẫn sử dụng kích thước cửa sổ là N để giới hạn tổng số gói tin chưa được biên nhận trên đường truyền. Tuy nhiên khác với GBN, bên gửi sẽ nhận được biên nhận ACK cho một số gói tin trong cửa sổ. Hình 3.22 là không gian số thứ tự của phía gửi SR. Hình 3.23 mô tả chi tiết hành động của bên gửi trong giao thức SR.



Hình 3.22 Khoảng số thứ tự của bên gửi và bên nhận

Bên nhận Selective Repeat sẽ biên nhận cho bất kỳ gói tin nhận đúng, cho dù không theo đúng thứ tự. Gói tin không đúng thứ tự vẫn được lưu giữ lại cho đến khi tất cả các gói tin còn thiếu (gói tin có số thứ tự nhỏ hơn) được chuyển đến, khi đó tất cả các gói tin sẽ được chuyển lên tầng trên theo đúng thứ tự.

Hình 3.23 tóm tắt các hoạt động khác nhau của bên nhận trong SR.

Hình 3.25 là một ví dụ hoạt động của SR trong trường hợp mất gói tin. Trong

Hình 3.25, bên nhận sẽ lưu giữ tạm gói tin 3,4 và gửi chúng cùng với gói tin 2 lên tầng trên khi gói tin 2 được chuyển đến.

1. Dữ liệu nhận được từ phía trên

Khi nhận được dữ liệu từ phía trên, bên gửi SR kiểm tra số thứ tự sẽ gửi. Nếu số thứ tự sẽ gửi nằm trong cửa sổ gửi, dữ liệu được đóng gói và gửi đi, ngược lại thì dữ liệu được lưu giữ trong bộ đệm hoặc gửi trả lên tầng trên để gửi sau, giống GBN.

2. Hết thời gian đợi - Timeout

Timer lại được sử dụng để phát hiện mất gói tin. Tuy nhiên, mỗi gói tin gửi đi có một timer riêng, bởi vì chỉ có duy nhất một gói tin được gửi lại khi hết thời gian đợi. Có thể sử dụng đồng hồ hệ thống giữ vai trò đồng bộ cho các timer.

3. Nhận được ACK

Nếu nhận được ACK, bên gửi đánh dấu gói tin đã được chuyển đúng. Nếu số thứ tự của gói tin vừa được biên nhận bằng send_base, cánh cửa sổ sẽ trượt tới gói tin có số thứ tự nhỏ nhất chưa được biên nhận. Nếu cửa sổ di chuyển và còn các gói tin chưa được truyền thì các gói tin đó sẽ được gửi.

Hình 3.23 Sự kiện và phản ứng của bên gửi

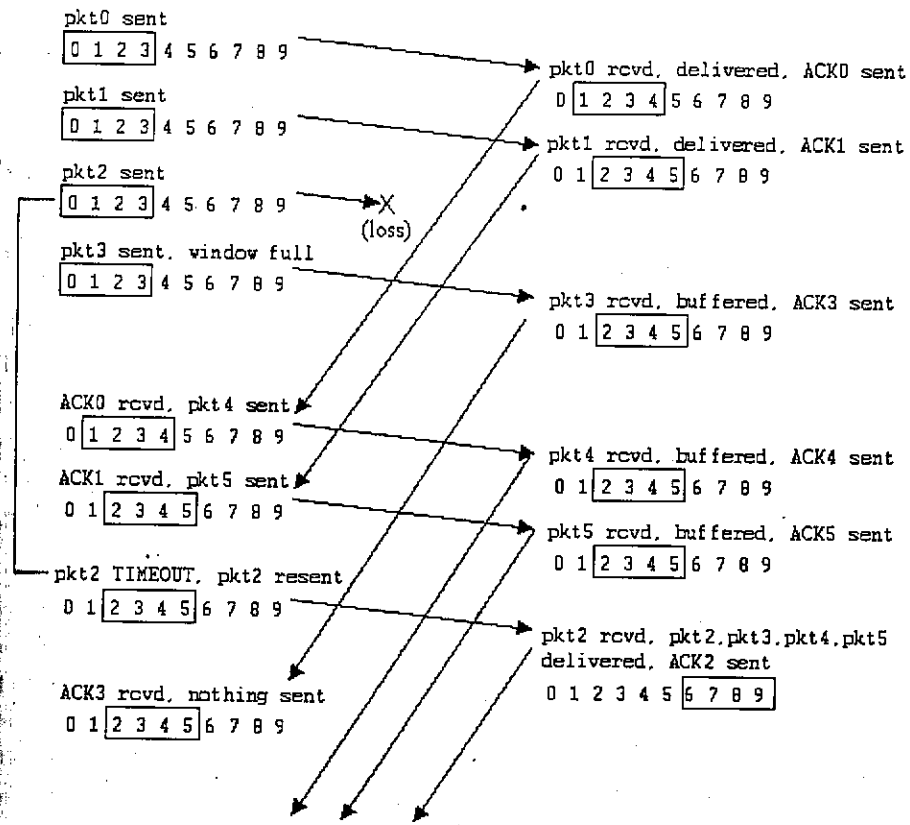
1) Nhận đúng gói tin với số thứ tự trong khoảng **[rcv_base,rcv_base+N-1**. Trong trường hợp này, gói tin nhận được nằm trong cửa sổ nhận. Bên nhận gửi biên nhận cho gói tin này. Nếu gói tin đó chưa được nhận từ trước, nó sẽ được ghi lại trong bộ đệm. Nếu gói tin đó có số thứ tự bằng với cận dưới của cửa sổ nhận (rcv_base trong hình 3.22) thì nó cùng các gói tin có số thứ tự liên tiếp đã lưu giữ từ trước (bắt đầu từ rcv_base) được chuyển lên tầng trên. Cửa sổ nhận sẽ trượt về phía trước một khoảng bằng với khoảng số gói tin đã chuyển lên tầng trên. Với ví dụ trên hình 3.25 khi nhận được gói tin có số thứ tự rcv_base=2 thì gói tin này cùng với gói tin rcv_base+1 và gói tin rcv_base+2 được chuyển lên tầng trên.

2. Nhận được gói tin với số thứ tự trong **[rcv_base-N,rcv_base-1]**. Trong trường hợp này, gửi biên nhận lại cho gói tin (mặc dù đã biên nhận từ trước).

3. ***Các trường hợp khác***. Bỏ qua gói tin đó.

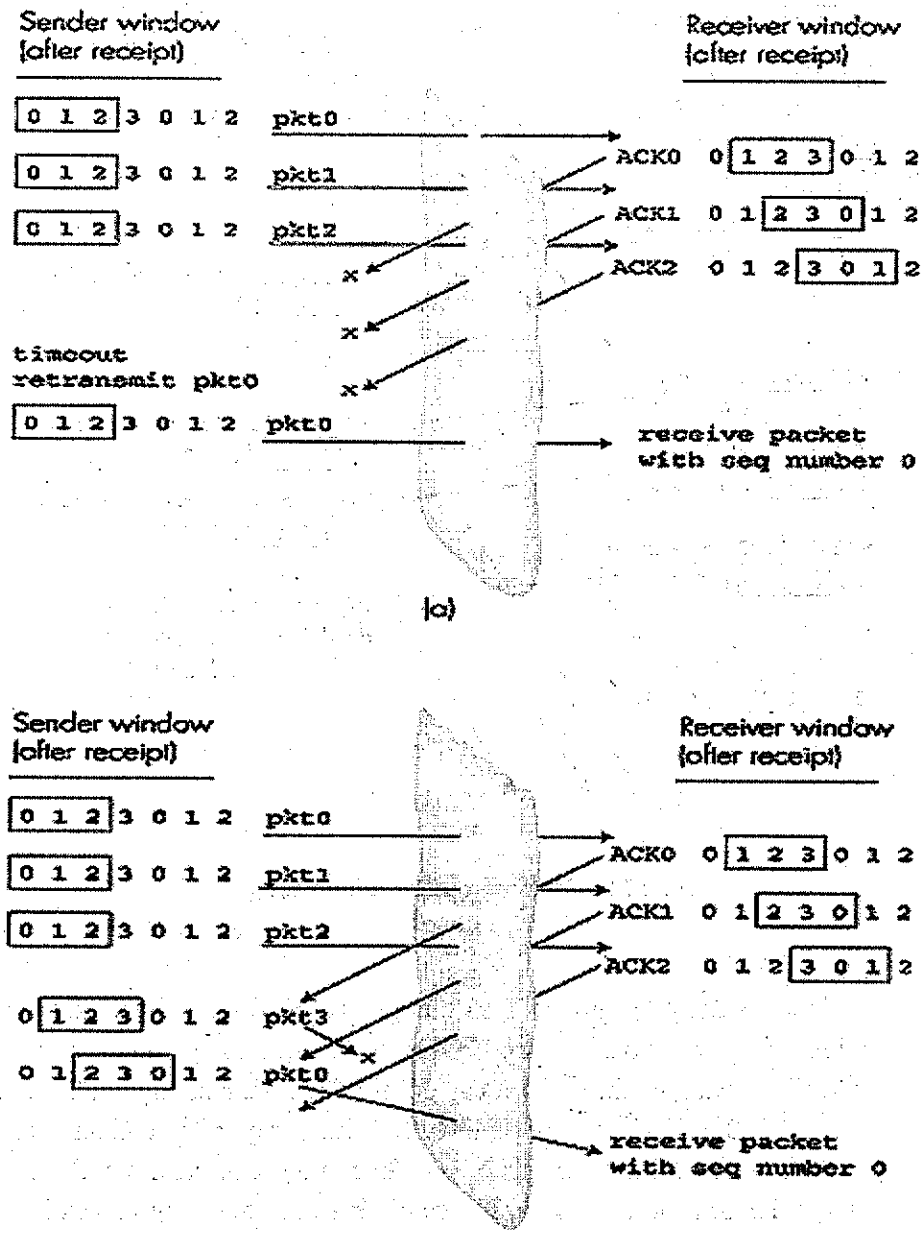
Hình 3.24 Sự kiện và phản ứng của bên nhận

Chú ý rằng ở trong bước hai trong Hình 3.23 bên nhận phải biên nhận lại (chứ không được bỏ qua) cho gói tin đến với số thứ tự nhỏ hơn giá trị biên của cửa sổ hiện thời. Điều này hết sức cần thiết. Ví dụ với không gian số thứ tự của bên gửi và bên nhận như trong Hình 3.22, nếu không nhận được ACK từ bên nhận xác nhận gói tin send_base đã được nhận, bên gửi sẽ gửi lại gói tin send_base, mặc dù rõ ràng rằng (với chúng ta, chứ không phải bên gửi) bên nhận đã nhận được gói tin đó. Nếu bên nhận không biên nhận gói tin này, cửa sổ bên gửi có thể sẽ không bao giờ trượt tới phía trước. Ví dụ này minh hoạ một đặc điểm quan trọng của giao thức SR (và nhiều giao thức tương tự khác). Sự xác định của bên gửi và bên nhận về cái gì đã được nhận, cái gì chưa được nhận không phải luôn luôn giống nhau. Với giao thức SR, điều này có nghĩa là cửa sổ bên gửi và bên nhận không bao giờ trùng khớp nhau.



Hình 3.25 SR trong quá trình hoạt động

Thiếu sự đồng bộ giữa cửa sổ bên gửi và bên nhận có thể gây hậu quả nghiêm trọng trong trường hợp khoảng số thứ tự nhỏ. Ví dụ điều gì có thể xảy ra với khoảng số thứ tự là 4, các gói tin được đánh số là 0, 1, 2 và 3, bằng lớn cửa sổ là 3. Giả sử các gói tin từ 0, 1, 2 được truyền đi và nhận chính xác tại phía bên nhận. Bên nhận gửi biên nhận cho 3 gói tin này. Khi đó, cửa sổ bên nhận tiến lên các gói tin thứ 4,5 và 6 với số thứ tự tương ứng là 3, 0, và 1. Bây giờ xem xét hai trường hợp. Trường hợp đầu tiên (Hình 3.26a) , ACK của 3 gói tin đầu tiên bị mất, bên gửi truyền lại các gói tin đó. Khi đó bên nhận nhận được tiếp theo gói tin có số thứ tự 0 - lại chính là gói tin 0 đầu tiên được gửi ban đầu.



Hình 3.26 Khi khoảng số thứ tự nhỏ: Truyền lại hay gói mới ?

Trong trường hợp thứ hai (Hình 3.26b), ACK cho ba gói tin được huyển đi thành công. Như vậy cửa sổ bên gửi sẽ trượt về phía trước và gửi i các gói tin 4, 5, và 6 với số thứ tự tương ứng là 3, 0, 1. Nếu gói tin với số ứ tự 3 bị mất, lúc ấy gói tin có số thứ tự 0 đến, gói tin này chứa dữ liệu ới (không phải gói tin 0 truyền lại).

Rõ ràng có một bức "màn chắn" giữa bên gửi và bên nhận vì bên hận không thể "nhìn" thấy hành động từ bên gửi. Bên nhận chỉ quan sát ược gói tin nào nó nhận được hay gửi đi. Hai trường hợp trong inh 3.26 là tương tự nhau. Không có phương pháp nào phân biệt được gói được truyền lại hay gói 5 được truyền lần đầu tiên. Rõ ràng nếu kích thước ửa sổ nhỏ hơn khoảng số thứ tự một đơn vị thì hệ thống không còn làm ệc đúng đắn. Nhưng độ lớn cửa sổ nên là bao nhiêu? Người ta chứng minh ược rằng độ lớn cửa sổ phải bé hơn hoặc bằng một nửa khoảng số thứ tự với ao thức SR.

Chúng ta giả thiết môi trường truyền không tin cậy ở dưới dẫn đến ệc các gói tin có thể bị giữ lại trên đường truyền. Đây là việc ít khi xảy ra i kênh truyền giữa phía gửi và phía nhận là một môi trường vật lý thực sự. ỳ nhiên khi kênh truyền này lại là một mạng máy tính thì việc một gói tin giữ lại trên kênh truyền hoàn toàn có thể xảy ra. Hệ quả của nó là xuất ện một gói tin với số thứ tự hay số biên nhận là x trong khi cả cửa sổ nhận gửi đều không chứa x. Trong trường hợp này, kênh truyền bị coi là một đệm, có thể tuỳ ý phát lại gói tin ở bất cứ thời điểm nào. Vì số thứ tự có được sử dụng lại nên trong một số trường hợp sẽ xảy ra hiện tượng trùng gói tin. Trong thực tế phải bảo đảm số thứ tự không được sử dụng lại cho khi bên gửi có thể tương đối chắc chắn về gói tin với số thứ tự x được trước đấy không còn tồn tại trong mạng. Điều này được thực hiện với thiết một gói tin không thể "tồn tại" trên mạng trong một khoảng thời n lớn hơn một khoảng thời gian cố định nào đấy. Thời gian "sống" lớn t của gói tin xấp xỉ là 3 phút với mạng TCP cao tốc [RFC 1323]. Có iều phương thức đánh số thứ tự để tránh việc xuất hiện lại gói tin.

TCP – GIAO THỨC GIAO VẬN HƯỚNG NỐI

Sau khi đã nghiên cứu những nguyên lý cơ bản của truyền dữ liệu tin , chúng ta sẽ bàn đến TCP – một giao thức tầng giao vận của Internet với tính hướng nối và tin cậy. Chúng ta sẽ thấy rằng để có thể cung cấp dịch

vụ truyền dữ liệu tin cậy, TCP áp dụng rất nhiều nguyên lý mà chúng ta đã đề cập ở phần trước, bao gồm cơ chế phát hiện lỗi, truyền lại, biên nhận tích lũy, timer, trường tiêu đề cho số thứ tự và số biên nhận. TCP được đặc tả trong các khuyến nghị RFC 793, RFC 1122, RFC 1323, RFC 2018, RFC 2581.

3.5.1 Kết nối TCP

Chức năng dồn kênh, phân kênh và phát hiện lỗi của TCP giống UDP. Tuy nhiên TCP và UDP có nhiều điểm khác biệt. Điểm khác nhau cơ bản nhất là UDP không hướng nối còn TCP hướng nối. UDP không hướng nối do có thể gửi dữ liệu mà không cần phải thiết lập trước đường truyền. TCP hướng nối vì trước khi tiến trình ứng dụng có thể bắt đầu gửi dữ liệu tới tiến trình khác, hai tiến trình này phải có thủ tục "bắt tay" với nhau, nghĩa là chúng phải gửi một số gói segment đặc biệt để xác định các tham số đảm bảo cho quá trình truyền dữ liệu. Trong giai đoạn thiết lập kết nối TCP, hai bên sẽ khởi tạo nhiều biến trạng thái TCP cho kết nối (xem mục 3.7).

"Kết nối" TCP không phải kết nối thực sự giữa hai điểm đầu mút (end-to-end) giống như mạch TDM hay FDM trong mạng chuyển mạch kênh. Nó cũng không phải là mạch ảo bởi vì trạng thái kết nối nằm hoàn toàn trên hệ thống đầu cuối. Giao thức TCP chỉ hoạt động trên thiết bị đầu cuối và không hoạt động trên các thiết bị trung gian (switch, bridge, router). Trong thực tế, các router trung gian chỉ có thể thấy các datagram, không nhìn thấy các kết nối.

Kết nối TCP cung cấp đường truyền dữ liệu **hai hướng** (song công – **full duplex**). Nếu có kết nối TCP giữa tiến trình A chạy trên một máy tính và tiến trình B chạy trên máy tính khác, khi đó dữ liệu ứng dụng có thể truyền từ A tới B cùng lúc với dữ liệu truyền từ B sang A. Kết nối TCP luôn luôn thuộc kiểu điểm nối điểm, giữa một bên gửi và một bên nhận (point to point). Chế độ truyền "multicasting" (tiến trình gửi có thể gửi đồng thời một thông điệp tới nhiều tiến trình nhận) không thực hiện được trong TCP.

Bây giờ ta hãy nhìn xem kết nối TCP được thiết lập như thế nào? Giả sử có tiến trình đang chạy trên một máy tính muốn khởi tạo đường

truyền tới tiến trình trong một máy tính khác. Nhớ lại rằng tiến trình nào khởi tạo kết nối là tiến trình khách (client) và tiến trình kia là tiến trình phục vụ (server). Đầu tiên tiến trình ứng dụng client yêu cầu thực thể TCP của mình thiết lập đường kết nối tới tiến trình nào đó trên server. Chương trình java thực hiện điều này bằng cách sử dụng mã:

Socket clientSocket = new Socket ("hostname" ,port number);

Sau đó thực thể giao vận trong máy client thiết lập kết nối TCP tới thực thể TCP trên máy phục vụ. Chúng ta sẽ thảo luận chi tiết thủ tục thiết lập đường truyền ở cuối mục này. Bây giờ chúng ta chỉ cần biết là đầu tiên máy khách sẽ gửi một gói tin TCP đặc biệt, máy server trả lời bằng một gói TCP đặc biệt thứ hai và cuối cùng client trả lời lại bằng một gói TCP đặc biệt thứ ba. Hai gói TCP đầu tiên không tải, có nghĩa là không có dữ liệu thực sự từ tầng ứng dụng, chỉ bắt đầu từ gói thứ ba mới mang dữ liệu. Vì ba gói dữ liệu đặc biệt này được trao đổi giữa hai máy tính trước khi kết nối, thủ tục thiết lập kết nối này còn được gọi là giai đoạn **bắt tay ba bước (three way handshake)**.

Trong lịch sử **Vinton Clef,Robert Kahn,và TCP/IP.**

Đầu những năm 1970, mạng chuyển mạch gói bắt đầu phát triển mạnh mẽ, và mạng APPAnet, tiền thân của Internet - chỉ là một trong các kiểu mạng như vậy. Mỗi kiến trúc mạng đều có giao thức riêng của mình. Hai nhà nghiên cứu, Vinton Cerf và Robert Kahn, nhận thấy tầm quan trọng của việc kết nối các mạng lại với nhau, đã cùng nhau phát triển giao thức liên mạng với tên gọi TCP/IP (viết tắt của Transmission Control Protocol/Internet Protocol). Mặc dù lúc đầu Cerf và Kahn xem giao thức này là một thực thể duy nhất, nhưng ngay sau đó nó được tách ra thành hai phần riêng biệt, hoạt động độc lập với nhau là TCP và IP. Cerf và Kahn đã công bố về TCP/IP vào 5/1974 trong IEEE Transacsions on Communications Technology.

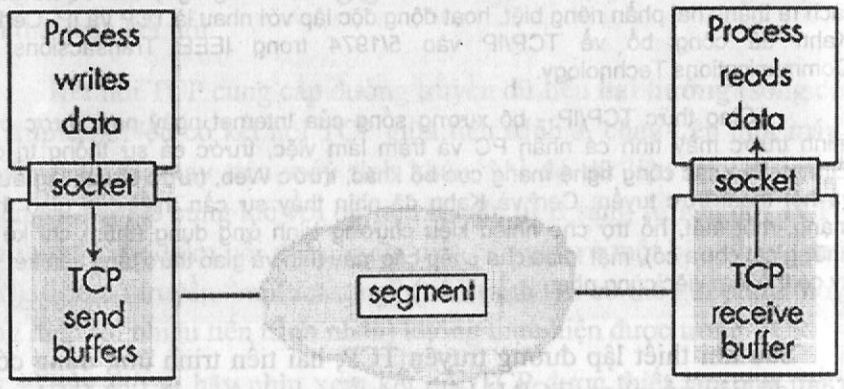
Giao thức TCP/IP - bộ xương sống của Internet ngày nay, được phát minh trước máy tính cá nhân PC và trạm làm việc, trước cả sự thống trị của Ethernet và các công nghệ mạng cục bộ khác, trước Web, trước streaming audio và hội thoại trực tuyến. Cerf và Kahn đã nhìn thấy sự cần thiết của giao thức mạng, một mặt, hỗ trợ cho nhiều kiểu chương trình ứng dụng (thậm chí kể cả những cái chưa có), mặt khác cho phép các máy tính và giao thức tầng liên kết bất kỳ có thể làm việc cùng nhau.

Sau khi thiết lập đường truyền TCP, hai tiến trình ứng dụng có thể trao đổi dữ liệu với nhau. TCP là kênh truyền song công nên máy tính có thể gửi và nhận đồng thời. Xét quá trình gửi dữ liệu từ tiến trình client tới tiến

trình server. Tiến trình client sẽ "đổ" luồng dữ liệu qua socket ("cửa" của tiến trình - xem mục 2.6). Khi đã qua cửa, tiến trình gửi sẽ không kiểm soát được dữ liệu mà chính thực thể TCP chạy trên máy client sẽ chịu trách nhiệm kiểm soát. Trong Hình 3.27, TCP đẩy dữ liệu vào bộ đệm gửi **(send buffer)**, một trong các bộ đệm được khởi tạo trong quá trình thiết lập kết nối. Sau đó TCP sẽ lấy và gửi dần dữ liệu trong bộ đệm gửi. Tuy nhiên, đặc tả TCP không xác định tường minh khi nào TCP phải gửi dữ liệu trong bộ đệm. Thường nó chỉ yêu cầu TCP "gửi khi thuận tiện". Lượng dữ liệu ứng dụng lớn nhất có thể đặt trong một segment giới hạn bởi MMS (maximum segement size). Giá trị MMS phụ thuộc vào chính phần mềm triển khai TCP (thường là hệ điều hành) và có thể cấu hình được. Các giá trị MMS phổ biến thường là 1500 byte, 536 byte hay 512 byte. (Độ lớn của segment thường được giới hạn để tránh hiện tượng phân mảnh IP, một hiện tượng sẽ được đề cập trong chương sau). Chú ý MMS là lượng dữ liệu ứng dụng lớn nhất trong segment, chứ không phải là kích thước lớn nhất của segment TCP bao gồm cả tiêu đề (header).

Thực thể TCP gói dữ liệu cùng với TCP header trong TCP segment. TCP segment được chuyển xuống dưới tầng mạng và được đặt trong gói tin của tầng mạng (IP datagram) để gửi qua mạng. Ở phía nhận, thực thể TCP sẽ đặt dữ liệu vào bộ đệm nhận **(receiver buffer)** của kết nối TCP. Ứng dụng sẽ đọc dòng dữ liệu này từ bộ đệm. Mỗi kết nối đều có bộ đệm gửi và bộ đệm nhận. Bộ đệm gửi và nhận cho dữ liệu được minh hoạ trên Hình 3.27.

Chú ý rằng bộ đệm, các biến trạng thái, và socket tạo thành kết nối TCP chỉ nằm trên hai thiết bị đầu cuối chứ không nằm trên các thiết bị trung gian (router, hub, switch...).



Hình 3.27 Bộ đệm của thực thể TCP

3.5.2 Cấu trúc TCP Segment

Sau khi đã nói qua về TCP, bây giờ chúng ta sẽ xem xét cấu trúc gói dữ liệu TCP (TCP segment). TCP segment bao gồm các trường tiêu đề và trường dữ liệu. Trường dữ liệu chứa một phần dữ liệu ứng dụng. Như đã nói ở trên, giá trị MMS giới hạn độ lớn trường dữ liệu của segment. Khi TCP gửi một file lớn - ví dụ file ảnh trong trang Web, nó phải chia file thành các đoạn có kích thước MMS (ngoại trừ đoạn cuối cùng có độ lớn bé hơn hoặc bằng MMS). Tuy nhiên độ lớn dữ liệu của các ứng dụng tương tác thường nhỏ hơn MMS. Ví dụ, với ứng dụng đăng nhập từ xa (Telnet), trường dữ liệu trong TCP segment thường chỉ là 1 byte. Độ lớn trường tiêu đề của TCP là 20 byte (của UDP là 12 byte). Segment được Telnet gửi có thể chỉ có 21 byte.

Hình 3.28 minh hoạ cấu trúc TCP segment. Tương tự UDP, tiêu đề TCP bao gồm trường số hiệu cổng nguồn, số hiệu cổng đích để thực hiện dịch vụ dồn, phân kênh dữ liệu cho các ứng dụng bên trên và trường Checksum. Tuy nhiên tiêu đề của TCP segment còn có các trường sau:

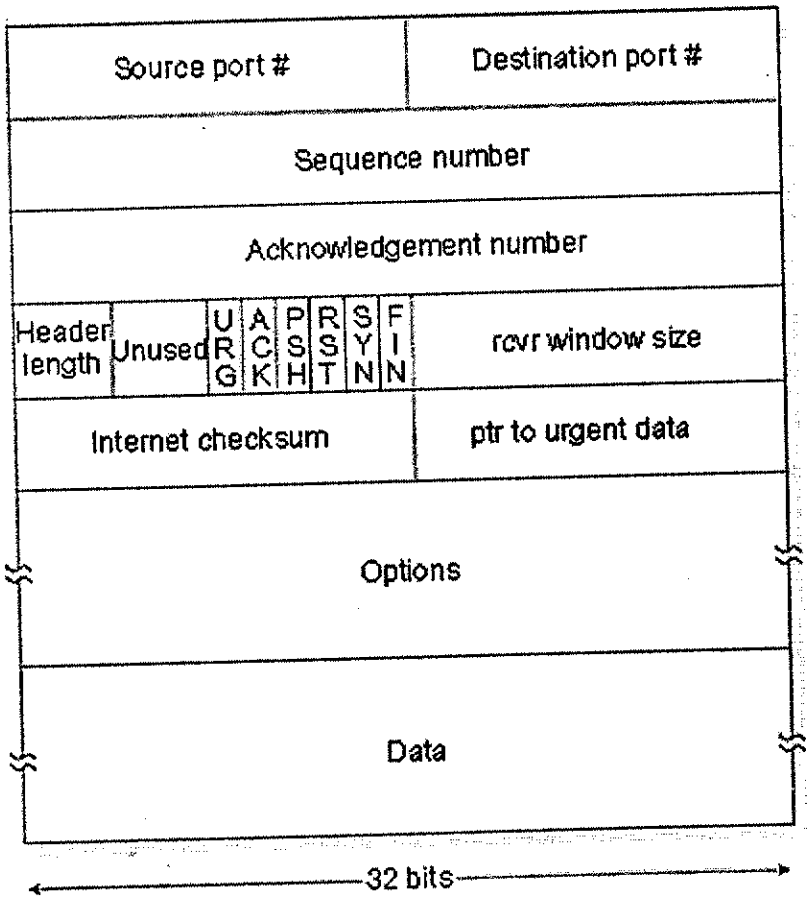
Trường **số thứ tự (sequence number)** 32 bit và trường **số biên nhận (acknowledge number)** 32 bit được bên gửi và bên nhận sử dụng trong việc cung cấp dịch vụ truyền dữ liệu tin cậy, sẽ được đề cập kĩ hơn trong phần dưới đây.

Trường **độ lớn cửa sổ (window size)** 16 bit được sử dụng để kiểm soát lưu lượng. Đấy chính là số lượng dữ liệu tối đa (tính theo byte) mà bên nhận có thể chấp nhận được.

Trường **độ dài tiêu đề (length field)** 4 bit xác định độ dài của tiêu đề TCP theo đơn vị là các từ 32 bit. Tiêu đề TCP có thể có độ dài thay đổi phụ thuộc trường option (Nếu trường option rỗng, thì chiều dài của tiêu đề TCP là 20 byte).

Trường **option** là tuỳ chọn, có thể thay đổi tuỳ ý. Trường này được sử dụng khi bên gửi, bên nhận có thể thương lượng về giá trị MMS hoặc giá trị gia tăng của cửa sổ trong mạng cao tốc. Lựa chọn nhãn thời gian (timestamping) cũng được định nghĩa. Xem RFC 854 và RFC 1323 để biết thêm chi tiết.

Trường cờ (**flag**) gồm 6 bit. Bit **ACK** được sử dụng để chỉ ra rằng giá trị đặt trong trường biên nhận là đúng. Các bit **RST**, **SYN** và **FIN** được sử dụng trong việc thiết lập hay đóng kết nối. Khi bit **PSH** được bật, thì đây là dấu hiệu để yêu cầu bên nhận phải chuyển dữ liệu lên tầng trên ngay lập tức. Cuối cùng, bit **URG** được dùng để báo hiệu dữ liệu trong segment được thực thể tầng trên phía gửi tạo ra là "khẩn cấp". Vị trí byte cuối cùng của dữ liệu khẩn cấp được xác định bởi con trỏ dữ liệu khẩn 16 bit (ptr to urgent data). TCP phải báo cho tầng trên biết có dữ liệu khẩn và đặt con trỏ vào cuối dữ liệu khẩn (Trong thực tế, PSH, URG và con trỏ dữ liệu khẩn không được sử dụng)

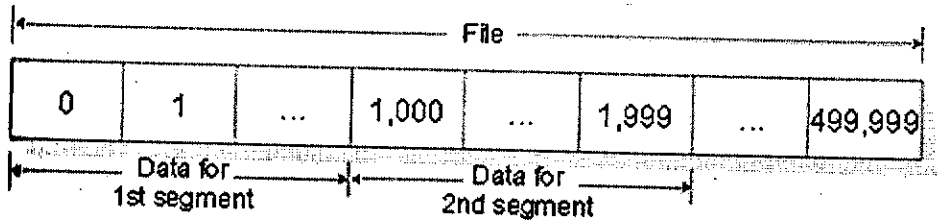


Hình 3.28 Cấu trúc gói dữ liệu TCP

3.5.3 Số thứ tự và Số biên nhận

Hai trong số những trường quan trọng nhất của tiêu đề TCP segment là trường số thứ tự và trường số biên nhận. Trước khi nói đến những trường này được sử dụng để cung cấp đường truyền dữ liệu tin cậy như thế nào, chúng ta cần nói đến những trường này nhận giá trị gì.

TCP xem dữ liệu là dòng các byte không có cấu trúc nhưng có thứ tự và TCP sẽ đánh số thứ tự cho từng byte của dòng dữ liệu này. Mỗi segment có một số thứ tự, là số thứ tự của byte đầu tiên của segment. Xét ví dụ sau: Giả sử có tiến trình trên máy A muốn gửi dòng dữ liệu tới tiến trình trên máy B thông qua kết nối TCP. Thực thể TCP trên máy A sẽ đánh số thứ tự cho từng byte trong dòng dữ liệu. Giả sử dòng dữ liệu này chứa file có kích thước 500.000 byte, giá trị MMS là 1000 byte, và byte đầu tiên của dòng dữ liệu được đánh số thứ tự 0. Trong Hình 3.29, TCP sẽ tạo ra 500 segment từ dòng dữ liệu này. Segment đầu tiên có số thứ tự 0, segment thứ hai có số thứ tự là 1000 , segment thứ ba có số thứ tự là 2000,.... Mỗi số thứ tự như vậy được chèn vào trường số thứ tự trong tiêu đề của TCP segment tương ứng.



Hình 3.29 Chia nhỏ file dữ liệu vào các TCP segment

Số biên nhận phức tạp hơn số thứ tự. Vì TCP là kênh truyền song công nên A có thể nhận được dữ liệu từ B trong khi nó gửi dữ liệu tới B (trên cùng kết nối TCP). Mỗi segment đến từ máy B có một số thứ tự cho dòng dữ liệu đi từ B sang A. ***Số biên nhận mà máy A đặt trong segment của nó sẽ là số thứ tự của byte tiếp theo mà máy A đang chờ máy B gửi tới.*** Để có thể hiểu rõ hơn chúng ta xét ví dụ sau: Giả sử rằng máy A đã nhận được tất cả các byte từ byte số 0 đến byte số 535 máy B gửi đến và giả sử máy A cũng gửi một segment tới máy B. Trong trường hợp này, máy A đợi byte thứ 536 và toàn bộ các byte tiếp theo trong dòng dữ liệu từ máy B. Khi

đó máy A đặt giá trị 536 vào trường số biên nhận của segment mà nó gửi tới máy B.

Một ví dụ khác, giả sử rằng máy A đã nhận được một segment từ máy B bao gồm byte 0 đến byte 535 và một segment khác bao gồm byte 900 đến byte 1000. Vì lý do nào đó, máy A không nhận được byte thứ 536 đến byte 899. Trong trường hợp này, máy A sẽ vẫn đợi byte thứ 536 (và các byte tiếp theo) để tạo lại dòng dữ liệu của máy B. Do đó, trong segment tiếp theo bên A gửi cho bên B, trường số biên nhận vẫn chứa giá trị 536. TCP biên nhận tất cả các byte cho đến byte đầu tiên chưa nhận được (còn thiếu trong dòng dữ liệu), cho nên có thể nói TCP biên nhận kiểu **tích luỹ (cummulative acknowledgement)**.

Ví dụ cuối cùng đưa ra vấn đề quan trọng nhưng đơn giản. Bên A nhận được segment thứ ba (byte thứ 900 đến 1.000) trước khi nhận được segment thứ hai (byte thứ 536 đến byte 899). Khi đó segment thứ ba đến không theo đúng thứ tự. Vấn đề ở đây sẽ là máy tính sẽ làm gì khi nó nhận được một segment không đúng thứ tự trong kết nối TCP ? Rất thú vị, các đặc tả RFC TCP không đưa ra bất cứ một quy tắc nào để giải quyết. Chính người lập trình phần mềm TCP sẽ đưa ra cách giải quyết. Về cơ bản, có hai lựa chọn: (1) Bên nhận ngay lập tức loại bỏ các byte không đúng thứ tự hoặc (2) bên nhận giữ lại các byte không đúng thứ tự và chờ đến khi nhận được các byte thiếu, tạo thành dòng dữ liệu liên tục. Rõ ràng giải pháp sau có hiệu quả hơn nếu đánh giá theo hiệu suất mạng, trong khi đó giải pháp thứ nhất đơn giản hóa việc cài đặt TCP. Trong quyển sách nhập môn này, chúng ta sẽ coi TCP phía nhận loại bỏ các segment không đúng thứ tự.

Trong Hình 3.30 chúng ta mặc định số thứ tự khởi tạo bắt đầu từ số 0. Nhưng trên thực tế, cả hai phía kết nối TCP đều chọn ngẫu nhiên giá trị số thứ tự khởi tạo ban đầu. Điều này sẽ giảm thiểu xác suất có một gói tin của một kết nối đã kết thúc giữa hai máy tính tồn tại quá lâu trên mạng và bị hiểu nhầm là một segment hợp lệ của một kết nối khác giữa chính hai máy tính đấy.

3.5.4 Telnet: Một ví dụ về số thứ tự và số biên nhận

Telnet đặc tả trong khuyến nghị RFC 854 là giao thức đăng nhập từ xa phổ biến ở tầng ứng dụng. Nó chạy trên nền TCP và được thiết kế để làm

việc giữa một cặp máy bất kỳ. Không giống các ứng dụng trao đổi dữ liệu đề cập trong chương hai, Telnet là ứng dụng mang tính tương tác. Chúng ta thảo luận ví dụ Telnet ở đây để có thể hiểu rõ hơn số thứ tự và số biên nhận của TCP.

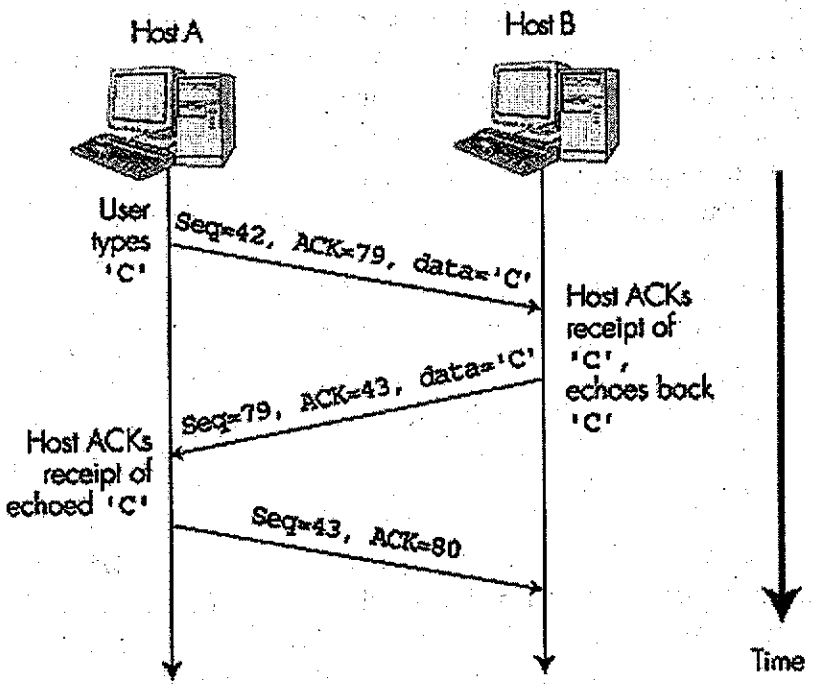
Giả sử máy A thiết lập một phiên Telnet với máy B. Máy A thiết lập phiên làm việc nên đóng vai trò client, máy B đóng vai trò server. Mỗi ký tự được người dùng gõ vào (tại phía client A) sẽ được gửi tới máy B; B sẽ gửi lại A bản sao của kí tự đó. Sau đó A sẽ hiển thị ký tự đó trong màn hình Telnet của người sử dụng. Tín hiệu "echo back" được sử dụng để chắc chắn rằng kí tự mà người sử dụng nhìn thấy đã được nhận và xử lý tại máy tính ở xa. Như vậy mỗi kí tự sẽ được truyền qua mạng hai lần trong khoảng thời gian người dùng gõ phím tới khi ký tự được hiển thị trên màn hình của người sử dụng.

Bây giờ giả sử rằng người sử dụng gõ một phím bất kì, chẳng hạn là 'C'. Hãy kiểm tra xem các TCP segment được trao đổi giữa client và server. Trên Hình 3.31, chúng ta coi số thứ tự bắt đầu của client là 42 và của server là 79. Như đã nói trên, số thứ tự của segment chính là số thứ tự của byte đầu tiên trong trường dữ liệu. Khi đó, segment đầu tiên được gửi từ phía client sẽ có số thứ tự là 42; segment đầu tiên được gửi từ phía server sẽ có số thứ tự là 79. Số biên nhận là số thứ tự của byte tiếp theo trong dòng dữ liệu đang đợi nhận. Như vậy, ngay sau khi kết nối TCP được thiết lập - nhưng trước dữ liệu thực sự được gửi, bên client sẽ đợi byte có số thứ tự 79 và bên server sẽ đợi byte có số thứ tự 42. Trên hình 3.31 ba segment đã được trao đổi. Segment đầu tiên được client gửi tới server, chứa một byte mã ASCII của ký tự 'C' trong trường dữ liệu. Trường số thứ tự của segment đầu tiên nhận giá trị 42. Vì client chưa nhận được bất kỳ dữ liệu nào từ server, nên trường số biên nhận trong segment này nhận giá trị 79.

Segment thứ hai được server gửi cho client. Có hai điểm chú ý với segment này, đầu tiên server biên nhận cho dữ liệu vừa nhận được. Với giá trị 43 trong trường số biên nhận, server báo cho client rằng nó nhận đúng tất cả các byte có số thứ tự không vượt 42 và bây giờ đang chờ nhận byte thứ 43. Thứ hai là segment này sẽ chứa lại kí tự 'C' (phản hồi lại). Lúc ấy trường dữ liệu của segment thứ hai chứa mã ASCII của ký tự 'C'. Giá trị

trường số thứ tự của segment thứ hai này là 79 - là số thứ tự của byte đầu tiên trong dòng dữ liệu chuyển từ server sang client. Chú ý rằng biên nhận cho dữ liệu từ client tới server được đặt ngay trong segment chứa dữ liệu từ server tới client. Đây gọi là biên nhận **ghép đuôi (piggybacking).**

Segment thứ ba được gửi từ client tới server. Mục đích duy nhất của nó là biên nhận dữ liệu nhận được từ server (chú ý segment thứ hai chứa dữ liệu là kí tự 'C' từ server tới client). Segment thứ ba có trường dữ liệu rỗng (biên nhận không đi kèm với dữ liệu gửi đến server). Trường số biên nhận của segment này nhận giá trị 80 vì client đã nhận được tất cả các byte có số thứ tự không vượt 79 và đang đợi byte có số thứ tự 80. Có lẽ bạn nghĩ thật không bình thường khi segment này cũng có số thứ tự khi nó chẳng chứa dữ liệu. Nhưng tiêu đề TCP có trường số thứ tự, nên phải đặt giá trị nào đó vào trong trường này.



Hình 3.31 Số thứ tự và số biên nhận trong ví dụ Telnet

3.5.5 Truyền dữ liệu tin cậy

Dịch vụ tầng mạng của Internet không tin cậy: IP không đảm bảo việc chuyển datagram, không đảm bảo gửi datagram đúng thứ tự cũng như không đảm bảo tính toàn vẹn dữ liệu. Với dịch vụ IP, datagram có thể bị tràn tại bộ đệm router và do đó không bao giờ đến được đích, dữ liệu có thể đến không đúng thứ tự hay các bit trong datagram có thể bị lỗi. Bởi vì segment của tầng giao vận được đặt trong IP datagram để truyền qua mạng nên segment của tầng giao vận cũng có thể phải gặp những vấn đề nêu trên.

TCP tạo ra đường truyền dữ liệu tin cậy trên dịch vụ không tin cậy của IP. 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 lặp dữ liệu, đúng thứ tự. Có nghĩa là dòng byte nhận được giống hệt dòng byte gửi đi. Mục này nghiên cứu cách thức cung cấp dịch vụ truyền dữ liệu tin cậy của TCP. Chúng ta sẽ thấy rằng dịch vụ truyền dữ liệu tin cậy của TCP sử dụng rất nhiều nguyên lý cơ bản đã nghiên cứu trong mục 3.4.

Hình 3.32 minh hoạ 3 sự kiện chính liên quan đến việc truyền hay truyền lại dữ liệu tại phía bên gửi TCP. Để đơn giản, chúng ta coi kết nối TCP giữa hai máy A và B chuyển dữ liệu từ máy A tới máy B. Tại phía gửi (máy A), thực thể TCP lấy dữ liệu của tầng ứng dụng, đóng gói trong các segment và chuyển xuống tầng mạng. Do đó nhận dữ liệu từ tầng ứng dụng, đóng gói dữ liệu trong các segment và gửi segment đi chính là sự kiện quan trọng đầu tiên mà thực thể TCP bên gửi phải xử lý. Ngay sau khi chuyển segment cho IP, TCP khởi động timer cho segment đó. Thời gian đợi hết (timeout) gây ra một ngắt tại máy A. TCP phản ứng với sự kiện timeout, đây chính là sự kiện thứ hai mà bên gửi TCP phải xử lý bằng cách truyền lại segment gây ra ngắt thời gian.

Sự kiện thứ ba mà bên gửi TCP phải xử lý là nhận được một segment biên nhận (ACK) từ bên gửi (chính xác hơn là một segment chứa giá trị trường biên nhận ACK hợp lệ). Ở đây, thực thể TCP phía gửi phải quyết định đó là ACK lần đầu tiên nhận được (tức là biên nhận cho một segment đã gửi nhưng chưa được biên nhận) hay chỉ là ACK trùng lặp (biên nhận lại một gói tin đã từng được biên nhận). Trong trường hợp thứ nhất thì bên gửi sẽ biết rằng tất cả các byte có số thứ tự không vượt giá trị biên nhận được nhận được đã được gửi thành công. Khi đó, bên gửi có thể cập nhật biến trạng thái TCP kiểm soát số thứ tự của byte cuối cùng mà nó cho rằng đã được nhận chính xác và theo đúng thứ tự tại phía bên nhận.

Trên thực tế

TCP cung cấp đường truyền tin cậy bằng cách sử dụng ACK và timer giống như trong mục 3.4. TCP biên nhận cho dữ liệu đã được nhận chính xác, và truyền lại segment nếu cho rằng segment hay biên nhận tương ứng của nó bị mất hoặc có lỗi. Phiên bản hiện thời của TCP cũng có cơ chế NAK ẩn, đấy là cơ chế truyền lại nhanh của TCP khi nhận được ba ACK cho cùng một segment được gửi. TCP sử dụng số thứ tự cho phép bên nhận xác định segment bị mất hoặc trùng lặp. Cái này tương tự như trường hợp giao thức truyền dữ liệu tin cậy của chúng ta. TCP không chắc chắn được segment hay biên nhận của nó bị lỗi, bị mất hay chỉ đến trễ, nó xử lý giống hệt nhau: truyền lại.

TCP cũng thực hiện việc gửi liên tục (cơ chế đường ống), cho phép bên gửi có thể gửi nhiều segment mà chưa cần nhận biên nhận ngay. Cơ chế này cho phép nâng cao hiệu suất của đường truyền. Số lượng tối đa các segment được gửi chưa cần biên nhận phụ thuộc vào cơ chế kiểm soát lưu lượng và kiểm soát tắc nghẽn của TCP. Cả hai cơ chế này đều được nghiên cứu trong các phần sau.

Để hiểu về phản ứng của bên gửi khi nhận được ACK trùng lặp, đầu tiên chúng ta phải xét tại sao bên nhận gửi ACK trùng lặp. Bảng 3.1 tóm tắt các chính sách chung của thực thể TCP nhận. Khi nhận được segment có số thứ tự lớn hơn số thứ tự đang được mong đợi, bên nhận phát hiện có đoạn trống trong dòng dữ liệu - nghĩa là thiếu segment. Vì TCP không sử dụng biên nhận phủ định (NAK) nên bên nhận không thể gửi biên nhận phủ định. Thay vào đó, nó biên nhận lại byte đúng thứ tự cuối cùng mà nó nhận được (tạo ra ACK trùng lặp). Nếu bên gửi TCP nhận được 3 ACK trùng lặp cho cùng một segment, nó sẽ cho rằng segment ngay sau segment được biên nhận ba lần bị mất. Trong trường hợp này, TCP thực hiện cơ chế truyền lại nhanh **(fast retransmit)** [RFC 258], gửi lại segment bị cho là mất trước khi timer của segment đó hết hạn (kết thúc).

```
/*assume sender is not constrained by TCP flow or
congestion control,that data from above is less than MSS in
size, and that data transfer is in one direction only
*/
sendbase=initial_sequence number /*see Figure 3.18*/
nextseqnum=initial_sequence number
```



```
loop (forever) {
   switch(event)

vent: data received from application above
reate TCP segment with sequence number nextseqnum
          start timer for segment nextseqnum
          pass segment to IP
          nextseqnum=nextseqnum+length(data)
reak;  /* end of event data received from above */

vent: timer timeout for segment with sequence number y
          retransmit segment with sequence number y
          compute new timeout interval for segment y
          restart timer for sequence number y
          break;  /* end of timeout event */

ent: ACK received, with ACK field value of y
 (y > sendbase) {/* cumulative ACK of all date up to y */
ncel all timers for segments with sequence numbers < y
ndbase=y
               }
e { /* a duplicate ACK for already ACKed segment */
rement number of duplicate ACKs received for y
(number of duplicate ACKs received for y==3) {
                    /* TCP fast retransmit */
                    resend segment with sequence number y
                    restart timer for segment y
               }
ak; /* end of ACK received event */
* end of loop forever */
```

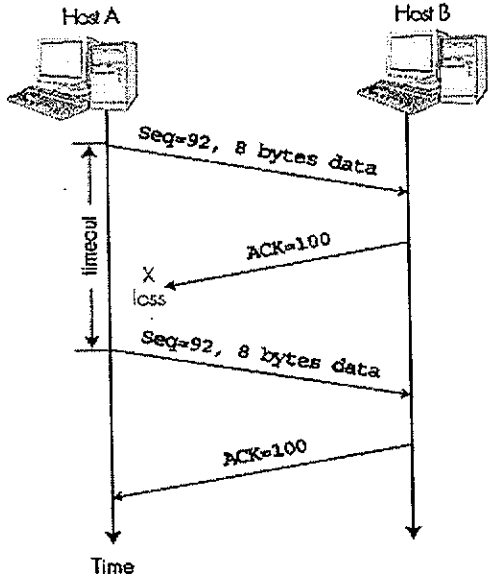
Hình 3.31 Bên gửi của TCP

Sự kiện	Hành động tiếp nhận của TCP
gment đến có số thứ tự là số thứ mong muốn. Tất cả dữ liệu đến thứ tự mong muốn đã được biên n. Không có khoảng trống trong liệu nhận được	Trì hoãn ACK. Đợi segment đúng thứ tự tiếp theo trong khoảng thời gian 500ms. Nếu segment này không xuất hiện mới gửi ACK
gment đến có số thứ tự là số thứ mong muốn. Segment đến trước g đợi gửi biên nhận. Không có ng trống trong dữ liệu nhận được	Ngay lập tức gửi đi ACK tích lũy duy nhất biên nhận cho cả hai segment đúng thứ tự.

Segment không đúng thứ tự đến, có số thứ tự cao hơn số thứ tự mong muốn nhận. Phát hiện có khoảng trống dữ liệu.	Ngay lập tức gửi đi ACK trùng lặp và chỉ ra số thứ tự của byte mong muốn nhận tiếp theo.
Segment đến lấp đầy một phần hoặc toàn bộ khoảng trống trong dữ liệu nhận được	Ngay lập tức gửi đi ACK biên nhận cho đoạn dữ liệu đúng thứ tự liên tục lớn nhất nhận được

Hình 3.32

Một vài trường hợp tiêu biểu

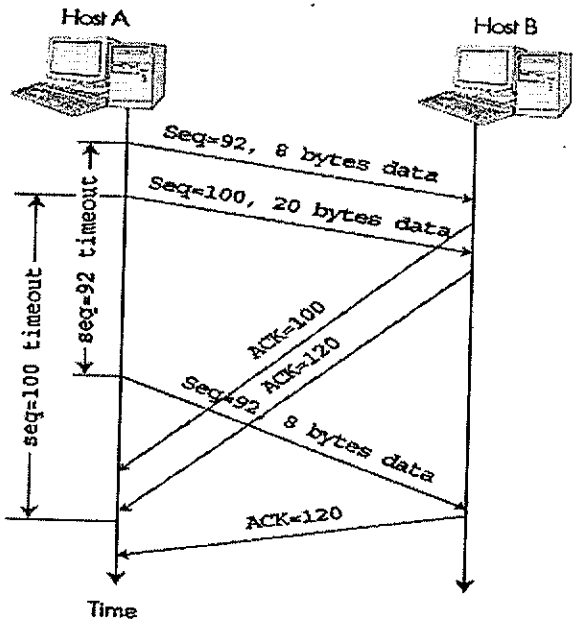


Hình 3.33 Truyền lại vì mất ACK

Chúng ta kết thúc trình bày ở đây để xem xét vài trường hợp đơn giản. Hình 3.33 mô tả trường hợp máy tính A gửi một segment tới máy tính B. Giả sử segment này có số thứ tự là 92 và có 8 byte dữ liệu. Sau khi gửi, máy A chờ segment ACK với giá trị biên nhận 100 từ máy B. Mặc dù segment gửi từ máy A đã đến máy B nhưng ACK gửi từ máy B đến máy A bị mất. Trong trường hợp này, khi hết thời gian đợi, máy A truyền lại một segment giống hệt cho B. Dĩ nhiên khi nhận được segment truyền lại, máy B sẽ phát hiện sự trùng lặp nhờ trường số thứ tự. Vì vậy thực thể TCP trên máy B sẽ loại bỏ segment truyền lại.

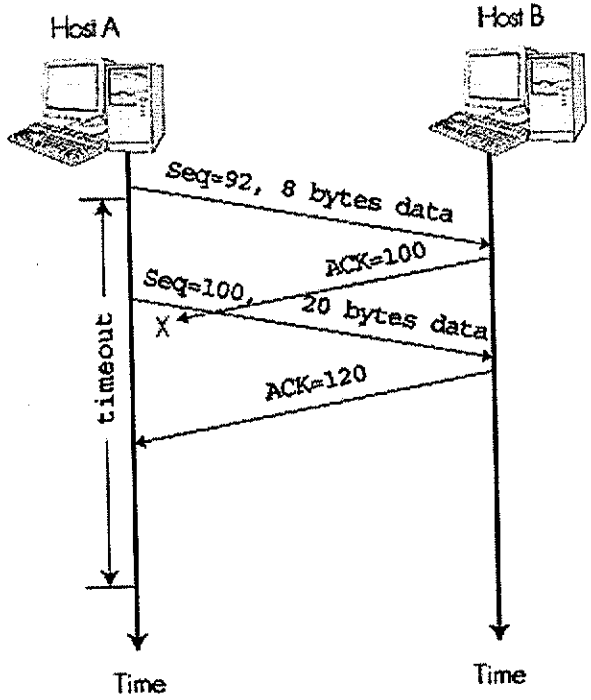
Trong trường hợp thứ hai, máy A gửi hai segment liên tiếp. Segment đầu tiên có số thứ tự là 92 và 8 byte dữ liệu, segment thứ hai có số thứ tự là 100 và 20 byte dữ liệu. Giả sử cả hai segment này đều đến máy B nguyên vẹn và máy B gửi biên nhận ACK riêng rẽ cho từng segment. ACK cho segment đầu tiên có số biên nhận là 100 và cho segment thứ hai là 120. Lại giả sử rằng cả hai ACK đều không đến được máy A trước khi hết thời gian đợi của segment đầu tiên. Khi hết thời gian đợi, máy A gửi lại segment đầu tiên có số thứ tự 92. Vậy máy A có gửi lại segment thứ hai không ? Theo quy tắc mô tả trên, máy A chỉ gửi lại segment thứ hai nếu hết thời gian đợi trước khi ACK có số biên nhận 120 hoặc lớn hơn đến. Vì vậy, như minh hoạ trên hình 3.33, nếu ACK thứ hai không mất và đến trước timeout của segment thứ hai thì máy A sẽ không phải gửi lại segment thứ hai .

Trong trường hợp thứ ba và cũng là trường hợp cuối cùng, giả sử máy tính A gửi hai segment giống như trong ví dụ hai. ACK của segment đầu tiên bị mất, nhưng trước khi hết thời gian đợi của segment đầu tiên, máy A nhận được ACK có số biên nhận 120 - do đó máy A hiểu rằng máy B đã nhận được tất cả các byte đến tận byte thứ 119, vì vậy máy A không phải gửi lại segment nào trong hai segment. Trường hợp này được minh hoạ trong Hình 3.34.



Hình 3.34 Segment không cần truyền lại vì ACK đến trước khi hết thời gian đợi

Mặc dù trong phần trước chúng ta đã nói TCP là giao thức kiểu Go-Back-N vì các giá trị biên nhận mang tính tích luỹ và bên nhận không biên nhận cho các segment đã nhận đúng nhưng không theo số thứ tự. Kết quả là (xem Hình 3.31 và Hình 3.18) TCP bên gửi chỉ cần ghi nhớ số thứ tự nhỏ nhất của byte đã được gửi nhưng chưa được biên nhận (**sendbase**) và số thứ tự cho byte tiếp theo sẽ được gửi đi (**nextseqnum**). Tuy nhiên cần lưu ý rằng mặc dù thành phần truyền dữ liệu tin cậy của TCP giống Go-Back-N, nhưng không phải giống hoàn toàn. Để phân biệt một số điểm khác nhau giữa TCP và Go-Back-N, chúng ta hãy xem điều gì sẽ xảy ra khi bên gửi gửi các segment liên tiếp 1, 2,..., N, tất cả segment này đều được nhận đúng thứ tự và không có lỗi. Giả sử ACK của segment n<N bị mất nhưng ACK của N-1 segment còn lại đến bên nhận trước khi hết thời gian đợi của từng segment. Trong trường hợp này, Go-Back-N sẽ truyền lại không chỉ packet n mà còn là tất cả những gói tin sau n+1, n+2, ..., N. TCP sẽ truyền lại nhiều nhất là segment thứ n. Thậm chí TCP sẽ không truyền lại segment thứ n nếu ACK cho segment thứ n+1 đến trước khi hết thời gian đợi (timeout) của segment thứ n.

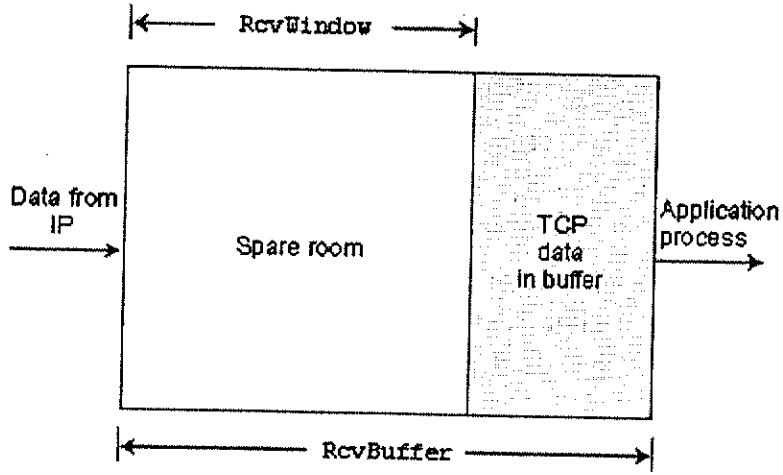


Hình 3.35 ACK tích luỹ tránh việc truyền lại segment đầu tiên

Gần đây có một số đề xuất [RFC 2018] mở rộng cơ chế biên nhận của TCP cho giống kiểu giao thức Selective Repeat. Ý tưởng chính trong những đề xuất này là cung cấp cho bên gửi những thông tin tường minh về segment nào đã được nhận đúng và segment nào chưa nhận được.

3.5.6 Kiểm soát lưu lượng

Nhắc lại rằng, thiết bị đầu cuối ở mỗi phía của kết nối TCP đều có bộ đệm dữ liệu (buffer). Khi nhận được đúng một dòng byte liên tục (đúng thứ tự), TCP sẽ đặt dòng byte này vào bộ đệm nhận (receive buffer). Tiến trình ứng dụng nhận sẽ đọc dữ liệu từ bộ đệm này, nhưng không nhất thiết là phải đọc ngay khi dữ liệu đến. Có thể tiến trình ứng dụng nhận phải thực hiện nhiều tác vụ khác nên chưa đọc ngay dữ liệu trong bộ đệm. Nếu ứng dụng đọc dữ liệu chậm thì bên gửi có thể làm tràn bộ đệm nhận do dữ liệu được gửi quá nhiều và quá nhanh. Chính vì lý do này TCP cung cấp dịch vụ **kiểm soát lưu lượng (flow control)** để tránh hiện tượng bên gửi làm tràn bộ đệm bên nhận. Kiểm soát lưu lượng là quá trình làm tương thích (matching) về tốc độ: tương thích giữa tốc độ gửi và tốc độ nhận . Như đã lưu ý ở phần trước, bên gửi TCP cũng bị giới hạn do tắc nghẽn trong mạng IP, đây chính là cơ chế **kiểm soát tắc nghẽn (congestion control)** của TCP. Mặc dù kiểm soát lưu lượng giống kiểm soát tắc nghẽn (hạn chế tốc độ gửi của bên gửi), tuy nhiên chúng được thực hiện với những mục đích khác nhau. Nhiều người coi hai thuật ngữ này tương đương nhau, vì vậy người đọc nên xem kỹ để phân biệt hai trường hợp.



Hình 3.36 Biến receive window và bộ đệm nhận

Để cung cấp cơ chế kiểm soát lưu lượng, TCP bên gửi sử dụng biến **receive window**. Đây là giá trị mà bên nhận báo cho bên gửi biết độ lớn vùng đệm còn rỗi của mình. Trong kết nối hai hướng, ở mỗi phía kết nối có một giá trị **receive window** riêng. Giả trị **receive window** thay đổi trong thời gian kết nối. Chúng ta hãy nghiên cứu giá trị receive window trong ví dụ truyền file. Giả sử máy A gửi một file lớn tới 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. Chúng ta định nghĩa một số biến sau:

LastByteread = số thứ tự của byte cuối cùng trong dòng dữ liệu mà tiến trình ứng dụng trong máy B đọc từ buffer

LastByteRcvd = số thứ tự byte cuối cùng trong dòng dữ liệu đến từ mạng và được để trong receive buffer trên máy B

Vì TCP không cho phép tràn bộ đệm nên:

LastByteRcvd – LastByteread <= RcvBuffer

Receive window là giá trị RcvWindow, là độ lớn vùng đệm rỗi:

RcvWindow =RcvBuffer - [LastByteRcvd - LastByteread]

Bởi vì độ lớn vùng đệm rỗi thay đổi theo thời gian nên giá trị **RcvWindow** cũng biến đổi. Biến **RcvWindow** được minh hoạ trong Hình 3.35.

Kết nối sử dụng biến **RcvWindow** để cung cấp dịch vụ kiểm soát lưu lượng như thế nào? Máy B báo cho máy A độ lớn vùng rỗi trong bộ đệm của mình bằng cách đặt giá trị **RcvWindow** hiện thời vào trong trường **window** của tất cả các segment gửi tới A. Ban đầu máy B thiết lập **RcvWindow=RcvBuffer.** Rõ ràng để đạt được điều này thì máy B phải kiểm soát vài biến kết nối .

Máy A cũng có hai biến **LastByteSent** và **LastByteAcked**. Độ lệch giữa hai biến này, **LastByteSent – LastByteAcked** là số lượng dữ liệu chưa được biên nhận mà A gửi qua kết nối. Bằng cách khống chế số lượng dữ liệu chưa được biên nhận nhỏ hơn giá trị RcvWindow, A đảm bảo không làm tràn bộ đệm tại B. Do vậy trong suốt thời gian kết nối, A phải đảm bảo:

LastByteSent – LastByteAcked ≤RcvWindow

Một vấn đề kỹ thuật nhỏ nẩy sinh ở đây. Giả sử bộ đệm ở máy B đầy, có nghĩa là RcvWindow = 0. Sau khi thông báo tới máy A là RcvWindow = 0, máy B không có gì để gửi tới máy A. Khi tiến trình ứng dụng ở máy B lấy dữ liệu lên làm cho bộ đệm rỗng thì TCP không gửi segment mới cùng với giá trị RcvWindow mới tới máy A – TCP chỉ gửi segment tới A khi có dữ liệu hoặc ACK để gửi. Bởi vậy, máy A sẽ không bao giờ được thông báo đã có thêm khoảng trống trong bộ đệm ở B. Máy A bị "khoá" và không thể truyền thêm dữ liệu. Để giải quyết vấn đề này, đặc tả TCP yêu cầu máy A tiếp tục gửi segment với một byte dữ liệu khi receive window của máy B bằng 0. Những segment này sẽ được B biên nhận. Khi bộ đệm bắt đầu có vùng rỗng thì trong gói biên nhận sẽ có cả giá trị khác 0 của RcvWindow.

Khác TCP, UDP không có cơ chế kiểm soát lưu lượng. Để hiểu vấn đề này, chúng ta hãy xem thực thể UDP gửi các segment từ tiến trình trên máy A tới tiến trình trên máy B. UDP sẽ đặt các segment (chính xác hơn là dữ liệu trong segment) vào trong một hàng đợi có độ lớn hữu hạn ứng với socket nào đó ("cửa" của tiến trình). Tiến trình đọc lần lượt từng segment trong hàng đợi. Nếu tốc độ đọc của tiến trình không đủ nhanh thì hàng đợi bị tràn và các segment đến sau sẽ bị mất.

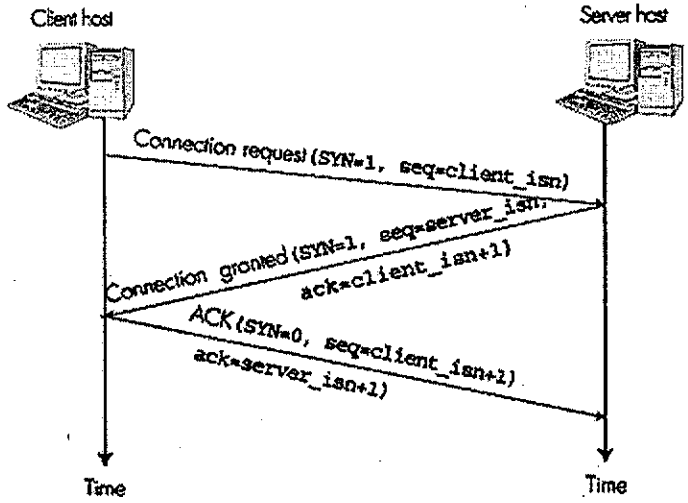
3.5.7 Quản lý kết nối TCP

Trong phần này chúng ta xem xét một kết nối TCP được thiết lập và giải phóng như thế nào. Mặc dù đây là vấn đề không hấp dẫn nhưng lại quan trọng bởi vì giai đoạn thiết lập kết nối TCP ảnh hưởng lớn đến độ trễ (chẳng hạn như khi duyệt Web). Bây giờ chúng ta xem một kết nối TCP được thiết lập như thế nào? Giả sử tiến trình chạy trên máy client muốn khởi tạo một kết nối tới tiến trình trên server. Đầu tiên tiến trình ứng dụng trên client yêu cầu thực thể TCP của nó (client) thiết lập một kết nối tới một tiến trình trên server. Sau đó thực thể TCP client khởi tạo kết nối TCP tới thực thể TCP trên server qua những bước sau:

Bước 1: Đầu tiên phía TCP client gửi một segment đặc biệt tới TCP server. Segment đặc biệt này không chứa dữ liệu của tầng ứng dụng nhưng cờ SYN (một bit thuộc trường cờ (flag)) trong phần tiêu đề được đặt giá trị 1. Vì thế đôi khi segment đặc biệt này được gọi là SYN segment. Ngoài ra

TCP client chọn số thứ tự ban đầu (client_isn) và đặt giá trị này vào trường số thứ tự của SYN segment. Segment này được đặt trong IP datagram để gửi tới server.

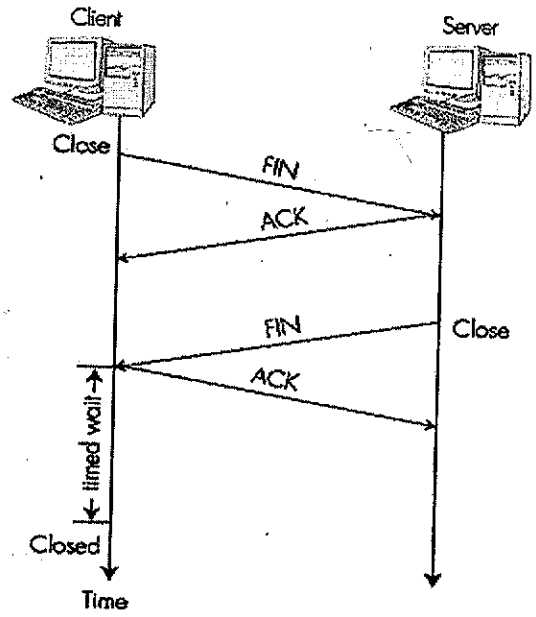
Bước 2: Khi IP datagram chứa TCP segment đến server (nếu đến được) thì server lấy SYN segment ra khỏi datagram, phân phối bộ đệm và các biến TCP phục vụ kết nối đồng thời gửi đi một segment đặc biệt thông báo chấp nhận kết nối từ client. Segment này cũng không chứa dữ liệu của tầng ứng dụng. Tuy nhiên nó chứa ba thông tin quan trọng trong phần tiêu đề. Thứ nhất bit SYN sẽ được thiết lập giá trị 1. Thứ hai, trường biên nhận trong tiêu đề nhận giá trị client_isn+1. Cuối cùng, server chọn số thứ tự bắt đầu của mình (server_isn) và đặt giá trị này vào trường số thứ tự trong tiêu đề của segment. Với segment chấp nhận kết nối, server ngụ ý "đã nhận được từ client gói SYN yêu cầu thiết lập kết nối với số thứ tự bắt đầu từ client_isn. Chấp nhận thiết lập kết nối này. Số thứ tự của server bắt đầu từ server_isn". Đôi khi đây được gọi là SYNACK segment.



Hình 3.37 Giai đoạn bắt tay ba bước trong thiết lập đường truyền của TCP

Bước 3: Khi nhận được segment chấp nhận kết nối, client cũng khởi tạo bộ đệm và các biến phục vụ kết nối. Client gửi segment thứ ba biên nhận cho segment chấp nhận kết nối của server (bằng cách đặt giá trị server_isn+1 vào trường số biên nhận trong tiêu đề của TCP segment). Bit SYN được đặt giá trị 0 vì kết nối đã được thiết lập.

Sau khi thực hiện xong ba bước này thì client và server có thể trao đổi các segment chứa dữ liệu. Bit SYN trong các segment sau được đặt giá trị 0. Như vậy, để thiết lập được kết nối hai máy phải trao đổi ba segment (xem Hình 3.37). Vì thế thủ tục kết nối được xem là **quá trình bắt tay ba bước (three way handshake)**.

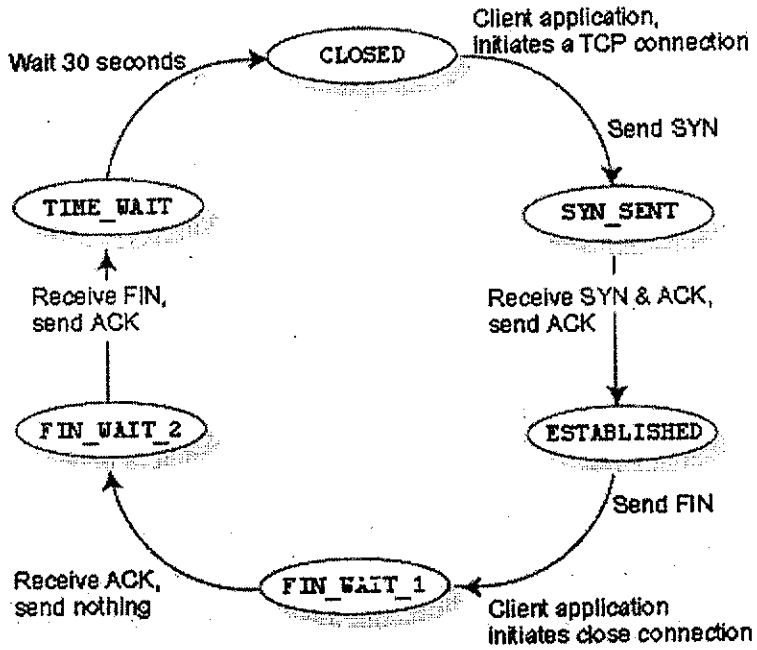


Hình 3.38 Kết thúc kết nối TCP

Bây giờ chúng ta xét đến việc đóng kết nối TCP. Cả hai tiến trình tham gia kết nối TCP đều có thể kết thúc kết nối. Khi kết nối đóng lại thì các tài nguyên phục vụ kết nối (bộ đệm và các biến TCP) trong máy được giải phóng. Ví dụ client quyết định đóng kết nối (Hình 3.38). Tiến trình ứng dụng client sẽ đưa ra lệnh đóng. Khi đó TCP client gửi một segment TCP đặc biệt đến tiến trình server. Đây là FIN segment vì cờ FIN trong segment này được đặt giá trị 1. Khi server nhận được segment FIN, nó sẽ gửi lại cho client một segment ACK biên nhận segment FIN của client. Kế tiếp server gửi lại một segment kết thúc FIN (có cờ FIN được đặt giá trị 1). Cuối cùng client gửi segment ACK biên nhận segment FIN của server. Tại thời điểm này thì tất cả tài nguyên trên hai máy đều được giải phóng.

Trong suốt thời gian kết nối TCP, giao thức TCP chạy trên mỗi máy chuyển qua các trạng thái TCP (TCP state). Hình 3.39 minh hoạ quá trình

thay đổi trạng thái TCP xảy ra bên phía client. TCP client bắt đầu ở trạng thái đóng (CLOSED). Ứng dụng bên phía client khởi tạo một kết nối TCP. Điều này đòi hỏi TCP client gửi SYN segment tới TCP server. Sau khi gửi SYN segment, TCP client chuyển sang trạng thái SYN_SENT. Trong trạng thái SYN_SENT, TCP client đợi SYNACK segment (biên nhận cho SYN segment của mình). Khi nhận được segment này, TCP client chuyển sang trạng thái ESTABLISHED. Ở trạng thái ESTABLISHED, TCP client có thể gửi và nhận TCP segment chứa dữ liệu (là dữ liệu thực sự do ứng dụng tạo ra).



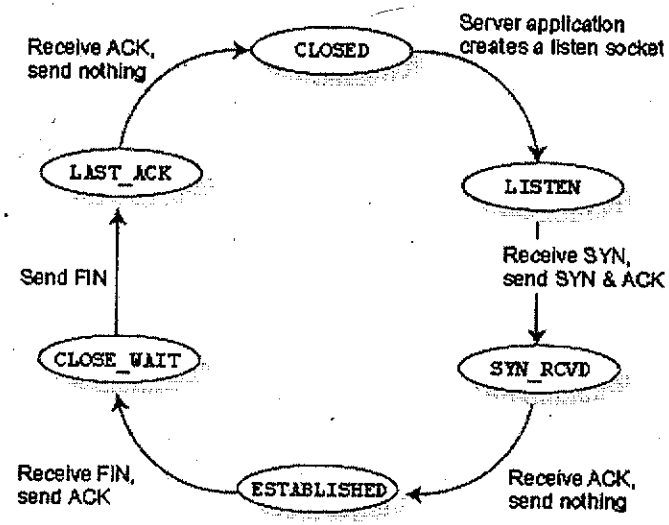
Hình 3.39 Dòng trạng thái của TCP

Giả sử ứng dụng client quyết định đóng kết nối (server tương tự). Khi đó TCP client gửi FIN segment và chuyển sang trạng thái FIN_WAIT_1. Trong trạng thái này, TCP client đợi segment biên nhận từ phía server. Sau khi nhận được segment này, TCP client chuyển sang trạng thái FIN_WAIT_2. Trong trạng thái FIN_WAIT_2, TCP client đợi FIN segment từ server. Sau khi nhận segment này, TCP client gửi segment ACK biên nhận tới server và chuyển sang trạng thái TIME_WAIT. Trong trạng thái TIME_WAIT, TCP client có thể gửi lại biên nhận ACK trong trường hợp ACK trước bị mất. Thời gian đợi ở trạng thái TIME_WAIT phụ thuộc

vào phần mềm triển khai TCP, nhưng thường nhận các giá trị 30 giây, một phút, hai phút. Sau khi hết thời gian đợi, kết nối chính thức được đóng và tất cả tài nguyên phía client (bao gồm cả số hiệu cổng) được giải phóng.

Hình 3.40 minh hoạ quá trình thay đổi trạng thái xảy ra ở TCP server. Giả sử client yêu cầu kết thúc trước.

Trong hai sơ đồ biến đổi trạng thái vừa nghiên cứu, chúng ta đã tìm hiểu về cơ chế thiết lập và giải phóng đường truyền của TCP.



Hình 3.40 Các trạng thái của kết nối TCP

3.6 KIỂM SOÁT TẮC NGHẼN TRONG TCP

Trong các phần trước, chúng ta đã nghiên cứu cách thức TCP cung cấp dịch vụ truyền tin tin cậy giữa hai tiến trình chạy trên những thiết bị đầu cuối khác nhau. Một dịch vụ cực kỳ quan trọng khác của TCP là cơ chế kiểm soát tắc nghẽn. Cơ chế này của TCP chỉ dựa 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. Trước khi đi sâu vào chi tiết, chúng ta hãy xem xét chung về cơ chế kiểm soát tắc nghẽn của TCP và mục tiêu kiểm soát tắc nghẽn của TCP khi nhiều kết nối TCP cùng chia sẻ một đường truyền bị tắc nghẽn.

Kết nối TCP kiểm soát tốc độ truyền của mình bằng cách giới hạn số lượng các segment đã gửi nhưng chưa được biên nhận. Định nghĩa w là số lượng cho phép các segment chưa cần biên nhận, thường được coi như **kích thước cửa sổ của TCP (TCP window)**. Lý tưởng là kết nối TCP cho phép truyền với tốc độ tối đa có thể (càng nhiều segment chưa biên nhận càng tốt) chừng nào chưa xảy ra hiện tượng mất segment do bị tắc nghẽn. Nói chung kết nối TCP bắt đầu với giá trị w tương đối nhỏ và sau đó "thăm dò" kênh truyền còn rỗi không bằng cách tăng dần giá trị w. Kết nối TCP tiếp tục được tăng w cho đến khi xảy ra mất dữ liệu (sự kiện hết thời gian đợi - timeout hay nhận được các biên nhận trùng lặp). Khi đó TCP sẽ giảm w tới một giá trị "an toàn" và sau đó lại bắt đầu "thăm dò" kênh truyền rỗi bằng cách tăng dần giá trị w.

Tổng quan về kiểm soát tắc nghẽn của TCP

Trong mục 3.5 chúng ta đã thấy mỗi kết nối TCP có bộ đệm gửi, bộ đệm nhận và một vài biến (LastByteRead, RcvWin...). Cơ chế kiểm soát tắc nghẽn của TCP bổ sung thêm hai biến nữa: **congestion window** (cửa sổ tắc nghẽn) và **threshold** (ngưỡng). 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. Như vậy khối lượng dữ liệu được gửi không được vượt quá CongWin và RcvWin, tức là:

Last ByteSent - LastByteAcked $\leq$ min {CongWin,RcvWin}.

Ngưỡng ký hiệu là **threshold** sẽ ảnh hưởng tới quá trình tăng của CongWin như trình bày dưới đây.

Chúng ta hãy xem giá trị CongWin tăng trong suốt kết nối TCP như thế nào. Để tập trung vào cơ chế kiểm soát tắc nghẽn (khác với kiểm soát lưu lượng), chúng ta giả thiết rằng bộ đệm nhận đủ lớn để có thể bỏ qua các hạn chế của cửa sổ nhận. Trong trường hợp này, số lượng dữ liệu gửi chưa cần được biên nhận chỉ bị giới hạn bởi CongWin. Chúng ta cũng giả thiết rằng phía gửi cần gửi nhiều dữ liệu.

Sau khi thiết lập kết nối TCP giữa hai hệ thống đầu cuối, tiến trình ứng dụng bên gửi chuyển dữ liệu tới bộ đệm gửi của TCP. TCP chia dữ liệu thành các khối với kích thước MMS, đặt các khối dữ liệu trong TCP

segment, và chuyển segment xuống tầng mạng để gửi đi. Cửa sổ tắc nghẽn của TCP điều tiết số lượng segment được gửi. Ban đầu, CongWin nhận giá trị 1 MMS, TCP gửi segment đầu tiên và được biên nhận. Nếu segment này được biên nhận trước khi timeout, phía gửi tăng CongWin lên 1 MMS và gửi đi hai segment. Nếu những segment này được biên nhận trong thời gian đợi của chúng, CongWin lại được tăng thêm 1 MMS cho mỗi segment được biên nhận. Khi đó CongWin mới là 4 MMS và phía gửi gửi đi bốn segment. Thủ tục này được thực hiện liên tục cho tới khi (1) CongWin vượt ngưỡng (threshold) hay (2) không nhận được biên nhận trong thời gian chờ biên nhận.

Trong giai đoạn này, cửa sổ tắc nghẽn (CongWin) tăng theo hàm số mũ. Ban đầu nó nhận giá trị 1 MSS, sau đó tăng lên 2 MMS, 4 MMS, 8 MMS.... Đây là giai đoạn **khởi đầu chậm (slow start)** vì giá trị cửa sổ khởi đầu với giá trị nhỏ (1 MMS). Tuy vậy giá trị cửa sổ tăng khá nhanh.

Giai đoạn slow-start kết thúc khi CongWin vượt ngưỡng. Khi đó giá trị CongWin sẽ tăng tuyến tính chứ không còn tăng theo hàm số mũ. Tức là nếu CongWin = w, sau khi nhận được biên nhận cho w segment, giá trị CongWin sẽ tăng lên 1, CongWin = w+1. Đây là giai đoạn **tránh tắc nghẽn (congestion avoidance)**.

Giai đoạn tránh tắc nghẽn tiếp tục khi vẫn nhận được biên nhận trong thời gian đợi. Tuy nhiên giá trị cửa sổ cũng như tốc độ gửi dữ liệu không thể tăng mãi. Đến lúc nào đó sẽ xảy ra sự cố mất gói dữ liệu ở router. Điều này dẫn đến sự kiện timeout ở phía gửi. Lúc này giá trị ngưỡng (threshold) nhận giá trị bằng một nửa CongWin, CongWin được đặt bằng 1 MMS. Bên gửi sẽ tiếp tục tăng nhanh giá trị CongWin theo hàm số mũ cho đến khi nó vượt ngưỡng.

Tóm lại:

Khi cửa sổ tắc nghẽn chưa vượt ngưỡng, cửa sổ sẽ tăng theo hàm mũ.

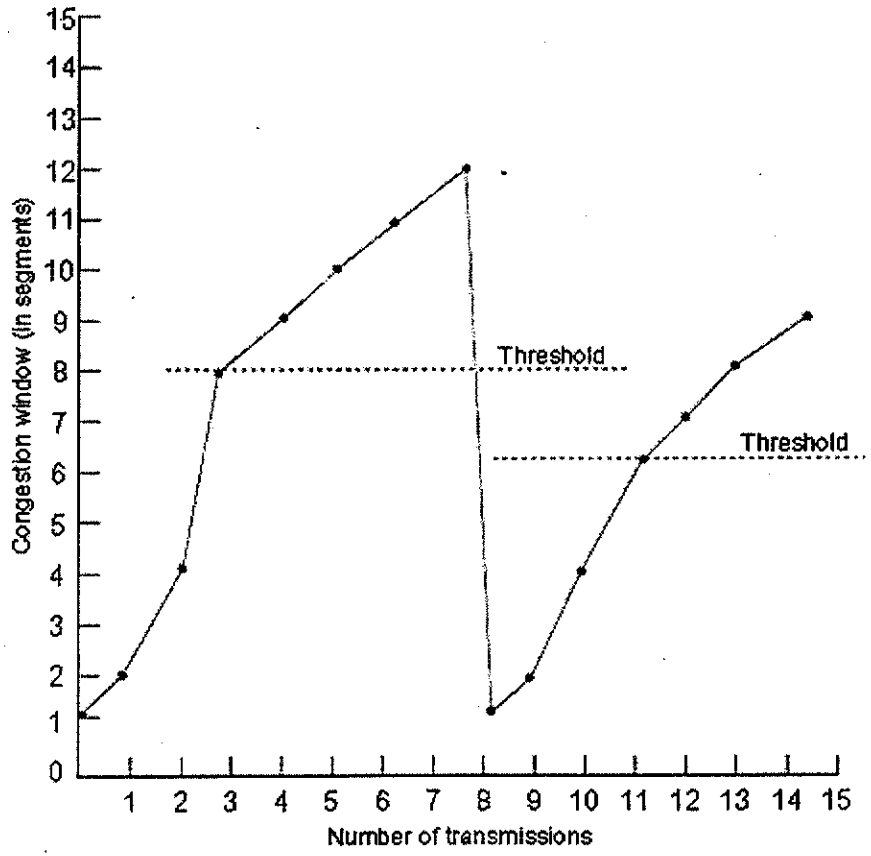
Khi cửa sổ tắc nghẽn vượt ngưỡng, cửa sổ sẽ tăng tuyến tính.

Khi hết thời gian đợi, giá trị ngưỡng bằng một nửa giá trị cửa sổ tắc nghẽn hiện thời và cửa sổ tắc nghẽn nhận giá trị 1.

Nếu bỏ qua giai đoạn slow-start, chúng ta sẽ thấy TCP tăng độ lớn cửa sổ theo cấp số cộng khi mạng chưa bị tắc nghẽn và giảm độ lớn cửa sổ

theo cấp số nhân (chia 2) ngay khi mạng bị tắc nghẽn. Vì vậy, TCP được coi là thuật toán **AIMD (additive-increase, multiplicative-decrease)**.

Giá trị độ lớn cửa sổ tắc nghẽn của TCP được minh hoạ trong Hình 3.41. Trong hình này, ngưỡng ban đầu bằng 8 MSS. Cửa sổ tắc nghẽn tăng nhanh theo luỹ thừa 2 trong giai đoạn slow-start và đạt ngưỡng tại t=3. Giá trị cửa sổ tắc nghẽn tăng tuyến tính đến khi xuất hiện mất mát dữ liệu. Giả sử khi dữ liệu bị mất, cửa sổ tắc nghẽn có giá trị 12 MMS. Ngưỡng mới được đặt bằng 0.5 CongWin = 6 MSS và cửa sổ tắc nghẽn bằng 1 MMS. Quá trình lại được tiếp tục. Thuật toán kiểm soát tắc nghẽn này do V.Jacobson đề xuất. Hiện nay có nhiều biến thể của thuật toán Jacobson (xem RFC 2581).



Hình 3.41 Cửa sổ kiểm soát tắc nghẽn

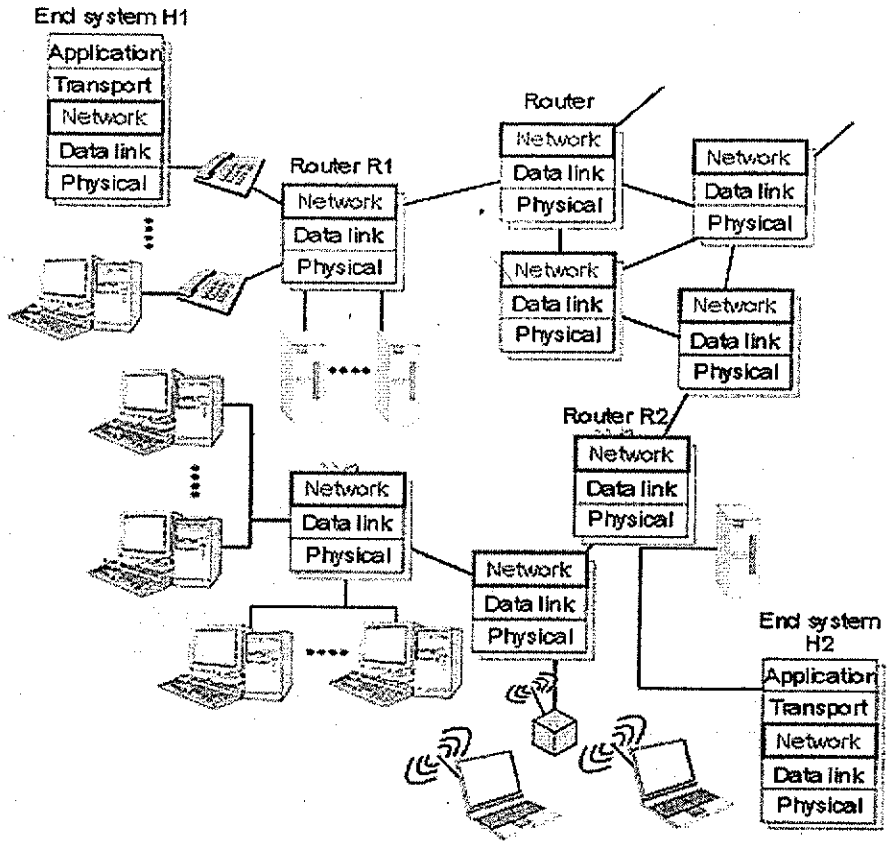
Tahoe, Reno và Vegas

Thuật toán kiểm soát tắc nghẽn của TCP ở đây được gọi là **Tahoe**. Một vấn đề với thuật toán Tahoe là khi một segment bị mất, người gửi có thể phải đợi trong một khoảng thời gian dài để gửi lại. Vì vậy một biến thể của Tahoe gọi là **Reno** đã được triển khai trong phần lớn các hệ điều hành. Giống Tahoe, Reno đặt độ lớn cửa sổ tắc nghẽn bằng 1 khi timeout (hết thời gian của bộ định thời). Tuy nhiên Reno có cơ chế truyền lại nhanh mà chúng ta đã khảo sát trong mục 3.5. Phía gửi sẽ gửi lại gói tin đã nhận được biên nhận trùng lặp ba lần liên tiếp ngay cả khi chưa hết thời gian đợi của gói tin này. Reno cũng sử dụng cơ chế khôi phục nhanh (fast recovery). Hiện nay phần lớn thực thể TCP sử dụng thuật toán Reno. Tuy nhiên có nhiều thuật toán cải tiến đáng kể hiệu suất của Reno – như **Vegas**.

Chương 4
TẦNG MẠNG

4.1 CÁC MÔ HÌNH DỊCH VỤ CỦA TẦNG MẠNG

Như đã trình bầy trong chương trước, tầng giao vận cung cấp dịch vụ truyền thông giữa hai tiến trình đang chạy trên hai máy tính khác nhau. Để cung cấp được dịch vụ này, tầng giao vận phải sử dụng dịch vụ cung cấp đường truyền giữa hai máy tính của tầng mạng. Nói cụ thể hơn, tầng mạng chuyển gói tin (segment) của tầng giao vận từ máy tính này đến máy tính khác. Tại máy tính gửi, tất cả các segment của tầng giao vận được chuyển xuống tầng mạng. Nhiệm vụ của tầng mạng là chuyển những segment này đến máy tính đích và gửi tới thực thể nào đó ở tầng giao vận bên trên. Công việc chuyển segment từ tầng giao vận máy tính nguồn đến tầng giao vận máy tính đích của tầng mạng chính là nội dung của chương này. Chúng ta sẽ thấy rằng không giống tầng giao vận, tầng mạng gồm nhiều máy tính và các router trung gian. Vì thế, giao thức tầng mạng là một trong những giao thức phức tạp nhất.



Hình 4.1 Một mô hình mạng đơn giản

Hình 4.1 minh hoạ một mạng đơn giản với hai máy tính (H1 và H2) và một số router trên đường truyền giữa H1 và H2. Tầng mạng ở máy tính gửi thực hiện bước gửi đầu tiên trên toàn bộ hành trình của gói tin. Ví dụ nếu H1 gửi gói tin đến H2 thì tầng mạng trên H1 sẽ truyền gói tin này đến router gần nhất: R2. Tại máy tính nhận (chẳng hạn H2), tầng mạng sẽ nhận gói tin từ router gần nó nhất (trong trường hợp này là R2) và chuyển lên cho tầng giao vận tại H2. Vai trò chính của router là chuyển gói tin từ một đầu vào nào đó (input) tới một đầu ra nào đó (output). Chú ý rằng các tầng trên của tầng mạng (giao vận, ứng dụng) không hoạt động tại các router trong Hình 4.1 bởi vì router không cần thiết phải chạy các giao thức ở tầng giao vận hay tầng ứng dụng (ngoại trừ các mục đích kiểm soát).

Vai trò của tầng mạng đơn giản chỉ là chuyển gói tin từ máy tính gửi đến máy tính nhận. Vì thế, tầng mạng có ba chức năng quan trọng sau đây:

Xác định đường đi (Path determination): Tầng mạng phải xác định các router trung gian hay tuyến đường (path) mà gói tin được truyền từ nơi gửi đến nơi nhận. Thuật toán xác định tuyến đường như vậy gọi là "**thuật toán định tuyến**" (routing algorithm). Thuật toán định tuyến sẽ quyết định đường đi của các gói tin từ máy tính nhận đến máy tính gửi (trong ví dụ là máy tính H1 và máy tính H2). Trọng tâm của chương này là các thuật toán định tuyến. Trong phần 4.2, chúng ta sẽ nghiên cứu lý thuyết của thuật toán định tuyến, tập trung vào hai kiểu thuật toán: *Link state* và *Distance vector*. Chúng ta sẽ thấy độ phức tạp của thuật toán định tuyến phụ thuộc vào số lượng router trên đường truyền. Điều này dẫn đến định tuyến phân cấp, một chủ đề được trình bày trong phần 4.3.

Chuyển mạch (Switching): Khi gói tin đến đầu vào, router phải quyết định gửi gói tin đến đầu ra thích hợp nào. Ví dụ, gói tin từ máy H1 đến router R1 sẽ phải được chuyển đến router kế tiếp trên đường tới H2. Trong phần 4.6 chúng ta sẽ nghiên cứu hoạt động của router và quá trình chuyển một gói tin từ đầu vào đến đầu ra trong một router diễn ra như thế nào.

Thiết lập đường truyền (Call setup): Trong phần trước trình bày về TCP, chúng ta thấy rằng hai thực thể truyền thông phải có một giai đoạn "bắt tay" trước khi trao đổi dữ liệu thực sự. Điều này cho phép bên gửi và bên nhận thiết lập các thông tin trạng thái cần thiết (ví dụ, số thứ tự khởi đầu, độ lớn cửa sổ). Với một số kiến trúc mạng khác (ví dụ ATM) đòi hỏi các router trên tuyến đường từ nguồn đến đích phải "bắt tay" nhau trước khi bắt đầu truyền dữ liệu thực sự. Trong tầng mạng, quá trình này được gọi là thiết lập đường truyền (*call setup*). Chúng ta sẽ thấy tầng mạng trong kiến trúc Internet không đòi hỏi công việc này.

Tuy nhiên, trước khi đi sâu vào chi tiết các khái niệm và triển khai của tầng mạng, chúng ta sẽ xem xét tổng quát những kiểu dịch vụ khác nhau của tầng mạng.

4.1.1 Mô hình dịch vụ mạng

Khi tầng giao vận ở thiết bị gửi chuyển các gói tin xuống tầng mạng, liệu tầng giao vận có thể tin cậy tầng mạng chuyển gói tin này đến đích không? Liệu khi gửi nhiều gói tin, chúng có được chuyển đến tầng giao vận của thiết bị nhận theo đúng thứ tự không ? Vận tốc gửi các gói tin có được xác định trước không? Tầng mạng có phản hồi lại các thông tin liên quan đến tắc nghẽn không? Mô hình dịch vụ mạng sẽ trả lời những câu hỏi này và nhiều vấn đề khác.

Chuyển mạch gói (datagram) và chuyển mạch ảo (virtual circuit)

Có lẽ điểm trừu tượng quan trọng nhất mà tầng mạng che dấu các tầng trên là việc có sử dụng **mạch ảo (Virtual Circuit - VC)** hay không. Về khía cạnh nào đó mạch ảo tương tự mạng điện thoại truyền thống (mặc dù mạng điện thoại sử dụng mạch thực). Có ba giai đoạn trong chuyển mạch ảo:

Thiết lập mạch ảo: trong cả giai đoạn thiết lập, phía gửi thông báo địa chỉ nhận với tầng mạng, yêu cầu tầng mạng thiết lập VC. Tầng mạng xác định tuyến đường giữa bên gửi và bên nhận, tức là chuỗi các cung đường (hay đường kết nối - link) cũng như các thiết bị chuyển mạch (switch – nút trung gian) mà tất cả các gói dữ liệu sẽ đi qua. Giai đoạn này yêu cầu việc cập nhật bảng định tuyến và dự trữ tài nguyên trong mỗi thiết bị chuyển mạch.

Truyền dữ liệu: Sau khi thiết lập, dữ liệu có thể được chuyển trong VC.

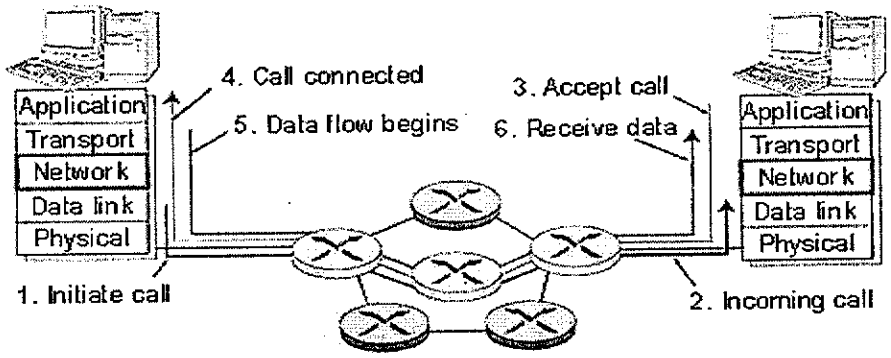
Giải phóng mạch ảo: Giai đoạn này bắt đầu khi phía gửi (hoặc phía nhận) báo cho tầng mạng yêu cầu đóng VC. Tầng mạng sẽ thông báo cho thiết bị đầu cuối bên kia cũng như các thiết bị chuyển mạch trên VC để cập nhật lại các bảng định tuyến, giải phóng tài nguyên.

Có sự khác biệt tuy nhỏ - nhưng quan trọng giữa thiết lập VC ở tầng mạng và thiết lập kết nối ở tầng giao vận (giai đoạn bắt tay 3 bước của TCP). Thiết lập kết nối ở tầng giao vận chỉ liên quan đến các thiết bị đầu cuối ở trên hai đầu mút. Hai thiết bị đồng ý thiết lập kết nối và thoả thuận các thông số của kết nối (ví dụ số thứ tự khởi tạo, độ lớn cửa sổ kiểm soát lưu lượng). Hai thiết bị đầu cuối này sẽ nhận biết được về sự kết nối ở tầng giao vận, nhưng các thiết bị chuyển mạch ở giữa thì không. Trái lại trong tầng mạng của mạng chuyển mạch ảo, tất cả các thiết bị chuyển mạch giữa hai thiết bị đầu cuối đều tham gia vào quá trình thiết lập mạch ảo, và do đó đều nhận biết được tất cả các VC đi qua.

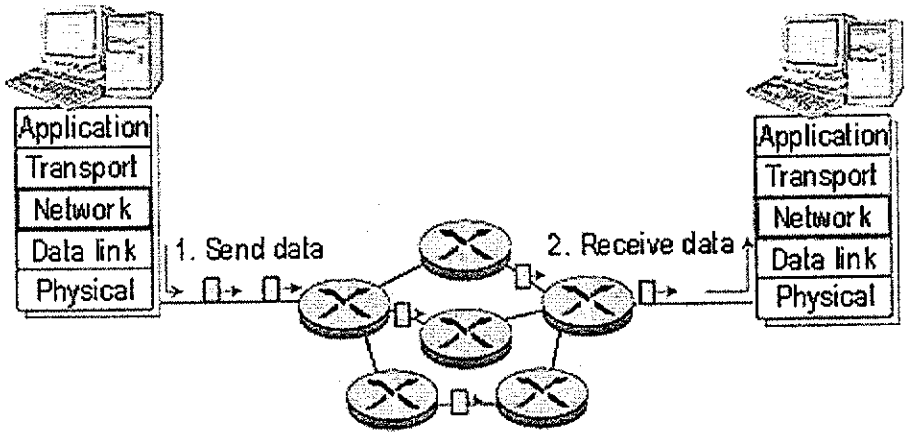
Thông điệp trao đổi giữa các thiết bị đầu cuối yêu cầu khởi tạo hay kết thúc mạch ảo hay thông điệp trao đổi giữa các thiết bị chuyển mạch yêu cầu thiết lập VC (để cập nhật bảng chuyển mạch) được gọi là **thông điệp báo hiệu (signaling message)**. Giao thức được sử dụng để trao đổi những thông điệp này là **giao thức báo hiệu (signaling protocol)**. Quá trình thiết lập VC được minh hoạ trong Hình 4.2. ATM, Frame Relay và X.25 là ba kiến trúc mạng sử dụng chuyển mạch ảo.

Trong mạng chuyển mạch gói, khi muốn gửi gói tin, thiết bị đầu cuối đặt vào gói tin địa chỉ thiết bị nhận và sau đó chuyển gói tin

vào mạng. Như minh hoạ trong Hình 4.3, không có giai đoạn thiết lập VC nào. Những thiết bị trung chuyển trong mạng chuyển mạch gói (được gọi là bộ định tuyến - router trên Internet) không duy trì bất kỳ trạng thái nào về VC vì không có VC. Thiết bị trung chuyển sẽ định tuyến gói tin đến đích bằng cách xác định địa chỉ đích, tìm kiếm trên bảng định tuyến và chuyển tiếp gói tin theo hướng đến đích (giống việc chuyển thư bình thường trong hệ thống bưu điện). Vì bảng định tuyến có thể được cập nhật liên tục, nên các gói tin được gửi từ thiết bị đầu cuối này đến thiết bị đầu cuối khác có thể đi theo nhiều tuyến đường khác nhau và đến đích không theo thứ tự. Mạng Internet công cộng ngày nay sử dụng dịch vụ chuyển mạch gói.



Hình 4.2 Mô hình dịch vụ chuyển mạch ảo



Hình 4.3 Mô hình chuyển mạch gói

Để chuyển gói tin của tầng giao vận, tầng mạng thường đưa ra dịch vụ chuyển mạch ảo hoặc dịch vụ chuyển mạch gói nhưng không bao giờ cung cấp cả hai dịch vụ này. Ví dụ, dịch vụ của mạng ATM là VC trong khi mạng Internet cung cấp dịch vụ chuyển mạch gói.

Một thuật ngữ tương đương của mạch ảo (VC) và mạch gói (Datagram) tương ứng là dịch vụ hướng nối (connection-oriented) và dịch vụ không hướng nối (connectionless). Dịch vụ chuyển mạch ảo được xếp vào lớp dịch vụ hướng nối vì phải thiết lập và kết thúc kết nối cũng như việc duy trì thông tin trạng thái của kết nối tại tất cả thiết bị chuyển mạch. Dịch vụ chuyển mạch gói được xếp vào lớp dịch vụ không hướng nối. Cả hai nhóm thuật ngữ đều có ưu điểm cũng như nhược điểm và đều được sử dụng phổ biến trong các tài liệu về mạng. Trong quyển sách này, chúng ta sử dụng thuật ngữ "dịch vụ chuyển mạch ảo" và "dịch vụ chuyển mạch gói" cho tầng mạng và sử dụng thuật ngữ "dịch vụ hướng nối" và "dịch vụ không hướng nối" cho tầng giao vận. Điểm khác biệt này sẽ giúp người đọc hiểu được những dịch vụ khác nhau của 2 tầng này.

Bảng 4.1 tổng kết những nét chính của mô hình dịch vụ Internet và kiến trúc mạng ATM. Chúng ta không đi sâu vào những khía cạnh chi tiết của mô hình dịch vụ ở đây. Kiến trúc hiện nay của Internet chỉ cung cấp duy nhất dịch vụ chuyển mạch gói, một dịch vụ theo kiểu **cố gắng tối đa (best-effort)**. Như minh hoạ trong bảng 4.1, dịch vụ này cũng giống với việc không cung cấp bất kỳ dịch vụ nào cả.

Kiến trúc mạng	Mô hình dịch vụ	Đảm bảo băng thông	Đảm bảo không mất	Thứ tự	Độ trễ	Kiểm soát tắc nghẽn
Internet	Cố gắng tối đa	Không	Không	Không	Không	Không
ATM	CBR	Có	Có	Có	Có	Không có tắc nghẽn
ATM	VBR	Có	Có	Có	Có	Không có tắc nghẽn
ATM	ABR	Bảo đảm tốc độ nhỏ nhất	Không	Có	Có	Không có tắc nghẽn
ATM	UBR	Không	Không	Có	Có	Không

Bảng 4. 1

Mạng không đảm bảo thời gian gửi các gói tin giống nhau, các gói tin không được đảm bảo đến đích theo đúng thứ tự, và thậm chí không đảm bảo gói tin đến được đích. Với định nghĩa này, một mạng không chuyển bất kỳ gói tin nào đến đích cũng được phân loại theo kiểu cố gắng tối đa (ví dụ mạng Internet công cộng hay bị tắc nghẽn).

Chúng ta hãy chuyển sang mô hình dịch vụ ATM. Ở đây, chúng ta sẽ tập trung vào mô hình dịch vụ đã được Diễn đàn ATM [ATM Forum] chuẩn hoá. Kiến trúc ATM cung cấp nhiều kiểu dịch vụ khác nhau (tức là chuẩn ATM có nhiều mô hình dịch vụ). Trong phạm vi cùng một mạng, những kết nối khác nhau có thể được cung cấp những lớp dịch vụ khác nhau.

Dịch vụ truyền với tốc độ cố định - Constant bit rate (CBR): là mô hình dịch vụ ATM đầu tiên được chuẩn hoá, ở đây có thể thấy được vai trò các công ty điện thoại đằng sau ATM. Dịch vụ mạng CBR là sự lựa chọn lý tưởng cho việc truyền dữ liệu đa phương tiện (ví dụ điện thoại số) theo thời gian thực với tốc độ truyền cố định. Mục tiêu của dịch vụ CBR là làm cho kết nối mạng trông giống như một đường kết nối thực sự (bằng dây đồng hay cáp quang) giữa bên gửi và bên nhận. Trong dịch vụ CBR, các gói tin ATM (trong thuật ngữ ATM là các **tế bào ATM – ATM cell**) được truyền qua mạng với một độ trễ nào đó (được gọi là cell transfer delay, CTD). Biến thiên của độ trễ ("jitter" hay cell – delay variation, CDV), tỷ lệ các cell bị mất hay đến trễ (cell – lost rate, CLR) được đảm bảo không vượt quá một giá trị ngưỡng. Tốc độ truyền tối đa của mỗi kết nối được xác định trước (pick cell rate, PCR) và bên gửi có thể gửi dữ liệu với tốc độ này. Các giá trị PCR, CTD, CDV và CLR đã được máy tính gửi và mạng ATM thoả thuận trước trong giai đoạn thiết lập kết nối CBR.

Lớp dịch vụ ATM thứ hai là **dịch vụ truyền với tốc độ không xác định (Unspecified bit rate - UBR)**. Không giống dịch vụ CBR (đảm bảo tốc độ, độ trễ, mất mát dữ liệu), UBR không đảm bảo những điều này ngoại trừ việc gửi các cell theo đúng thứ tự. Như vậy dịch vụ UBR giống mô hình dịch vụ cố gắng tối đa của Internet. Dịch vụ UBR không cung cấp thông tin phản hồi cho bên gửi về việc các cell có đến được đích hay không. Với mạng UBR, tính tin cậy của truyền dữ liệu được triển khai trong các giao thức ở tầng cao hơn. Dịch vụ UBR phù hợp với những ứng dụng truyền dữ liệu không cần tốc độ truyền cố định như email, newsgroup.

Nếu UBR được xem như một dịch vụ theo kiểu cố gắng tối đa thì **dịch vụ truyền với tốc độ có sẵn (available rate bit - ABR)** có thể phân loại vào nhóm dịch vụ theo kiểu cố gắng tối đa nhưng ưu việt hơn. Hai tính năng bổ sung quan trọng nhất của dịch vụ ABR là:

Tốc độ truyền cell nhỏ nhất (MRC) được đảm bảo cho kết nối ABR. Tuy nhiên, khi tài nguyên của mạng rỗi, bên gửi có thể gửi với tốc độ cao hơn MCR.

Có phản hồi về tắc nghẽn từ tầng mạng. Mạng ATM có thể cung cấp thông tin phản hồi cho bên gửi (là bit thông báo tắc nghẽn hay tốc độ gửi thấp) để bên gửi điều chỉnh tốc độ gửi.

ABR không đảm bảo một băng thông tối thiểu, nhưng cố gắng truyền dữ liệu nhanh nhất có thể. Như vậy, ABR phù hợp với các ứng dụng truyền dữ liệu yêu cầu độ trễ nhỏ (ví dụ duyệt Web).

Mô hình ATM cuối cùng là **dịch vụ truyền với tốc độ biến đổi (variable bit rate - VBR)**. Trong dịch vụ VBR thời gian thực, tỷ lệ mất gói dữ liệu, độ trễ có thể chấp nhận được thỏa thuận trước giống dịch vụ CBR. Tuy nhiên, tốc độ gửi thực sự được phép thay đổi theo các tham số do người dùng đưa vào. Điều này cho phép sử dụng tài nguyên có hiệu quả hơn, nhưng xét theo các tiêu chí về mất mát dữ liệu, độ trễ thì VBR tương tự CBR.

Có thể tìm hiểu kỹ hơn về ATM trong ATM Forum's Traffic Management Specification 4.0 [ATM Forum 1996].

3.1.2 Nguồn gốc của dịch vụ chuyển mạch gói và chuyển mạch ảo

Lịch sử phát triển mô hình dịch vụ mạng Internet và ATM phản ánh nguồn gốc của chúng. Với khái niệm trọng tâm là mạch ảo và dịch vụ CBR là dịch vụ đầu tiên, ATM rõ ràng xuất phát từ mạng điện thoại truyền thống (sử dụng "mạch thực"). Các định nghĩa sau này về lớp dịch vụ UBR và ABR đã ghi nhận tầm quan trọng của việc phát triển các ứng dụng truyền dữ liệu. Với kiến trúc VC và trọng tâm hỗ trợ truyền dữ liệu theo thời gian thực cùng

với sự bảo đảm về hiệu năng hệ thống (thậm chí kể cả với dịch vụ ABR), tầng mạng phức tạp hơn rất nhiều so với mô hình Internet theo kiểu cố gắng tối đa. Mạng điện thoại đặt tính phức tạp vào bên trong mạng vì mạng này kết nối các thiết bị đầu cuối câm (dump device), ví dụ các máy điện thoại quay số.

Ngược lại mạng toàn cầu Internet phát sinh từ nhu cầu kết nối các máy tính (được xem là thiết bị đầu cuối thông minh) với nhau. Với thiết bị đầu cuối phức tạp, kiến trúc Internet lựa chọn mô hình dịch vụ mạng đơn giản nhất có thể và đặt các chức năng phụ trợ (ví dụ như truyền dữ liệu tin cậy), cũng như các ứng dụng mạng ở tầng cao hơn triển khai trên các thiết bị đầu cuối. Điều này ngược với mô hình mạng điện thoại, và kết quả là: Mô hình dịch vụ mạng của Internet không đảm bảo bất kỳ một dịch vụ nào do vậy có thể dễ dàng kết nối các mạng sử dụng những công nghệ kết nối rất khác nhau (ví dụ vệ tinh, Ethernet, cáp quang, sóng vô tuyến). Các công nghệ này có tốc độ và tỷ lệ mất mát dữ liệu khác nhau. Kết nối các mạng IP được thảo luận chi tiết trong mục 4.4.

Như chúng ta đã thấy trong chương 2, những ứng dụng như thư điện tử, Web, và thậm chí cả dịch vụ của tầng mạng như DNS được triển khai trên các máy tính (là thiết bị đầu cuối). Các dịch vụ mới có thể nhanh chóng được sử dụng rộng rãi thông qua các giao thức tầng ứng dụng.

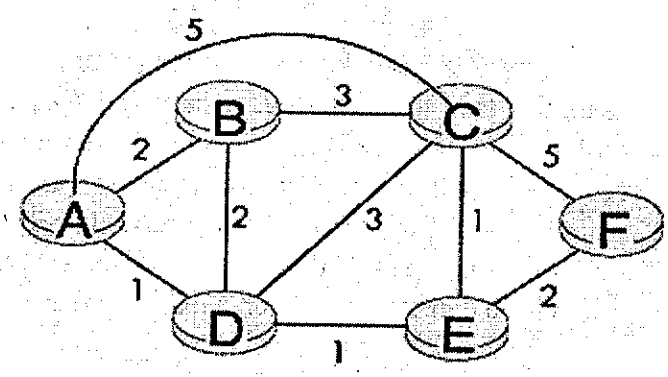
Tuy nhiên có sự tranh cãi lớn trong cộng đồng Internet về việc cải tiến kiến trúc của tầng mạng để hỗ trợ các ứng dụng đa phương tiện thời gian thực.

4.2 CÁC NGUYÊN LÝ ĐỊNH TUYẾN

Để truyền gói dữ liệu từ máy tính gửi đến máy tính nhận, tầng mạng phải quyết định đường đi hoặc các router mà gói dữ liệu phải đi qua. Dù mạng chuyển mạch gói (các gói tin khác nhau có thể đi theo các tuyến đường khác nhau) hay mạng mạch ảo (tất cả các gói tin được truyền trên cùng một tuyến đường định trước) thì tầng mạng đều phải xác định đường đi cho gói tin. Đây là công việc của các giao thức định tuyến ở tầng mạng.

Trái tim của giao thức định tuyến là thuật toán xác định đường đi cho gói tin – thuật toán định tuyến. Mục tiêu của thuật toán định tuyến hết sức đơn giản: với một tập hợp router cùng với liên kết giữa các router, thuật toán định tuyến phải xác định đường đi tốt nhất từ thiết bị nguồn đến thiết bị đích. Đường đi tốt đơn giản có thể là đường đi có giá nhỏ nhất. Tuy nhiên trong thực tế chúng ta sẽ thấy các vấn đề liên quan đến chính sách (policy) – ví dụ router X thuộc tổ chức Y không được chuyển tiếp các gói tin được tạo ra từ mạng của tổ chức Z-có thể làm phức tạp các thuật toán của router lên nhiều.

Người ta thường sử dụng đồ thị để xây dựng các thuật toán định tuyến như minh hoạ trên Hình 4.3. Trong chương này chúng ta sử dụng kỹ thuật biểu diễn sơ đồ mạng dưới dạng đồ thị. Ở đây, nút (node) của đồ thị biểu diễn router - điểm quyết định việc định tuyến gói tin - và những đoạn thẳng ("cung" trong lý thuyết đồ thị) nối các nút biểu diễn đường truyền vật lý thực sự giữa các router. Cung được đặc trưng bởi đại lượng "giá" (cost) là chi phí của việc gửi gói tin qua đường truyền tương ứng. Giá có thể phản ánh mức tắc nghẽn trên đường truyền (thời gian trễ trung bình) hoặc khoảng cách vật lý thực sự giữa hai router (ví dụ, đường truyền xuyên đại dương sẽ có giá cao hơn đường truyền giữa các khu vực trên đất liền). Để đơn giản, chúng ta coi mỗi cung trên đồ thị có một giá và không quan tâm đến việc xác định giá bằng cách nào.



Hình 4.4 Ví dụ một mô hình mạng

Với mô hình đồ thị, vấn đề tìm kiếm tuyến đường từ nguồn đến đích có chi phí thấp nhất yêu cầu xác định chuỗi các cung sao cho:

- Cung đầu tiên trong chuỗi xuất phát từ nguồn.
- Đích của cung cuối cùng trong chuỗi là đích.
- Với mọi i, cung thứ i và i-1 cùng kết nối vào một nút.

Với đường đi có giá nhỏ nhất, tổng chi phí tất cả các cung trên tuyến đường là nhỏ nhất. Chú ý nếu tất cả các cung có giá như nhau thì đường đi có giá nhỏ nhất cũng là đường đi ngắn nhất giữa nguồn và đích.

Ví dụ như trong Hình 4.4, đường đi có giá nhỏ nhất giữa nút A (nguồn) và nút C (đích) là đường ADEC.

Một bài tập đơn giản: hãy trình bày cách tìm đường đi có giá thấp nhất từ A đến F. Phần lớn mọi người sẽ tìm bằng cách kiểm tra Hình 4.3, lần theo các router từ A đến F qua nhiều con đường và tự thuyết phục rằng đường đi mà mình chọn có giá nhỏ nhất so với tất cả các đường khác. (Có tất cả 12 tuyến đường khác nhau nối A và F). Quá trình xác định như thế là ví dụ của thuật toán định tuyến tập trung - chạy trong bộ não của bạn với đầy đủ thông tin về mạng. Nói chung, chúng ta có thể phân loại thuật toán định tuyến vào hai kiểu: toàn cục hay phân tán.

Thuật toán định tuyến toàn cục (global) xác định đường đi với giá thấp nhất giữa nguồn và đích bằng cách sử dụng tất cả thông tin về tổng thể mạng. Đầu vào của thuật toán là tất cả các nút, cung và giá của các cung. Rõ ràng router phải bằng một cách nào đó thu được các thông tin này trước khi bước vào giai đoạn tính toán thực sự. Thuật toán có thể được chạy tại một nơi (thuật toán định tuyến tập trung) hoặc chạy tại nhiều nơi. Tuy nhiên điểm phân biệt chính yếu là thuật toán định tuyến toàn cục phải có trước đầy đủ thông tin về đồ thị mạng. Trong thực tế, thuật toán như vậy được gọi là thuật toán **link state** vì thuật toán phải biết được giá của mỗi liên kết trên mạng. Chúng ta sẽ nghiên cứu thuật toán global link state trong mục 4.2.1.

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. Ban đầu mỗi nút chỉ biết về giá của các cung có kết nối trực tiếp với nó. Sau đó, thông qua các bước tính toán và trao đổi thông tin với các nút hàng xóm (hai nút được gọi là hàng xóm nếu giữa chúng có một đường kết nối vật lý trực tiếp và trong

thuật ngữ đồ thị gọi là hai đỉnh kề nhau), nút sẽ dần dần xác định được đường đi có giá nhỏ nhất đến một tập hợp đích nào đó. Chúng ta sẽ nghiên cứu thuật toán định tuyến phân tán - thuật toán **distance vector** trong mục 4.2.2. Được gọi là thuật toán distance vector bởi vì nút không biết được đường đi cụ thể đến đích mà chỉ biết đến nút hàng xóm trên đường đến đích và tổng giá của đường đi đến đích.

Một kiểu phân loại thuật toán thứ hai là theo tính chất tĩnh hay động. Trong thuật toán định tuyến tĩnh, tuyến đường thay đổi rất ít theo thời gian, thường là kết quả do con người tác động (ví dụ đặt lại cấu hình cho bảng định tuyến trong router). Thuật toán định tuyến động cho phép thay đổi các tuyến đường khi lưu lượng mạng hay kiến trúc liên kết mạng bị thay đổi. Thuật toán động có thể được chạy định kỳ hoặc gửi thông điệp trực tiếp khi cấu trúc mạng hay giá các cung bị thay đổi. Thuật toán động có thể xử lý được khi mạng thay đổi nhưng lại nảy sinh các vấn đề mới như định tuyến lặp (mục 4.2.2).

Thuật toán định tuyến thường được sử dụng trong Internet gồm hai kiểu chính: thuật toán global link state tĩnh và thuật toán distance vector động. Chúng ta sẽ phân tích những thuật toán này lần lượt trong mục 4.2.1 và mục 4.2.2. Những thuật toán định tuyến khác sẽ được nói qua trong mục 4.2.3.

4.2.1. Thuật toán định tuyến Link state

Trong thuật toán Link state, cấu trúc mạng và giá của tất cả các liên kết đều phải được xác định trước. Đây là dữ kiện đầu vào của thuật toán link state. Trong thực tế, điều này được thực hiện bằng cách mỗi nút sẽ gửi thông báo quảng bá về định danh của mình và giá các cung liên kết trực tiếp đến nó tới tất cả các router khác trên mạng. Việc quảng bá rộng rãi trạng thái liên kết có thể được thực hiện ngay khi nút không biết về đầy đủ các nút khác trên mạng. Ban đầu nút chỉ biết được thông tin về các hàng xóm của mình cũng như giá các cung đến các hàng xóm. Nhưng sau đó nó sẽ xác định được topo của phần còn lại của mạng khi nhận những thông báo quảng bá từ các nút khác. (Trong chương 5, chúng ta thấy các router hàng xóm trao

đổi thông tin với nhau như thế nào). Kết quả của việc quảng bá trạng thái liên kết là tất cả các nút có thể đầy đủ thông tin về tổng thể mạng. Sau đó mỗi nút đều có thể chạy thuật toán Link state và xác định đường đi có giá thấp nhất tới mọi nút.

Thuật toán Link state được trình bày ở đây là thuật toán Dijkstra (đặt theo tên của người phát minh ra). Thuật toán Dijkstra xác định đường đi có giá thấp nhất từ một nút nguồn (không mất tổng quát, giả sử là A) đến tất cả các nút khác trên mạng. Thuật toán Dijkstra có nhiều bước và sau ***k*** bước sẽ xác định được đường đi có giá thấp nhấp tới ***k*** nút đích. Chúng ta định nghĩa một số ký hiệu sau:

c(i, j): giá liên kết từ nút i đến nút j. Nếu nút i và nút j không có đường kết nối trực tiếp thì $c(i, j) = \infty$. Để đơn giản, chúng ta coi c(i, j) = c(j, i).

D(v): giá hiện tại thấp nhất của tuyến đường đi từ nút nguồn đến nút v.

P(v): nút phía trước nút v (hàng xóm của v) trên tuyến đường hiện có giá thấp nhất từ nguồn tới nút v.

N: tập hợp của các nút đã xác định được đường đi ngắn nhất.

Thuật toán Link state gồm có bước khởi tạo cho vòng lặp. Số các bước bằng tổng số nút trên mạng. Khi kết thúc, thuật toán sẽ xác định được đường đi ngắn nhất từ nút nguồn đến tất cả các nút khác trên mạng.

Thuật toán Link state (LS):

```
Khởi tạo:
    N = {A}
    for (tất cả các nút v)
        if v kề A
            thì D(V) = c(A, v)
        else D(v) = ∞

Repeat
    Tìm w không ở trong N có D(w) nhỏ nhất
    Bổ sung w vào N
    Cập nhật D(v) cho tất cả v kề với w và không nằm trong N:
        D(v) = min ( (D(v), D(w) + c(w, v) )
     /* Giá mới đến v khác giá cũ đến v hoặc biết được giá đường đi ngắn nhất
     đến w cộng với giá từ w đến v */
Until tất cả các nút nằm trong N
```


Bước	N	D(b), p(B)	D(C), p(C)	D(D), p(D)	D(E), p(E)	D(F), p(F)
0	A	2, A	5.A	1, A	∞	∞
1	AD	2, A	4, D		2, D	∞
2	ADE	2, A	3, E			4, E
3	ADEB		3, E			4, E
4	ADEBC					4, E
5	ADEBCF					

Bảng 4.2 Bảng trạng thái các bước cho mạng minh hoạ trên hình 4.4

Xét đồ thị mạng trong Hình 4.4 và tính đường đi có giá thấp nhất từ A đến tất cả các nút khác. Bảng 4.2 cho thấy các kết quả tính của thuật toán, mỗi dòng trong bảng ứng với trạng thái của thuật toán sau khi kết thúc một bước. Sau đây chúng ta sẽ phân tích một số bước đầu tiên:

Trong bước khởi tạo, giá hiện tại thấp nhất của đường đi từ A đến các nút hàng xóm B, C và D tương ứng là 2, 5, và 1. Chúng ta có một chú ý nhỏ ở đây, giá đến C được đặt là 5 (ngay sau đây chúng ta sẽ thấy đây không phải là đường đi tốt nhất) vì đây là giá của đường nối trực tiếp từ A đến C. Giá đến E và F được đặt là vô cùng vì giữa A và E, F không có đường kết nối trực tiếp.

Trong bước đầu tiên chúng ta tìm kiếm trên những nút chưa được đưa vào tập N và xác định nút có giá đến thấp nhất. Đó là nút D với giá là 1 và do đó D được bổ sung vào N. Dòng 12 của thuật toán LS được thực hiện để cập nhật D(v) cho tất cả các nút v, kết quả nhận được được trình bày trong dòng thứ 2 (bước 1) trong bảng 4.2. Giá của đường đi đến B không đổi. Giá đường đi đến C (nhận giá trị 5 trong bước khởi tạo trước) qua D có giá trị nhỏ hơn là 4. Đây là đường tốt hơn được chọn và nút phía trước của C trên đường đi ngắn nhất từ A sẽ là D. Tương tự vậy, giá đường đi đến E (qua D) được tính là 2 và bảng được cập nhật tương ứng.

Trong bước thứ hai, đường đi đến nút B và E đều có giá thấp nhất và chúng ta bổ sung E vào tập N (Bây giờ N chứa A, D và E). Giá đến các nút chưa nằm trong N (gồm B, C và F) được cập nhật trong dòng 12 của thuật toán LS, kết quả là dòng 3 của bảng 4.2.

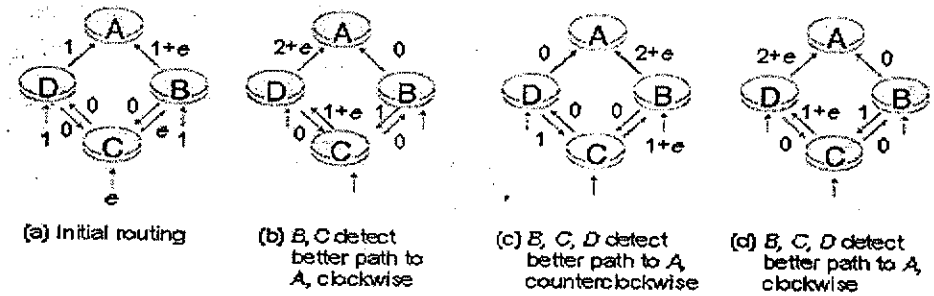
....

Khi thuật toán LS kết thúc, với mỗi nút chúng ta xác định được nút ngay trước nó trên tuyến đường có giá thấp nhất xuất phát từ nguồn. Với mỗi nút phía trước, chúng ta lại có nút phía trước nữa... Cuối cùng chúng ta xác định được toàn bộ đường đi từ nguồn đến tất cả các nút đích.

Độ phức tạp tính toán của thuật toán này bằng bao nhiêu? Với n nút (không kể nút nguồn), để tìm đường đi có giá thấp nhất từ nguồn đến tất cả các đích, khối lượng tính toán là bao nhiêu trong trường hợp xấu nhất? Trong vòng lặp đầu tiên, chúng ta cần kiểm tra qua tất cả n nút để xác định nút w có giá nhỏ nhất không nằm trong n; trong vòng lặp thứ hai, chúng ta cần kiểm tra n – 1 nút để xác định giá thấp nhất; trong vòng lặp thứ ba là n-2 nút ... Tổng số các nút mà chúng ta cần phải kiểm tra qua tất cả các bước là $n(n+1)/2$ và theo đó chúng ta có thể nói rằng thuật toán Link state có độ phức tạp là $O(n^2)$. (Thuật toán này có thể được cải tiến bằng cách sử dụng cấu trúc dữ liệu HEAP, độ phức tạp chỉ còn theo hàm logarit của n).

Trước khi kết thúc trình bày về thuật toán LS, chúng ta hãy xét ví dụ minh họa một cấu hình mạng giống như trên Hình 4.5. Giá của mỗi liên kết (cung) bằng tải hiện tại trên nó (như thế giá sẽ là độ trễ mà gói tin phải chịu). Trong trường hợp này giá không có tính chất đối xứng, nghĩa là c(A, B) chỉ bằng c(B, A) nếu tải trên cả hai hướng AB là như nhau. Giả sử hai nút B và D gửi một đơn vị dữ liệu, nút C gửi khối lượng dữ liệu là e tới A. Định tuyến ban đầu được minh hoạ trên Hình 4.5(a), giá của mỗi cung ứng với tải trên cung đó.

Trong bước tiếp theo của thuật toán LS, nút C (với giá liên kết cơ bản đã được xác định trong Hình 4.5a) nhận thấy đường đi đến A theo chiều kim đồng hồ có giá là 1, trong khi theo chiều ngược lại có giá là 1+e. Do đó, đường đi đến A có giá thấp nhất của C bây giờ là theo chiều kim đồng hồ. Tương tự, B nhận thấy đường đi đến A có giá thấp nhất mới cũng theo chiều kim đồng hồ, kết quả được trình bày trong Hình 4.5b. Trong bước tiếp theo, nút B, C và D nhận thấy đường đi đến A ngược chiều kim đồng hồ có giá là 0 và tất cả các nút định lại tuyến đường theo ngược chiều kim đồng hồ. Trong bước tiếp theo B, C và D lại thay đổi việc định tuyến theo chiều kim đồng hồ.



Hình 4.5 Xung đột định tuyến

Làm sao có thể ngăn ngừa sự dao động như trên (điều này luôn xuất hiện với những thuật toán chọn độ tắc nghẽn hoặc thời gian trễ làm giá cho đường truyền). Một giải pháp được đưa ra là định giá cho đường truyền không phụ thuộc vào tải trên đường đi - một giải pháp khó có khả năng chấp nhận vì mục tiêu của định tuyến là tránh những đường truyền hay tắc nghẽn (có độ trễ cao). Một giải pháp khác là làm thế nào để tất cả các router không chạy thuật toán LS tại cùng một thời điểm. Giải pháp này dường như hợp lý hơn vì chúng ta hy vọng rằng thậm chí nếu các router có chạy thuật toán LS với cùng chu kỳ, thì thuật toán sẽ đưa ra những kết quả khác nhau tại mỗi nút.

4.2.2. Thuật toán Distance vector

Nếu thuật toán LS sử dụng thông tin về toàn bộ trạng thái mạng, thuật toán Distance vector (DV) là thuật toán lặp, không đồng bộ và phân tán. Thuật toán được xem là phân tán vì mỗi nút nhận thông tin từ những nút hàng xóm có đường kết nối trực tiếp, thực hiện các bước tính toán và phân tán kết quả tính toán tới tất cả các nút hàng xóm. Tính lặp được thể hiện ở chỗ quá trình này được thực hiện liên tục cho đến khi không còn thông tin được trao đổi giữa các cặp hàng xóm (chúng ta sẽ thấy đây là thuật toán tự kết thúc – nó tự dừng chứ không cần một tín hiệu kết thúc). Thuật toán không đòi hỏi các nút không được hoạt động trong khi trao đổi với những nút khác – đây chính là đặc điểm không đồng bộ. Chúng ta sẽ thấy thuật toán với các đặc điểm dị bộ, lặp, tự kết thúc, và phân tán thú vị và đáng quan tâm hơn so với thuật toán tập trung.

Cấu trúc dữ liệu chính trong thuật toán DV là **bảng khoảng cách (distance table)** được đặt ở mỗi nút. Trong bảng khoảng cách, mỗi hàng ứng với một nút đích trên mạng và mỗi cột ứng với một nút hàng xóm có đường kết nối trực tiếp đến. Giả sử nút X muốn định tuyến đến đích Y qua nút hàng xóm Z. Trong bảng khoảng cách của nút X, $D^X(Y, Z)$ là tổng của giá đường liên kết trực tiếp giữa X và Z - c(X, Z) với giá đường đi bé nhất từ Z đến Y. Đó là:

$$D^X(Y, Z) = c(X, Z) + \min_w\{D^Z(Y, w)\}$$

Biểu thức $\min_w$ trong đẳng thức trên được lấy trên tất cả các hàng xóm của Z (kể cả X, như chúng ta sẽ thấy dưới đây).

Đẳng thức giúp chúng ta hình dung ý tưởng của thuật toán DV - mỗi nút phải biết được giá nhỏ nhất của đường đi từ tất cả các hàng xóm của nó đến bất kỳ đích nào. Và khi giá nhỏ nhất đến đích nào đó thay đổi, nút phải báo cho tất cả các hàng xóm biết.

Trước khi trình bày thuật toán DV, xét kiến trúc mạng và bảng khoảng cách của nút E trên Hình 4.6. Đây là bảng khoảng cách của nút E sau khi thuật toán DV hội tụ. Hãy nhìn vào dòng đầu tiên với đích đến là A.

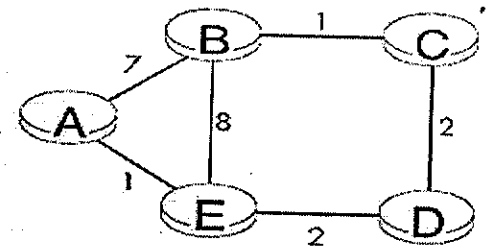
Rõ ràng giá đường kết nối trực tiếp đến A từ E phải là 1, do đó $D^E(A, A) = 1$.

Bây giờ hãy chú ý đến giá trị của $D^E(A, D)$ – giá đường đi từ E đến A qua D. Trường này trong bảng khoảng cách là giá của đường đi từ E đến D (2) cộng với giá đường đi nhỏ nhất từ D đến E. Chú ý rằng giá thấp nhất của đường đi từ D đến A là 3 - đường đi qua E! Do vậy giá thấp nhất từ E đến A qua D là 5. Chúng ta không quan tâm - mặc dù có thể băn khoăn ở đây - là đường đi từ E đến A qua D lại quay lại E.

Tương tự, trường trong bảng khoảng cách với đường đi qua B: $D^E(A, B) = 14$. Hãy chú ý rằng tại sao ở đây không phải là 15?

Các vòng tròn trong bảng khoảng cách là giá nhỏ nhất của đường đi đến các đích (ứng với các hàng). Cột ứng với vòng tròn xác định nút tiếp theo trên đường đi đến đích có giá thấp nhất. Từ đó có thể dễ dàng xây dựng bảng định tuyến cho mỗi nút (với một đích cụ thể nào đó, cần gửi gói tin ra theo đường ra nào).

Trong khi trình bày bảng vector cho nút E ở trên, chúng ta vẫn có được một cái nhìn toàn cục, biết được giá của tất cả các liên kết trên mạng. Thuật toán DV phân tán được trình bày ở đây không sử dụng đến những thông tin tổng thể như vậy.



destination	$D^E()$	cost to destination via A	B	D
	A	①	14	5
	B	7	8	⑤
	C	6	9	④
	D	4	11	②

Hình 4.6

Thuật toán distance vector(DV)

Tại mỗi nút X:

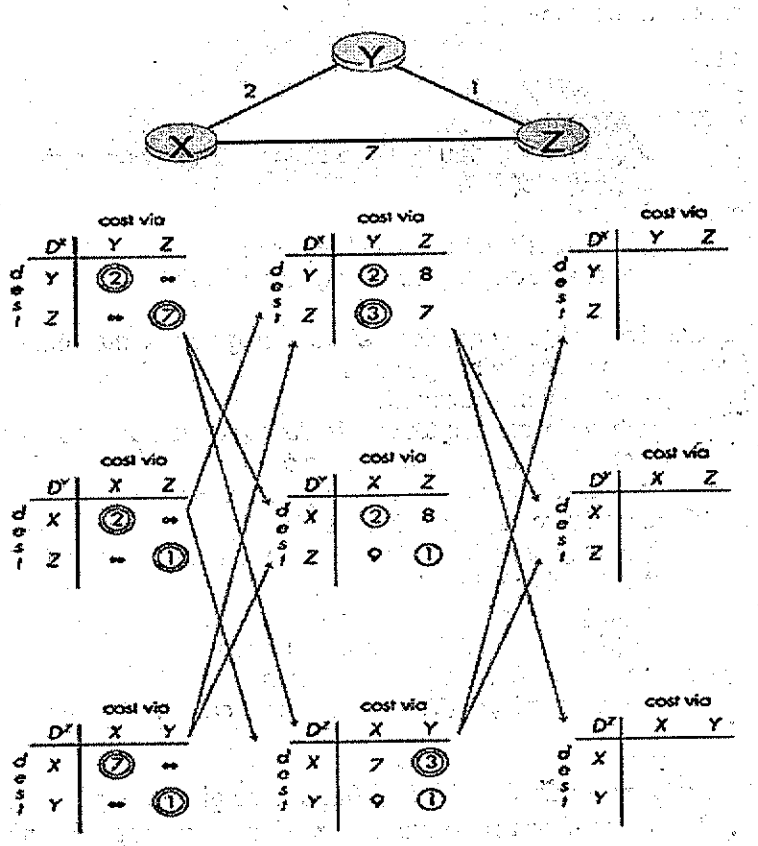
1:Khởi tạo:
2: for (tất cả các nút kề với v)
3: $D^X(*, v) = \infty$ /* dấu "*" là để chỉ cho tất cả các hàng*/
4: $D^X(v, v) = c(X, v)$
5: for (tất cả các đích Y)
6: gửi $\min_w D(Y, w)$ đến mỗi hàng xóm
7: /* w chạy trong tập các hàng xóm của X*/
8:Lặp
9: Đợi cho đến khi (thấy giá liên kết đến hàng xóm V thay đổi hoặc nhận được sự cập nhật của hàng xóm V)
if (c(X, V) thay đổi một lượng d)
/* thay đổi giá của tất cả các đường đi đến đích qua v bằng d*/
/* chú ý: d có thể dương hoặc âm*/
for (tất cả các đích y: $D^X(y, V) = D^X(y, V) + d$
else if (nhận được cập nhật từ V đến Y)
/* đường đi ngắn nhất từ V đến nút nào đó Y thay đổi*/
/* V đã gửi giá trị mới của $\min_w D^V(Y, w)$*/
/* gọi giá trị mới nhận được là "newval"*/
for (đích y): $D^X(Y, V) = c(X, V) + newval$
if (chúng ta có $\min_w D^X(Y, w)$ mới cho đích Y nào đó)
gửi giá trị $\min_w D^X(Y, w)$ mới đến tất cả các hàng xóm.
Mãi mãi

Thật vậy, mỗi nút sẽ chỉ biết thông tin về giá đường liên kết tới nút hàng xóm cũng như thông tin nó nhận được từ những hàng xóm này. Thuật toán Distance vector chúng ta sẽ nghiên cứu còn được gọi là thuật toán

Bellman–Ford. Nó được áp dụng trong nhiều giao thức định tuyến trong thực tế, bao gồm: Internet BGP, ISO IDRP, Novell IPX và mạng ARPAnet.

Bước mấu chốt nằm từ dòng 15 đến 21, ở đó nút cập nhật bảng khoảng cách khi nhận được sự thay đổi về giá của liên kết đến hàng xóm hoặc nhận được thông tin cập nhật từ hàng xóm. Một bước quan trọng khác là dòng 24, ở đó nút gửi cập nhật đến tất cả các hàng xóm nếu đường đi có giá nhỏ nhất đến đích nào đó bị thay đổi.

Hình 4.7 minh hoạ hoạt động của thuật toán DV cho mạng đơn giản gồm 3 nút. Hoạt động của thuật toán được thực hiện một cách đồng bộ: tất cả các nút đồng thời nhận được thông điệp từ hàng xóm của chúng, tính toán bảng khoảng cách mới và báo cho các hàng xóm về sự thay đổi giá đường đi ngắn nhất. Sau khi nghiên cứu ví dụ này, bạn có thể nghĩ rằng thuật toán hoạt động đúng trong chế độ không đồng bộ - việc tính toán và cập nhật sự thay đổi của mỗi nút có thể diễn ra tại bất kỳ thời điểm nào.



Hình 4.7 Ví dụ về thuật toán Distance Vecto

Các ô khoanh tròn trong Hình 4.7 ứng với giá nhỏ nhất đến đích nào đó nằm trong hàng tương ứng. Khoanh hai vòng tròn biểu diễn giá nhỏ nhất mới được xác định (trong dòng 4 hoặc dòng 21 của thuật toán DV). Khi đó các thông tin cập nhật sẽ được gửi đến các nút hàng xóm (dòng 24 của thuật toán DV) - ứng với mũi tên giữa các cột trong Hình 4.7.

Cột ngoài cùng bên trái trong Hình 4.7 là các bảng khoảng cách của nút X, Y và Z sau bước khởi tạo.

Bây giờ hãy xem làm thế nào để nút X tính lại bảng khoảng cách (cột giữa của Hình 4.7) sau khi nhận được thông tin cập nhật từ nút Y và Z. Khi nhận được cập nhật từ Y và Z, X thực hiện dòng 21 của thuật toán DV:

$$D^X(Y, Z) = c(X, Z) + \min_w D^Z(Y, w)$$
$$= 7 + 1$$
$$= 8$$
$$D^X(Z, Y) = c(X, Y) + \min_w D^Y(Z, w)$$
$$= 2 + 1$$
$$= 3$$

X biết giá trị $\min_w D^Z(Y, w)$ và $\min_w D^Y(Z, w)$ vì nút Z và Y gửi những giá trị này đến X (và X cũng nhận được theo dòng 10 của thuật toán DV). Việc tính lại bảng khoảng cách của Y và Z ở cột giữa trong Hình 4.7 được thực hiện tương tự.

Giá trị $D^X(Z, Y) = 3$ có nghĩa là giá nhỏ nhất từ X đến Z giảm từ 7 xuống 3. Do đó, X gửi cập nhật đến Y và Z để thông báo cho chúng giá thấp nhất mới đến Z. Chú ý X không cần cập nhật cho Y, Z về giá của nó đến Y vì giá trị này không bị thay đổi. Khi Y tính lại bảng khoảng cách không phát hiện ra thay đổi, do đó Y không gửi cập nhật đến X và Z.

Quá trình nhận cập nhật từ hàng xóm, tính lại bảng khoảng cách và cập nhật các thay đổi đến hàng xóm được thực hiện cho đến khi không còn thông điệp nào được trao đổi. Trong trường hợp này vì không có thông tin cập nhật được gửi nên các nút không phải tính lại bảng khoảng cách và thuật toán ở trạng thái không hoạt động: tất cả các nút ở trạng thái đợi trong dòng 9 của thuật toán DV. Thuật toán DV sẽ ở trong trạng thái không hoạt động cho đến khi giá của một liên kết nào đó thay đổi.

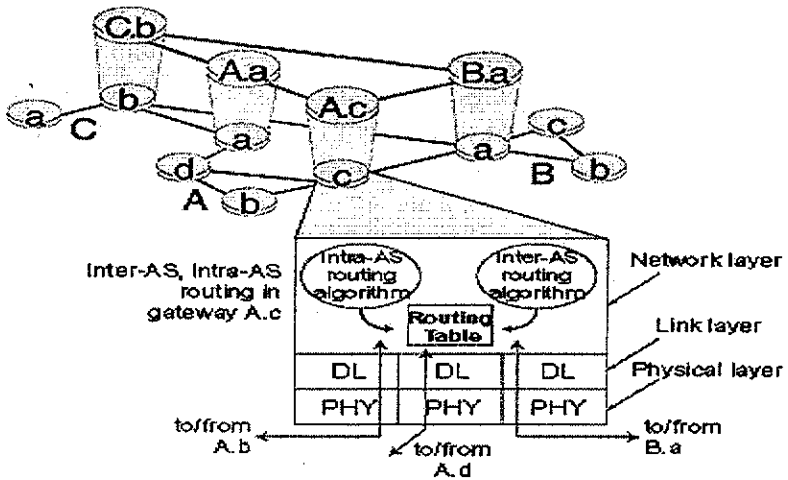
4.3 ĐỊNH TUYẾN PHÂN CẤP

Trong các phần trước, chúng ta đã thấy mạng là tập hợp các router liên kết với nhau. Tất cả các router sẽ giống nhau nếu như đều sử dụng cùng một thuật toán định tuyến để xác định đường đi trên toàn bộ hệ thống mạng.

Trong thực tế, mô hình mạng như vậy và việc xem các router giống hệt nhau cùng thực hiện một thuật toán định tuyến quá đơn giản với hai lý do quan trọng sau:

Phạm vi (scale): Khi số lượng các router lớn, khối lượng thông tin phải tính toán, lưu trữ và trao đổi giữa các bảng chứa thông tin định tuyến trên mỗi router (ví dụ các cập nhật về đường đi ngắn nhất) cũng trở nên cực lớn. Mạng Internet ngày nay bao gồm hàng triệu router liên kết với nhau và hơn 50 triệu máy tính. Lưu trữ thông tin về tất cả các máy tính cũng như các router đòi hỏi một lượng bộ nhớ khổng lồ. Các thông tin trao đổi cập nhật giữa các router sẽ "ngốn" toàn bộ băng thông của đường truyền. Thuật toán distance vector trên hàng triệu router chắc chắn sẽ không bao giờ hội tụ. Do đó nảy sinh ra nhu cầu làm giảm độ phức tạp trong việc xác định đường đi trên một mạng lớn như Internet.

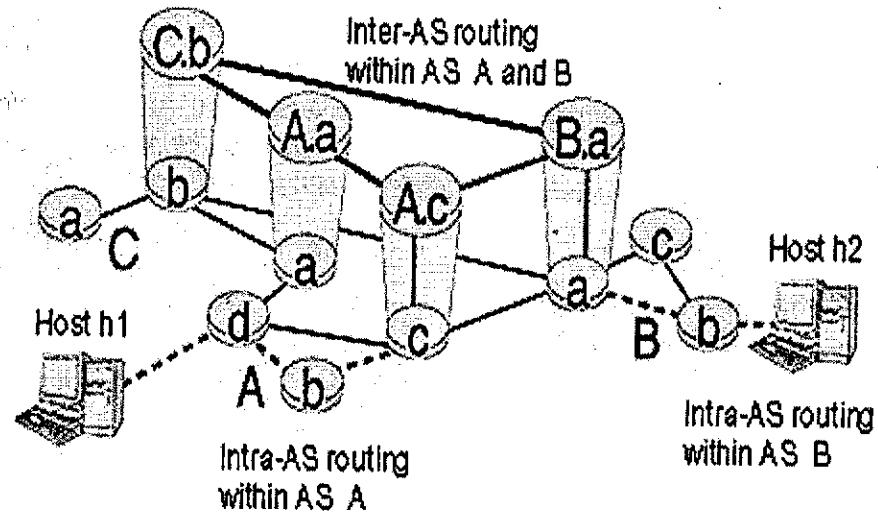
Quản trị (Administrative automomy): Mặc dù các nhà thiết kế thường bỏ qua yêu cầu của các tổ chức - chẳng hạn khả năng lựa chọn thuật toán định tuyến hay che dấu cấu trúc mạng bên trong của tổ chức với bên ngoài – nhưng trên thực tế đây là những vấn đề quan trọng. Lý tưởng mà nói, một tổ chức phải giữ khả năng quản trị và kiểm soát mạng máy tính của mình nhưng vẫn có khả năng kết nối với các mạng bên ngoài.



Hình 4.8 Ví dụ miền tự quản

Cả hai vấn đề trên đều có thể giải quyết bằng cách nhóm các router thành các vùng hay **Miền tự quản (Autonomous System - AS)**. Các router trong cùng AS sử dụng cùng một thuật toán định tuyến (ví dụ như thuật toán LS hay DV) và biết đầy đủ về nhau (giống trường hợp đã trình bày ở trước). Thuật toán định tuyến chạy trong mỗi AS được gọi là **intraautonomous system routing protocol**. Dĩ nhiên cần phải kết nối các AS với nhau – và vì thế một số router trong AS có thêm nhiệm vụ định tuyến gói tin ra ngoài AS (đích nằm trong một AS khác). Các router định tuyến gói tin ra phía ngoài như vậy được gọi là gateway router. Để định tuyến gói tin đi giữa các AS (có thể phải đi qua nhiều AS trên toàn bộ tuyến đường), các gateway router phải biết cách xác định đường đi giữa các AS. Thuật toán định tuyến được sử dụng tại các gateway router là interautonomous system routing protocol.

Nói tóm lại, vấn đề phạm vi và quản trị được giải quyết bằng cách sắp xếp router vào các miền tự quản (AS). Tất cả router trong một AS sử dụng cùng một thuật toán định tuyến. Các gateway router trên mỗi AS sử dụng thuật toán interautonomous system routing để định tuyến giữa các AS. Vấn đề phạm vi được giải quyết ở chỗ mỗi router chỉ cần biết về các router khác trong cùng AS và gateway router của AS đó (chứ không cần phải biết tất cả các router). Vấn đề quản trị được giải quyết vì tổ chức có thể lựa chọn sử dụng bất kỳ thuật toán định tuyến intra-AS nào - miễn là thuật toán inter-AS mà nó sử dụng có khả năng liên kết với các AS khác.



Hình 4.9 Ví dụ định tuyến phân cấp

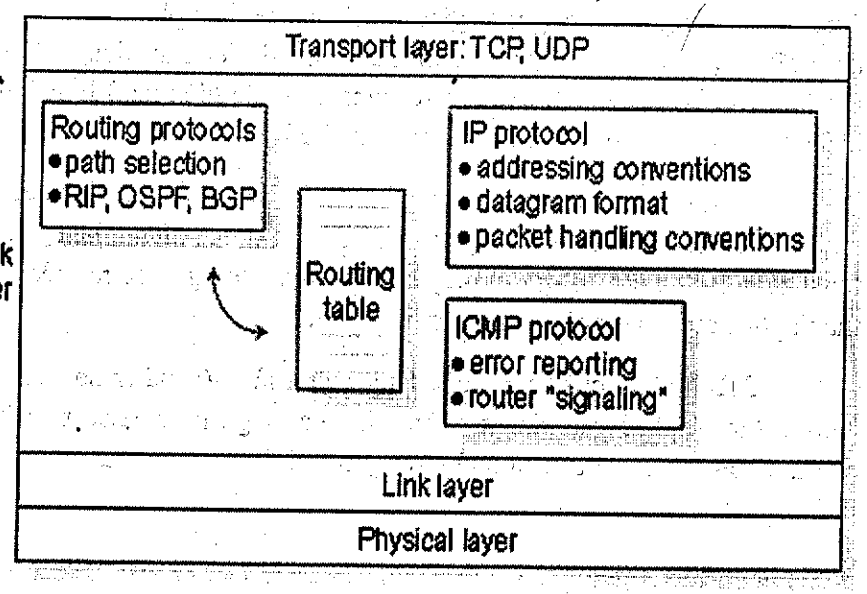
Trên Hình 4.9 có ba AS: A, B và C. Miền A có 4 router: A.a, A.b, A.c, và A.d, sử dụng cùng một thuật toán intra-AS của miền A. Bốn router này đều có đầy đủ các thông tin về nhau cũng như các liên kết trong miền A. Tương tự miền B có 3 và miền C có 2 router. Các thuật toán intra-AS trong các miền A, B, C không nhất thiết giống nhau. Các gateway router là A.a, A.c, B.a, và C.b. Ngoài các thuật toán intra-AS để trao đổi với các router trong miền, bốn gateway router này phải sử dụng thuật toán inter-AS để định tuyến giữa các AS. Về mặt topo, chúng sử dụng giao thức inter-AS, được minh họa bằng các đường kết nối mầu đen đậm ở mức cao hơn. Các đường kết nối này có thể là đường kết nối vật lý thực sự (giữa A.c và B.a), có thể là đường ảo (ví dụ giữa A.c và A.a). Trên hình vẽ chúng ta cũng thấy A.c sử dụng cả intra-AS để định tuyến với A.b, A.d và inter-AS để định tuyến với B.a

Giả sử máy tính h1 nối với router A.d cần gửi gói tin tới máy tính h2 trong AS B, như trên Hình 4.9. Bảng định tuyến tại A.d cho biết, router A.c chịu trách nhiệm gửi gói tin ra bên ngoài AS. Gói tin từ A.d tới router đầu tiên A.c sử dụng giao thức định tuyến intra-AS của A. Có một điểm cực kỳ quan trọng cần chú ý là router A.d không cần biết gì về cấu trúc nội tại trong miền B và C và cũng như topo giữa ba miền A, B và C. Router A.c nhận gói tin, xác định đích của gói tin đó nằm ngoài miền A (miền B), bảng định tuyến của inter-AS sẽ xác định rằng để gửi tới miền B thì phải chuyển tới B.a. Khi gói tin tới B.a, giao thức inter-AS xác định rằng gói tin này tới máy tính nào đó trong miền B và chuyển cho giao thức intra-AS của B. Cuối cùng router B.a chuyển gói tin đó tới máy tính đích h2 sử dụng giao thức intra-AS của B. Trong Hình 4.9, các giao thức intra-AS của A và B là các đường kẻ đứt, giao thức inter-AS giữa các miền là đường kẻ đậm.

4.4 INTERNET PROTOCOL

Trong các phần trước, chúng ta đã trình bày các nguyên lý chung của tầng mạng mà chưa nói đến bất kỳ một kiến trúc mạng cụ thể nào. Chúng ta cũng nói tới các mô hình dịch vụ khác nhau của tầng mạng, các thuật toán định tuyến xác định đường đi giữa thiết bị gửi và thiết bị nhận, phân cấp hệ thống mạng để giải quyết vấn đề phạm vi. Trong phần này,

chúng ta sẽ nói tới tầng mạng của Internet – mà một phần trong đó được xem là tầng IP. Chúng ta sẽ thấy IP chỉ là một phần (dù là phần quan trọng) trong kiến trúc tầng mạng của Internet (xem Hình 4.10).



Hình 4.10 Vị trí và các thành phần của Tầng mạng

Như đã nói trong trong phần 4.1, tầng mạng của Internet sử dụng dịch vụ chuyển mạch gói (datagram) chứ không phải dịch vụ chuyển mạch kênh (VC). Tại máy gửi, khi nhận được một segment từ tầng giao vận, tầng mạng đặt segment trong gói dữ liệu IP (IP datagram) với các trường địa chỉ gửi, địa chỉ nhận... và gửi datagram này tới router đầu tiên trên đường tới đích. Điều này tương tự như một người sau khi viết thư, bỏ lá thư vào phong bì ghi địa chỉ người nhận và thả phong bì thư vào hộp thư. Tầng mạng của Internet cũng giống như hệ thống bưu điện đều không có bất kỳ một sự liên hệ trước nào với bên nhận trước khi chuyển "bưu kiện" (datagram hay lá thư) tới phía nhận. Hơn nữa, như đã nói trong phần 4.1, tầng mạng của Internet và bưu điện sẽ chỉ cung cấp một dịch vụ kiểu "cố gắng tối đa", nghĩa là không có bất kỳ đảm bảo nào về việc gói tin có đến đích hay không, đến trong một khoảng thời gian bao lâu hay đến theo đúng thứ tự gửi.

Trong Hình 4.10, tầng mạng trong kiểu mạng chuyển mạch gói giống như mạng Internet có ba thành phần chính:

Thành phần thứ nhất là giao thức mạng: xác định địa chỉ tầng mạng; ý nghĩa các trường trong datagram (là gói dữ liệu - PDU của tầng mạng); hành động của router và thiết bị đầu cuối khi nhận được datagram. Giao thức mạng trong Internet gọi là giao thức Internet hay phổ biến hơn là giao thức **IP**. Hiện nay có hai phiên bản giao thức IP được sử dụng. Các phần 4.4.1 đến 4.4.4 nói về IPv4 (được sử dụng chủ yếu hiện nay) [RFC 701] và phần 4.7 nói tới IPv6 (được đưa ra để thay thế IPv4).

Thành phần thứ hai của tầng mạng là bộ phận xác định đường đi: xác định tuyến đường của datagram trên đường đi tới đích. Phần 4.2 nói về thuật toán tạo ra các bảng định tuyến như vậy. Chúng ta sẽ nghiên cứu các thành phần của bộ định tuyến trong phần 4.5.

Thành phần cuối cùng của tầng mạng là chức năng báo lỗi và khả năng trả lời một số yêu cầu về thông tin của tầng mạng. Giao thức báo lỗi của Internet – ICMP được nói đến trong mục 4.4.5.

4.4.1 Địa chỉ IPv4

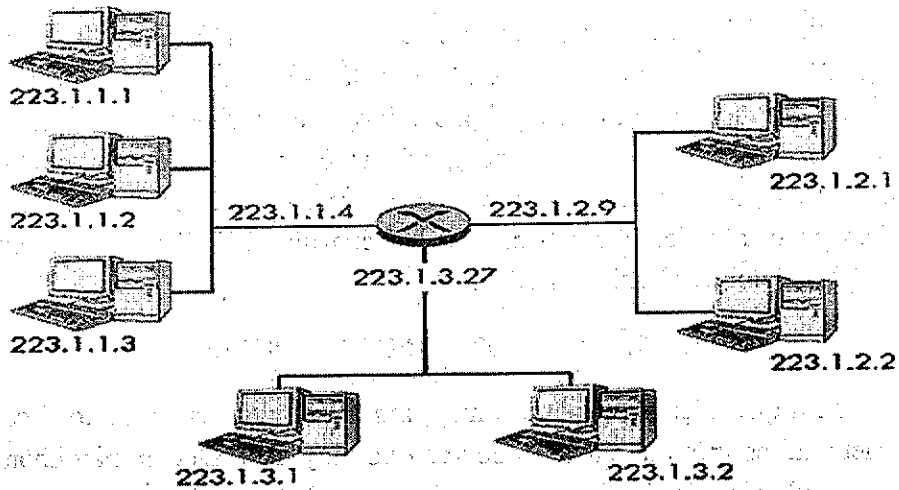
Đầu tiên chúng ta sẽ nói về địa chỉ IPv4. Mặc dù vấn đề địa chỉ tương đối đơn giản song mối quan hệ giữa địa chỉ và giao thức tầng mạng lại quan trọng.

Trước khi thảo luận về địa chỉ IP, chúng ta hãy xét máy tính và router nối vào mạng như thế nào. Máy tính thường có một đường kết nối duy nhất vào hệ thống mạng mà qua đó thực thể IP trong máy tính sẽ sử dụng khi gửi datagram. Nằm giữa máy tính và đường kết nối vật lý là một **giao diện ghép nối (interface)**. Ngược lại router khác hoàn toàn máy tính. Công việc của router là chuyển một datagram từ kết nối (incoming link) này tới kết nối khác (outgoing link). Router có thể có nhiều kết nối, và bộ phận nằm giữa router với kết nối cũng được gọi là giao diện. Như vậy router có nhiều giao diện, mỗi giao diện ứng với một kết nối. Vì tất cả các máy tính và router đều phải có khả năng gửi và nhận IP datagram nên mỗi giao diện phải có một địa chỉ IP. Do đó địa chỉ IP ứng với giao diện chứ không phải với máy tính hay router.

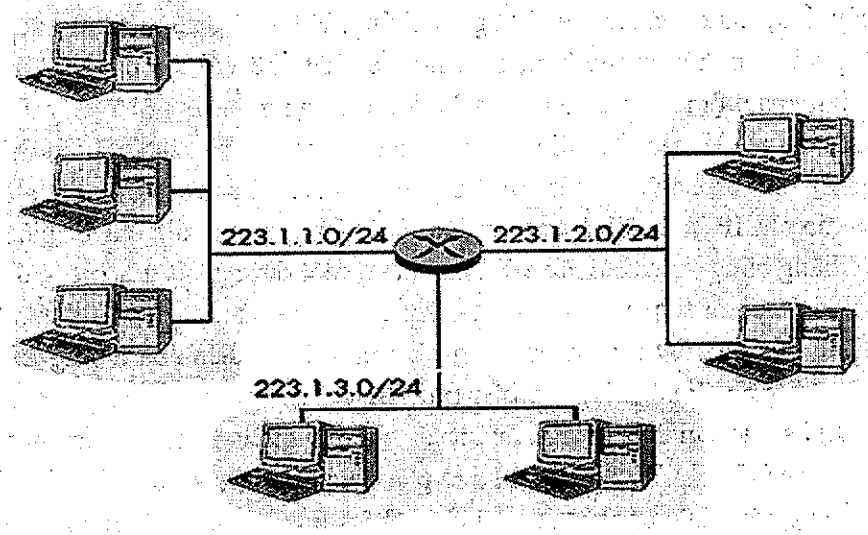
Địa chỉ IP có độ dài 32 bit (4 byte) và do đó không gian địa chỉ có 2^{32} địa chỉ. Địa chỉ IP được viết theo ký pháp **dấu chấm thập phân (dotted-decimal notation)** vì mỗi byte của địa chỉ được viết dưới dạng thập phân và phân cách với các byte khác bằng ký tự chấm (.). Xét địa chỉ IP 193.32.216.9. 193 là số thập phân ứng với nhóm 8 bít đầu của địa chỉ, 32 là số thập phân ứng với nhóm 8 bít thứ hai của địa chỉ... Bởi vậy địa chỉ 193.32.216.9 đổi ra hệ nhị phân sẽ là:

11000001 00100000 11011000 00001001.

Mỗi giao diện ghép nối của máy tính hay router trên mạng toàn cầu Internet phải có một địa chỉ IP được xác định duy nhất. Địa chỉ này không thể chọn một cách tùy ý mà phụ thuộc vào "mạng" mà nó kết nối vào. Trong ngữ cảnh này, thuật ngữ "mạng" không có ý là một cấu trúc tổng thể gồm các máy tính, router và các liên kết giữa chúng. Hiện tại thuật ngữ này được sử dụng với ý nghĩa cụ thể hơn, có quan hệ chặt chẽ với địa chỉ IP. Hình 4.11 minh họa một router có 3 giao diện được sử dụng để kết nối 7 máy tính. Quan sát địa chỉ IP của giao diện ứng với mỗi máy tính và router. Giao diện của 3 máy tính ở phần trên bên trái trong Hình 4.11 và router nối với chúng đều có địa chỉ IP là 223.1.1.xxx.: nghĩa là 24 bit đầu của địa chỉ IP giống nhau. Chúng cũng được kết nối với nhau bằng một đường kết nối vật lý duy nhất (trong trường hợp này là môi trường quảng bá sử dụng cáp Ethernet) mà không cần qua bất kỳ router trung gian nào. Giao diện của những máy tính này và giao diện phía trên bên trái của router tạo nên **mạng IP (IP network)** hay đơn giản là **mạng**. 24 bit địa chỉ đầu giống nhau là phần mạng trong cấu trúc địa chỉ IP, 8 bit còn lại là phần máy tính (host) của địa chỉ IP. Chính mạng này cũng có một địa chỉ là 22.3.1.1.0/24 trong đó kí hiệu /24 là **mặt nạ mạng (network mask)** với ý nghĩa 24 bit đầu tiên của địa chỉ 32 bit xác định địa chỉ mạng. Những bit này cũng được xem là **tiền tố mạng (network prefix)**. Mạng 22.3.1.1.0/24 gồm giao diện của 3 máy tính (223.1.1.1; 223.1.1.2; 223.1.1.3) và một giao diện của router (223.1.1.4). Bất kỳ máy tính nào nối với mạng 223.2.2.0/24 đều phải có địa chỉ dưới dạng 223.1.1.xxx. Trên Hình 4.11 còn có hai mạng khác: 223.1.2.0/24 và 223.1.3.0/24 . Hình 4.12 minh hoạ 3 mạng IP trong Hình 4.11.



Hình 4.11



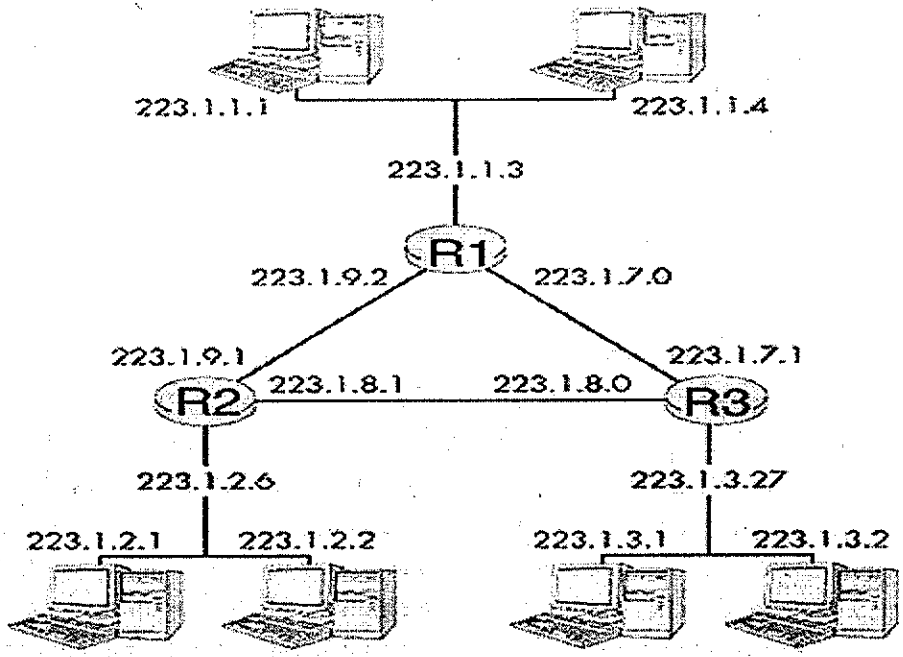
Hình 4.12

Định nghĩa IP về "mạng" không chỉ với phân đoạn mạng Ethernet nối nhiều máy tính với một router. Trên Hình 4.13 ba router đôi một nối với nhau qua các đường liên kết điểm tới điểm (point-to-point). Mỗi router có ba giao diện, hai giao diện kết nối tới hai router kia và một giao diện dành cho kết nối quảng bá với các máy tính. Có bao nhiêu mạng IP ở đây? Ba mạng 223.1.1.0/24, 223.1.2.0/24 và 223.1.3.0/24 tương tự các mạng nói tới trong

Hình 4.11. Nhưng chú ý có thêm 3 mạng nữa trong Hình 4.13 mạng 223.1.9.0/24 cho hai giao diện nối router R1 và R2, mạng 223.1.8.0/24 cho hai giao diện nối router R2 và R3 và mạng 223.1.7.0/24 ứng với hai giao diện nối router R3 và R1

Với một hệ thống liên mạng gồm nhiều router và máy tính, chúng ta có thể sử dụng một công thức để xác định các mạng trong hệ thống. Chúng là loại bỏ tất cả giao diện của các máy tính và router. Khi đó sẽ tạo ra các mạng cô lập, mỗi mạng cô lập đó được coi là một mạng IP. Áp dụng cách thức này trong ví dụ trên Hình 4.13, chúng ta có 6 mạng IP tách biệt. Mạng toàn cầu Internet gồm hàng triệu hệ thống mạng như vậy. Ký pháp mạng và địa chỉ mạng có vai trò then chốt trong kiến trúc định tuyến Internet. Sau khi đã có định nghĩa về mạng, chúng ta tiếp tục trình bày chi tiết về địa chỉ IP. Kiến trúc địa chỉ Internet đầu tiên đưa ra 4 lớp địa chỉ minh họa trên Hình 4.14. Lớp địa chỉ thứ năm, bắt đầu bằng 11110 được dự trữ cho việc sử dụng sau này. Với lớp địa chỉ A, 8 bit đầu là địa chỉ mạng và 24 bit cuối là địa chỉ cho giao diện (máy tính) trong mạng. Như thế có 2^7 mạng lớp A (bit đầu tiên trong 8 bit địa chỉ mạng luôn nhận giá trị 0), mỗi mạng lớp A lại có thể có 2^{24} giao diện. Lớp B có 2^{14} mạng, mỗi mạng có 2^{16} giao diện. Lớp địa chỉ C dùng 24 bit làm địa chỉ mạng và chỉ có 8 bit làm địa chỉ máy. Lớp D dự trữ làm địa chỉ multicast.

Bốn lớp địa chỉ trình bày trên hiện tại không còn được áp dụng trong kiến trúc địa chỉ IP nữa. Điều kiện phần mạng của địa chỉ IP có độ dài là một, hai hoặc ba byte không hợp lý khi số lượng các tổ chức với mạng cỡ nhỏ hay trung bình ngày càng tăng. Mạng lớp C (/24) chỉ có thể có $2^8 - 2 = 254$ máy tính (2 trong số 2^8 địa chỉ được dự trữ cho một số mục đích khác) - một số lượng quá nhỏ với nhiều tổ chức. Tuy nhiên, mạng lớp B (/16) có thể tới 65.634 máy tính lại là quá lớn. Một tổ chức với 2000 máy tính phải sử dụng địa chỉ lớp B (/16). Với kiểu gán địa chỉ như vậy thì không gian địa chỉ lớp B sẽ nhanh chóng bị cạn kiệt nhưng không gian địa chỉ lại không được sử dụng hiệu quả. Ví dụ, tổ chức sử dụng địa chỉ lớp B cho 2000 máy tính thì sẽ có khoảng 63000 địa chỉ còn lại bị lãng phí trong khi đáng.ra có thể phân phối cho các tổ chức khác.



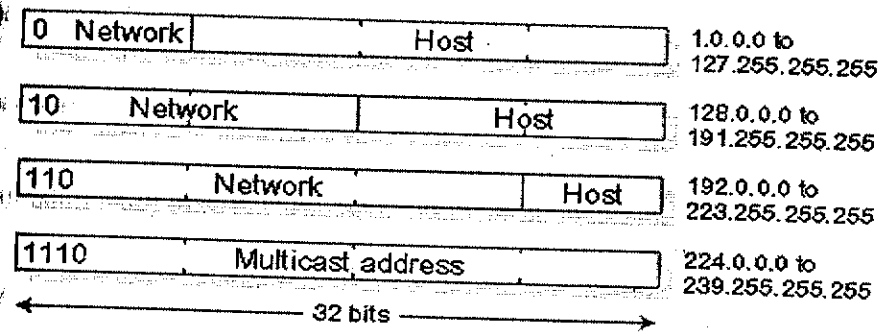
Hình 4.13 Có 6 mạng IP

Năm 1993, IETF chuẩn hóa **Định tuyến liên miền không phân lớp (Classless interdomain routing - CIDR-** phát âm là "cider") [RFC 1519]. Với cơ chế không phân lớp, phần mạng của địa chỉ IP có thể có độ dài tùy ý, không nhất thiết phải là 8, 16 hay 24 bit. Khuôn dạng của một địa chỉ không phân lớp sẽ là **a.b.c.d/x**, trong đó x là số lượng bit dùng để làm địa chỉ mạng. Trong ví dụ trước, tổ chức cần không gian địa chỉ cho 2000 máy tính chỉ cần một không gian 2.048 địa chỉ dưới dạng a.b.c.d/21, cho phép 63000 địa chỉ còn lại được phân phối cho các tổ chức khác. Trong trường hợp này, 21 bit đầu tiên xác định địa chỉ mạng của tổ chức và tất cả địa chỉ IP của máy tính trong tổ chức đều phải có phần địa chỉ mạng giống nhau. 11 bit còn lại xác định cụ thể máy tính nào trong tổ chức. Trên thực tế, tổ chức có thể tiếp tục chia 11 bit đó trong thủ tục **tạo mạng con (subnetting)** [RFC 950] để tạo ra các mạng con bên trong mạng a.b.c.d/21.

Trên thực tế

Ví dụ trình bày việc một ISP kết nối 8 tổ chức vào mạng Internet minh họa việc phân phối địa chỉ không phân lớp có thể đáp ứng khả năng định tuyến phân cấp như thế nào. Giả sử như trong hình 4.19, ISP Fly-By-Night quảng cáo với thế giới bên ngoài là nó có thể gửi các datagram đến các máy có 20 bit địa chỉ đầu trùng với 200.23.16.0/20. Thế giới bên ngoài không cần biết trong không gian địa chỉ 200.23.16.0/20 thực tế có 8 tổ chức con, mỗi tổ chức có hệ thống mạng riêng. Khả năng sử dụng một tiền tố mạng (prefix) quảng cáo cho nhiều mạng được gọi là **nhóm đường (route aggregation hay route summarization)**.

Nói chung nhóm đường hoạt động tốt khi không gian địa chỉ được cấp phát cho ISP và sau đó ISP cấp phát tiếp cho các tổ chức khách hàng. Nhưng chuyện gì xảy ra nếu địa chỉ không được phân phối theo cách thức phân cấp? Chẳng hạn tổ chức 1 không chấp nhận dịch vụ với chất lượng thấp của ISP Fly-By-Night và quyết định lựa chọn ISP mới là ISPs-R-Us? Như minh họa trên Hình 4.15, ISPs-R-Us có không gian địa chỉ là 199.31.0.0/16, nhưng không gian địa chỉ IP của tổ chức 1 lại nằm bên ngoài không gian địa chỉ này. Sẽ làm gì ở đây ? Chắc chắn, tổ chức 1 có thể cấu hình lại tất cả router và máy tính để có địa chỉ IP trong miền địa chỉ của ISPs–R –Us. Nhưng đây là giải pháp tốn kém, và trong tương lai rất có thể tổ chức 1 lại lựa chọn một ISP mới. Giải pháp ở đây sẽ là cho phép tổ chức 1 giữ lại địa chỉ 200.23.18.0/23. Trong trường hợp này, như trong hình 4.15, Fly-By-Night-ISP tiếp tục quảng cáo không gian địa chỉ 200.23.16.0/20 và ISPs-R-Us tiếp tục quảng cáo không gian địa chỉ 199.31.0.0/16. Tuy nhiên ISPs-R-U bây giờ cũng quảng cáo không gian địa chỉ cho tổ chức 1: 200.23.18.0/23. Khi những router khác trong Internet nhận được quảng cáo không gian địa chỉ 200.23.16.0/20 (từ Fly-By-Night-ISP) và 200.23.18.0/23 (từ ISPs–R-Us) và muốn định tuyến tới một địa chỉ nào đó trong 200.23.18.0/23, chúng sẽ sử dụng **quy tắc địa chỉ tiền tố dài nhất (longest prefix rule)** để định tuyến tới ISPs-R-Us, vì ISPs-R-Us quảng cáo địa chỉ có tiền tố dài nhất ứng với địa chỉ đích.



Hình 4.14 Cấu trúc địa chỉ mạng với 4 lớp A,B,C và D

Gán địa chỉ cho mỗi giao diện

Sau khi biết được địa chỉ IP, bạn có thể hỏi bằng cách nào máy tính nhận được địa chỉ IP? Địa chỉ IP có hai phần: Phần mạng và phần máy tính. Phần máy tính của địa chỉ có thể được cấp phát theo nhiều cách như sau:

Cấu hình bằng tay. Địa chỉ IP được người quản trị hệ thống cấu hình vào máy tính (thường trong file cấu hình).

Giao thức cấu hình địa chỉ động (Dynamic host configuartion protocol -DHCP) [RFC 2131]. DHCP là phiên bản mở rộng của giao thức BOOTP [RFC 1542]. Với DHCP, khi máy DHCP phục vụ (server) trong mạng (mạng cục bộ chẳng hạn) nhận yêu cầu DHCP từ một máy khách, nó sẽ phân phối (gán) một địa chỉ IP cho máy khách yêu cầu. DHCP được sử dụng rộng rãi trong mạng cục bộ hay truy cập Internet từ nhà (kết nối modem với ISP).

Để có địa chỉ mạng không đơn giản. Người quản trị mạng phải liên lạc với ISP của mình. ISP đã được phân phối một không gian địa chỉ tương đối lớn và sẽ cấp một phần trong không gian địa chỉ này cho tổ chức thuê bao. Ví dụ ISP được phân phối không gian địa chỉ 200.23.16.0/20. Đến lượt mình ISP chia không gian địa chỉ này thành 8 không gian địa chỉ nhỏ hơn và cấp phát 8 không gian này cho các tổ chức thuê bao như dưới đây (để tiện theo dõi, phần mạng trong địa chỉ được gạch chân)

Không gian địa chỉ của ISP: 11001000 00010111 00010000 00000000 200.2.3.16.0/20

Tổ chức 0 11001000 00010111 00010000 00000000
200.23.16.0/23

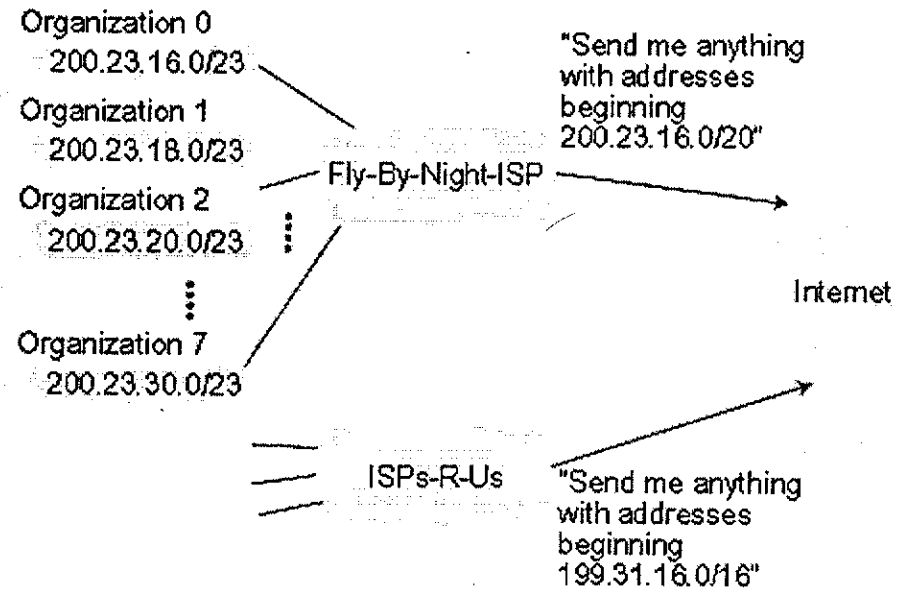
Tổ chức 1 11001000 00010111 00010010 00000000
200.23.18.0/23

Tổ chức 7 11001000 00010111 00010100 0000000
200.23.30.0/23

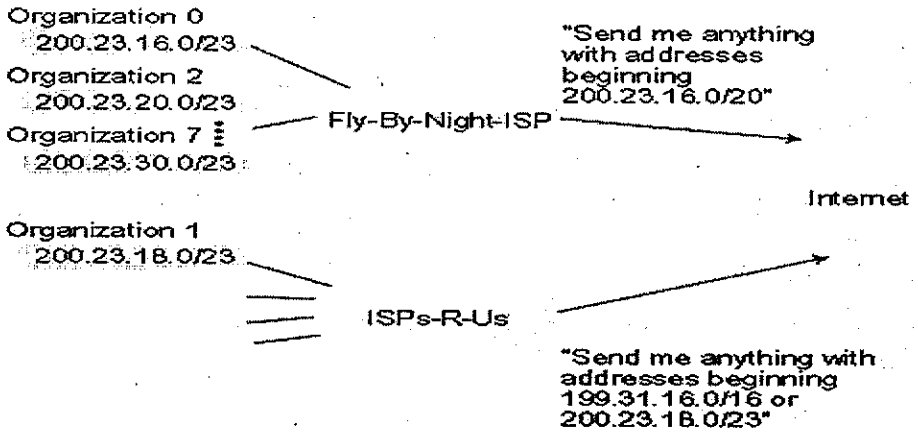
Vậy làm thế nào để ISP nhận được không gian địa chỉ? Địa chỉ IP được tổ chức ICANN (The Internet Corporation for Assigned Names and

Numbers) [ICANN 2000] quản lí theo các nguyên tắc chỉ đạo ghi trong RFC 2050. Vai trò của tổ chức phi lợi nhuận ICANN không chỉ cấp phát địa chỉ IP mà còn quản trị các root DNS server. Nó chịu trách nhiệm đặt tên miền cũng như giải quyết các tranh chấp về tên miền. Hiện nay việc phân phối địa chỉ IP được cơ quan đăng kí Internet cấp vùng quản lý. Giữa năm 2000, có 3 cơ quan đăng kí như vậy: American Registry for Internet Number (**ARIN**) cho châu Mĩ và một số phần của châu Phi. Reseaux IP Europeans (**RIPE**) cho châu Âu và một số nước chung quanh và Asia Pacific Network Information Center (**APNIC**) cho khu vực châu Á.

Các máy tính có khả năng di động (mobile) có thể thay đổi mạng theo cách động (khi di chuyển) hoặc tĩnh (theo thời gian). Khi định tuyến, phải xác định mạng trước và sau đó mới tới máy tính trong mạng, nên địa chỉ IP của máy tính di động sẽ phải thay đổi khi máy tính thay đổi mạng. Các kỹ thuật xử lý trường hợp này đang được phát triển [RFC 2002, RFC 2131].



Hình 4.15 Ví dụ 2 ISP



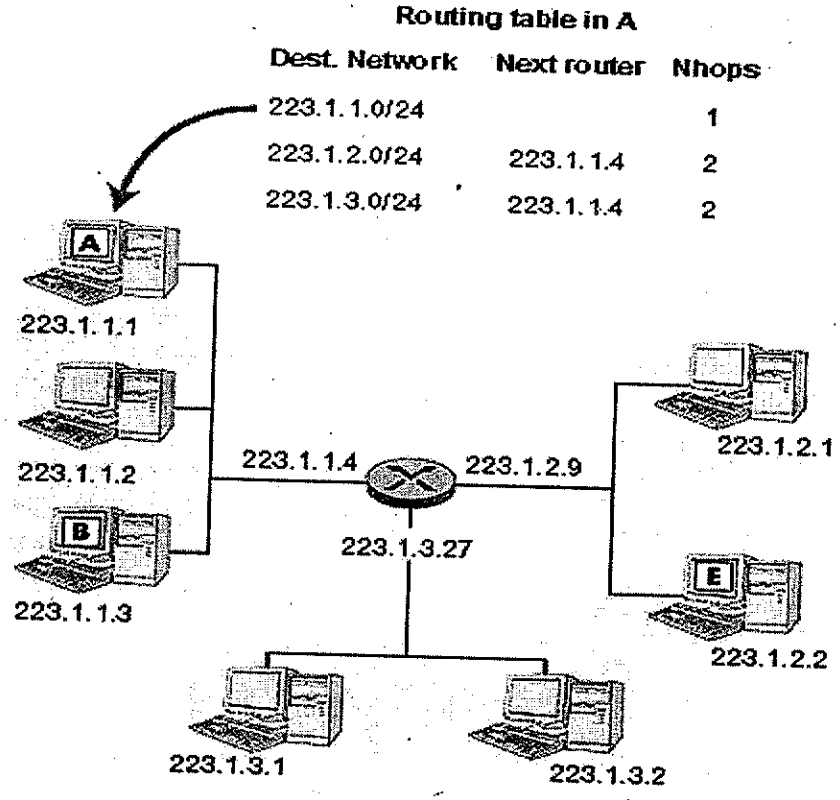
Hình 4.16

4.4.2 Chuyển datagram từ nguồn tới đích: vấn đề địa chỉ và định tuyến

Giờ đây khi đã có định nghĩa về giao diện, mạng cũng như các hiểu biết cơ bản về địa chỉ IP, chúng ta hãy thử xem máy tính và các router chuyển một gói tin IP (IP datagram) từ nơi gửi đến nơi nhận như thế nào. Để đơn giản, ta coi khuôn dạng chung của gói tin IP giống như trong Hình 4.17. Mỗi IP datagram có trường địa chỉ gửi và trường địa chỉ nhận. Máy tính gửi sẽ điền vào trường địa chỉ gửi 32 bit địa chỉ IP của mình và điền vào trường địa chỉ nhận 32 bit địa chỉ IP của máy tính nhận (giống như các trường FROM và TO trên phong bì thư). Trường dữ liệu của datagram thường là gói TCP hoặc UDP segement. Khuôn dạng gói dữ liệu IP được thảo luận chi tiết trong các phần sau.

Misc fields	Source IP Address	Destination IP address	Data

Hình 4.17 Khuôn dạng chung gói dữ liệu tầng mạng

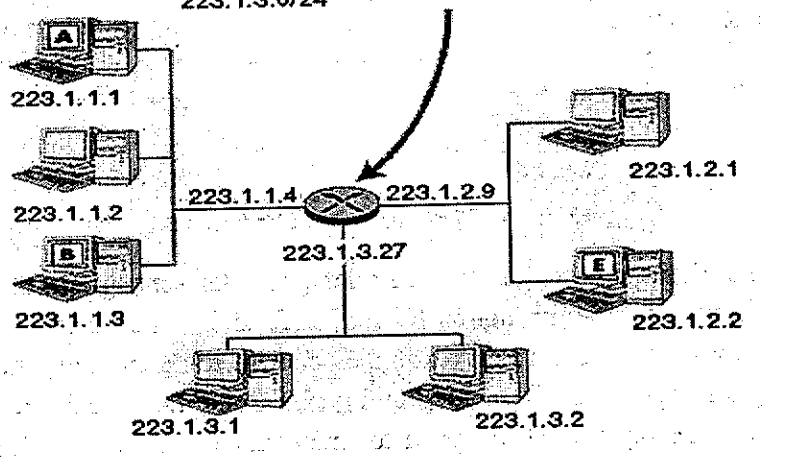


Hình 4.18 Ví dụ định tuyến từ A đến B

Sau khi máy tính gửi tạo ra IP datagram, tầng mạng làm thế nào để gửi datagram từ máy tính nguồn tới máy tính đích? Câu trả lời phụ thuộc vào việc máy tính nguồn và máy tính đích có nằm trong cùng một mạng hay không (thuật ngữ "mạng" sử dụng giống như trong 4.4.1). Đầu tiên giả sử máy tính A muốn gửi IP datagram tới máy tính B nằm trong cùng mạng 223.1.1.0/24 với A. Điều này được thực hiện như sau: đầu tiên thực thể IP trong máy A dò trong bảng định tuyến cục bộ của mình (xem Hình 4.18) và tìm thấy hàng 223.1.1.0/24 trùng với các bit đầu (địa chỉ mạng) trong địa chỉ IP của máy B. Bảng định tuyến chỉ ra rằng số lượng các thiết bị trung gian để đến mạng 223.1.1.0 là 1, nghĩa là B nằm trong cùng một mạng với A. Do đó máy A có thể gửi trực tiếp đến B mà không cần qua các router trung gian. Sau đó máy A chuyển gói dữ liệu IP cho tầng liên kết dữ liệu để chuyển gói dữ liệu đó tới B (công việc của tầng liên kết dữ liệu chuyển dữ liệu giữa hai thiết bị nối trực tiếp với nhau sẽ được trình bày trong chương 5).

Xét tiếp trường hợp máy A gửi gói dữ liệu tới máy E nằm trên mạng khác. A dò trên bảng định tuyến của mình và thấy 223.1.1.0/24 có địa chỉ mạng trùng với phần mạng trong địa chỉ IP của E. Vì số lượng các thiết bị trung gian là 2, nên máy A biết máy đích nằm trên mạng khác và do đó sẽ phải chuyển qua router trung gian. Ngoài ra bảng định tuyến tại A cũng cho biết để gửi tới E, đầu tiên A phải gửi gói dữ liệu tới địa chỉ IP 223.1.1.4 – là địa chỉ IP của giao diện router nằm trên cùng một mạng với A. Thực thể IP trong máy A chuyển gói dữ liệu xuống tầng liên kết dữ liệu và yêu cầu chuyển tới địa chỉ IP 223.1.1.4. Một chú ý rất quan trọng ở đây là mặc dù gói dữ liệu IP được gửi tới giao diện của router (qua tầng liên kết dữ liệu), địa chỉ đích trong gói dữ liệu vẫn là địa chỉ đích cuối cùng (địa chỉ E) chứ không phải là địa chỉ router trung gian.

Dest. Network	Next router	Nhops	Interface
223.1.1.0/24	–	1	223.1.1.4
223.1.2.0/24	–	1	223.1.2.9
223.1.3.0/24	–	1	223.1.3.27



Hình 4.19 Chuyển dữ liệu từ A đến E

Khi gói dữ liệu tới router thì trách nhiệm chính của router là chuyể gói dữ liệu hướng tới đích cuối cùng. Như minh họa trong Hình 4.19, rout tìm trên bảng định tuyến và thấy hàng 223.1.2.0/24 trùng với phần mạng đ chỉ IP của E. Như vậy gói dữ liệu phải được chuyển đến giao diện của rout có địa chỉ IP là 223.1.2.9. Vì số lượng các thiết bị trung gian giữa router v đích là 1, nên router biết rằng đích (E) nằm trên cùng một mạng với gi diện ứng với địa chỉ 223.1.2.9, vì thế router chuyển gói dữ liệu tới giao d (cổng) này, và sau đó gói dữ liệu được chuyển trực tiếp tới E.

Chú ý rằng trong Hình 4.19, các hàng trong cột "next router" là rỗng vì mỗi mạng (223.1.1.0/24, 223.1.2.0/24, và 223.1.3.0/24) được kết nối trực tiếp với router. Trong trường hợp này, chúng không cần phải đi qua các router trung gian để đến đích. Tuy nhiên nếu máy A và máy E cách nhau 2 router thì trong trong bảng định tuyến của router đầu tiên trên tuyến đường từ A tới E, dòng tương ứng với đích E sẽ chỉ ra phải qua 2 chặng nữa mới tới được đích và phải chỉ ra địa chỉ IP của router thứ hai trên tuyến đường. Router đầu tiên sẽ chuyển gói dữ liệu tới router thứ hai nhờ vào giao thức của tầng liên kết dữ liệu giữa hai router. Kế đó router thứ hai sẽ chuyển gói dữ liệu tới máy đích nhờ giao thức tầng liên kết dữ liệu giữa router thứ hai và máy đích.

Định tuyến cho gói dữ liệu trên mạng Internet tương tự việc người lái xe hỏi đường cảnh sát giao thông tại mỗi bùng binh. Trên đường di chuyển từ nguồn tới đích, gói dữ liệu sẽ đi qua nhiều router. Tại mỗi router, gói tin dừng lại và "hỏi" router đi đường nào để tới đích. Trừ khi router trên cùng mạng với máy tính đích, về cơ bản, bảng định tuyến trong router sẽ nói với gói dữ liệu: "Tôi không biết chính xác đi đến đích như thế nào, nhưng tôi biết hướng tới nó là đi theo đường này". Sau đó gói dữ liệu sẽ được gửi đi trên đường kết nối này để đến một router khác và lại hỏi tiếp tục hỏi đường.

Rõ ràng bảng định tuyến của router đóng vai trò then chốt trong việc định tuyến gói tin qua mạng Internet. Nhưng làm thế nào để cấu hình và bảo trì bảng định tuyến trong một hệ thống liên mạng cực lớn với nhiều tuyến đường giữa đích và nguồn (như trong mạng Internet). Rõ ràng bảng định tuyến phải được cấu hình sao cho các gói dữ liệu được đi theo tuyến đường tốt nhất từ nguồn tới đích. Chính các thuật toán định tuyến đã nói tới trong mục 4.2 thực hiện công việc định tuyến và cập nhật bảng định tuyến. Thuật toán định tuyến của Internet được trình bày trong mục 4.5.

4.3 Khuôn dạng gói dữ liệu IP

Sau đây là các trường trong gói dữ liệu IPv4:

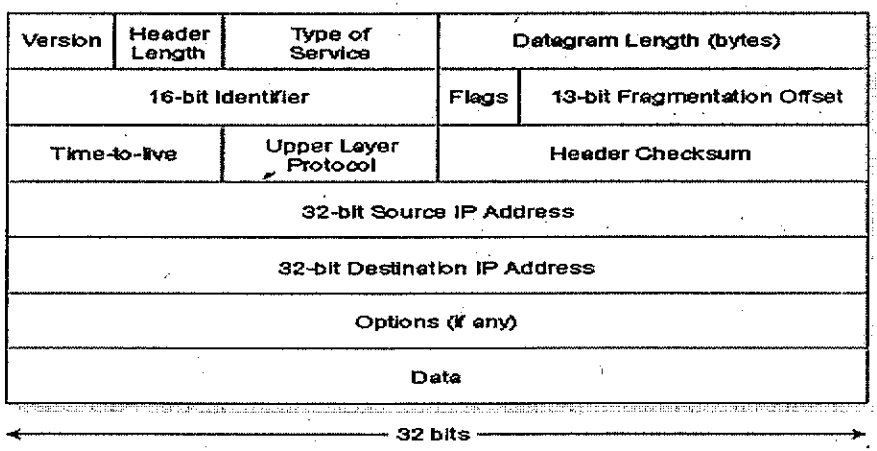
Phiên bản (version): Trường 4 bit này xác định phiên bản giao thức của gói dữ liệu. Qua trường phiên bản, router mới xác định được ý nghĩa các trường còn lại của gói dữ liệu IP. Các phiên bản IP khác nhau sử dụng các khuôn dạng dữ liệu khác nhau. Khuôn dạng gói dữ liệu IP hiện tại - IPv4

- được minh họa trong Hình 4.20. Khuôn dạng gói dữ liệu IPv6 được trình bày trong phần 4.7.

Độ dài tiêu đề (Header length): Gói dữ liệu IPv4 có thể có nhiều trường mang tính lựa chọn (không bắt buộc phải có). 4 bit này được dùng để xác định vị trí bắt đầu của dữ liệu thực sự trong gói dữ liệu IP. Tuy nhiên phần lớn gói dữ liệu IP không chứa các trường lựa chọn nên tiêu đề của gói dữ liệu thường cố định là 20 byte.

Kiểu dịch vụ (Type of service – TOS): Trường kiểu dịch vụ (TOS) giúp phân biệt các kiểu khác nhau của gói dữ liệu IP, để từ đó có thể xử lý theo những cách khác nhau. Ví dụ khi mạng quá tải, cần phân biệt được gói dữ liệu chứa thông tin kiểm soát mạng (ICMP) với gói dữ liệu thực sự (thông điệp HTML) hay giữa datagram chứa dữ liệu thời gian thực (ứng dụng điện thoại qua Internet) với datagram không chứa dữ liệu thời gian thực (ứng dụng FTP). Gần đây Cisco (công ty chiếm thị phần router lớn nhất) đã sử dụng 3 bit đầu tiên của trường TOS để định nghĩa các mức dịch vụ khác nhau mà router có thể cung cấp. Các mức dịch vụ cụ thể được người quản trị router thiết lập theo những tiêu chí của tổ chức.

Độ dài gói dữ liệu (datagram length): đây là tổng độ dài của gói dữ liệu IP (cả phần tiêu đề lẫn phần dữ liệu) tính theo byte. Độ dài trường này là 16 bit nên về lý thuyết kích thước tối đa của gói dữ liệu IP là 65.535 byte. Tuy nhiên, hiếm khi kích thước gói dữ liệu vượt quá 1500 byte và thường giới hạn là 576 byte.



Hình 4.20 Khuôn dạng gói dữ liệu IP

Định danh, cờ và vị trí phân đoạn (Identifier, Flags, Fragmention Offset). 3 trường này được sử dụng khi phân mảnh gói IP (fragmentation), một chủ đề chúng ta sẽ xem xét chi tiết dưới đây. Chú ý phiên bản mới của IP (IPv6) không cho phép phân mảnh gói dữ liệu tại các router.

Thời gian tồn tại (Time-to-live-TTL): Trường thời gian tồn tại (TTL) được sử dụng để bảo đảm gói dữ liệu không thể lưu chuyển mãi mãi trong mạng (nguyên nhân có thể do định tuyến lặp nên các gói tin truyền lòng vòng theo một chu trình). Trường này bị giảm đi một (-1) mỗi lần gói tin đi qua router. Nếu trường TTL bằng 0, router sẽ loại bỏ gói tin.

Giao thức (Protocol): Trường này chỉ được sử dụng khi gói dữ liệu IP đến được máy tính đích. Giá trị của trường này xác định giao thức tầng giao vận ở máy tính đích sẽ nhận được phần dữ liệu trong gói dữ liệu IP. Ví dụ giá trị 6 có ý nghĩa phần dữ liệu cần chuyển tới thực thể TCP, giá trị 17 có ý nghĩa phần dữ liệu phải chuyển đến thực thể UDP. RFC 1700 liệt kê các giá trị này. Chú ý rằng vai trò của trường giao thức trong gói dữ liệu IP tương tự vai trò trường số hiệu cổng trong segment của tầng giao vận. Trường giao thức được xem là điểm nối giữa tầng mạng và tầng giao vận cũng như trường số hiệu cổng là điểm nối giữa tầng giao vận với ứng dụng cụ thể. Trong chương 5 chúng ta cũng sẽ thấy trong frame của tầng liên kết dữ liệu cũng có một trường đặc biệt để nối với tầng mạng.

Checksum của tiêu đề (Header checksum): Trường checksum trong tiêu đề giúp router phát hiện lỗi trong tiêu đề gói dữ liệu IP được gửi đến. Giá trị checksum được tính bằng cách xem phần tiêu đề là một chuỗi các từ hai byte, cộng các từ này lại và sau đó lấy bù một. Như đã thảo luận trong phần 3.3, số bù một của tổng này được gọi là Internet checksum. Router tính lại Internet checksum cho mỗi gói dữ liệu IP nhận được và có thể phát hiện ra lỗi nếu như giá trị checksum tính lại khác giá trị checksum trong gói dữ liệu. Router thường loại bỏ những gói dữ liệu bị lỗi. Chú ý rằng router phải tính lại checksum, trường TTL và có thể một số trường khác. RFC 1071 trình bày phương thức tính checksum nhanh. Một vấn đề thường được đặt ra ở đây là tại sao TCP/IP thực hiện kiểm tra lỗi ở cả tầng giao vận lẫn tầng mạng? Có nhiều nguyên nhân của việc này. Thứ nhất: các router không bắt buộc phải kiểm tra lỗi, vì vậy tầng giao vận không thể dựa vào tầng mạng để làm việc này. Thứ hai: TCP/UDP và IP không nhất thiết nằm trong cùng một nhóm giao thức. TCP có thể chạy trên giao thức khác (ví dụ ATM) và IP có thể chuyển dữ liệu không phải là TCP hay UDP segment.

Địa chỉ IP nguồn và đích: Những trường này là 32 bit địa chỉ IP của máy tính gửi và máy tính nhận. Tầm quan trọng của địa chỉ đích là rõ ràng. Trong phần 3.2 chúng ta thấy rằng địa chỉ IP máy gửi (cùng với số hiệu cổng nguồn và đích) được máy nhận sử dụng để hướng dữ liệu ứng dụng tới socket phù hợp.

Lựa chọn (Option): Các trường này cho phép mở rộng tiêu đề IP. Phần lựa chọn trong tiêu đề hiếm khi được sử dụng. Sự tồn tại của phần lựa chọn trong tiêu đề làm phức tạp việc xử lý các gói tin vì tiêu đề của gói dữ liệu có phần lựa chọn không có độ dài cố định, do đó không xác định được vị trí bắt đầu của dữ liệu thực sự. Như vậy thời gian xử lý gói dữ liệu IP tại mỗi router có thể khác nhau. Đây là nhược điểm của các mạng hiệu suất cao. Vì thế, IPv6 sẽ loại bỏ các trường lựa chọn.

Dữ liệu (payload): Cuối cùng là trường quan trọng nhất - trường dữ liệu. Thông thường trường dữ liệu của gói IP là gói dữ liệu của tầng giao vận (TCP hay UDP segment). Tuy nhiên, trường dữ liệu có thể là các kiểu dữ liệu khác, ví dụ thông điệp ICMP (sẽ được trình bày trong phần 4.4.5).

4.4.4 Phân mảnh (Fragmentation) và Hợp nhất (Reassembly) gói tin IP

Trong chương 5 chúng ta sẽ thấy không phải tất cả các giao thức của tầng liên kết dữ liệu đều có khả năng truyền các gói tin (packet) có cùng độ lớn. Một vài giao thức có khả năng gửi những gói tin lớn trong khi một vài giao thức chỉ có thể gửi những gói tin nhỏ. Ví dụ, gói tin của mạng Ethernet có độ lớn lên tới 1500 byte, trong khi gói tin trên những liên kết ở mạng diện rộng có độ lớn không được vượt quá 576 byte. Số lượng dữ liệu tối đa của gói tin trên một đường truyền vật lý được định nghĩa là **MTU (maximum transfer unit)**. Gói dữ liệu IP được đặt trong gói dữ liệu của tầng liên kết dữ liệu giữa hai router kế tiếp nhau trên đường truyền. Vì thế giá trị MTU của giao thức ở tầng liên kết dữ liệu giới hạn độ dài gói tin IP. Giới hạn kích thước gói tin IP không phải là vấn đề lớn. Vấn đề ở đây là kết nối giữa các router dọc theo tuyến đường từ nơi gửi đến nơi nhận có thể sử dụng các giao thức liên kết dữ liệu khác nhau có giá trị MTU khác nhau.

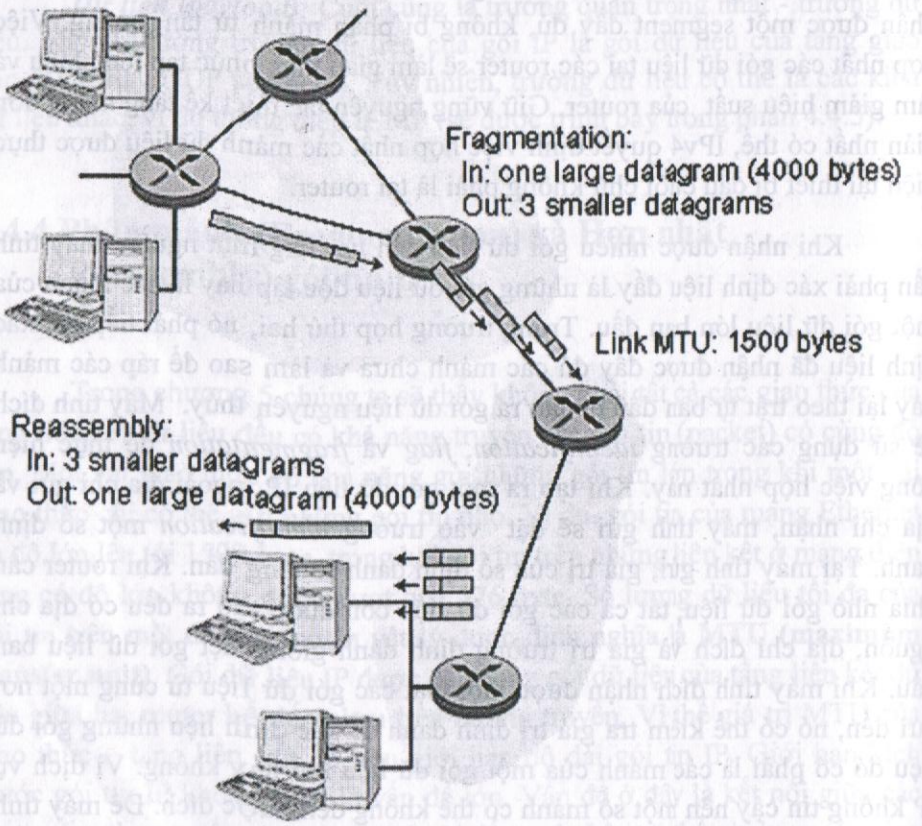
Để hiểu vấn đề rõ hơn, xét router với nhiều kết nối, mỗi kết nối có một giao thức liên kết dữ liệu khác nhau với giá trị MTU khác nhau. Giả sử khi nhận được gói dữ liệu đến từ kết nối nào đó, căn cứ vào địa chỉ đích, router kiểm tra bảng định tuyến để xác định cần gửi gói tin đi ra theo kết nối nào. Tuy nhiên đường kết nối ra ngoài này có giá trị MTU nhỏ hơn độ dài gói dữ liệu IP. Làm thế nào router có thể đặt gói tin IP lớn trong gói tin của tầng liên kết dữ liệu có kích thước nhỏ hơn? Giải pháp cho vấn đề này là phân mảnh (fragmentation) 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 (fragment).

Các mảnh tách rời này cần được ráp lại trước khi chuyển lên tầng giao vận tại máy tính nhận. Rõ ràng là cả TCP và UDP đều mong muốn nhận được một segment đầy đủ, không bị phân mảnh từ tầng mạng. Việc hợp nhất các gói dữ liệu tại các router sẽ làm giao thức phức tạp lên nhiều và làm giảm hiệu suất của router. Giữ vững nguyên tắc thiết kế tầng mạng đơn giản nhất có thể, IPv4 quyết định việc hợp nhất các mảnh dữ liệu được thực hiện tại thiết bị đầu cuối chứ không phải là tại router.

Khi nhận được nhiều gói dữ liệu đến từ cùng một nguồn, máy tính cần phải xác định liệu đấy là những gói dữ liệu độc lập hay là các mảnh của một gói dữ liệu lớn ban đầu. Trong trường hợp thứ hai, nó phải tiếp tục xác định liệu đã nhận được đầy đủ các mảnh chưa và làm sao để ráp các mảnh này lại theo trật tự ban đầu để tạo ra gói dữ liệu nguyên thủy. Máy tính đích sẽ sử dụng các trường *identification*, *flag* và *fragmentation* để thực hiện công việc hợp nhất này. Khi tạo ra một gói dữ liệu IP, ngoài địa chỉ gửi và địa chỉ nhận, máy tính gửi sẽ đặt vào trường *identification* một số định danh. Tại máy tính gửi, giá trị của số định danh sẽ tăng dần. Khi router cần chia nhỏ gói dữ liệu, 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. Khi máy tính 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 phải là các mảnh của một gói dữ liệu lớn hay không. Vì dịch vụ IP không tin cậy nên một số mảnh có thể không đến được đích. Để máy tính nhận có thể chắc chắn là đã nhận được mảnh cuối cùng của gói dữ liệu ban đầu, thì trường cờ của mảnh cuối cùng phải có giá trị 0, trong khi trường cờ

của các mảnh khác có giá trị 1. Tương tự để máy nhận xác định được liệu có mất mảnh nào không (và để ghép các mảnh theo đúng thứ tự), trường Offset được sử dụng để xác định vị trí của mảnh trong gói dữ liệu IP ban đầu.

Xét ví dụ trên Hình 4.21. Một gói dữ liệu có độ lớn 4000 byte đến router và phải gửi qua đường liên kết có MTU là 1500 byte. Điều này có nghĩa rằng 3980 byte dữ liệu trong gói dữ liệu ban đầu phải được tách ra thành ba mảnh phân biệt (mỗi mảnh trở thành một gói dữ liệu IP độc lập). Giả sử trong gói dữ liệu ban đầu giá trị trường định danh là 777. Giá trị các trường trong ba phân mảnh này được chỉ ra trong Bảng 4.3



Hình 4.21 Phân mảnh và hợp nhất gói dữ liệu IP

Dữ liệu của gói IP chỉ được chuyển lên tầng giao vận tại máy tính nhận khi tầng IP tái tạo hoàn chỉnh gói dữ liệu IP ban đầu. Nếu một số mảnh dữ liệu bị mất, không đến được đích, thì toàn bộ gói dữ liệu sẽ bị loại bỏ và không được chuyển lên tầng giao vận. Nhưng như đã nghiên cứu trong chương trước, nếu sử dụng TCP ở tầng giao vận, thì thực thể TCP sẽ khắc phục mất mát do phía gửi sẽ gửi lại gói dữ liệu ban đầu.

Fragment	Bytes	ID	Offset	Flag
1	1480 byte trong trường dữ liệu	identification=777	offset=0 (dữ liệu bắt đầu từ byte thứ 0)	flag=1 (còn mảnh nữa)
2	1480 byte trong trường dữ liệu	identification=777	offset=1480 (dữ liệu bắt đầu từ byte thứ 1480)	flag=1 (còn mảnh nữa)
3	1020 byte (=3980 - 1480 - 1480) trong trường dữ liệu	identification=777	offset=2960 (dữ liệu bắt đầu từ byte thứ 2960)	flag=0 (đây là mảnh cuối cùng)

Bảng 4.3 Ví dụ phân mảnh gói tin

Phân mảnh và hợp nhất khiến nhiệm vụ xử lý gói tin tại router (tạo ra các mảnh) và thiết bị nhận (hợp nhất các mảnh) phức tạp hơn. Vì thế người ta cố gắng giảm thiểu việc phân mảnh dữ liệu. Điều này thường được thực hiện bằng cách giới hạn độ lớn gói dữ liệu của tầng giao vận (TCP hay UDP segment) bởi một giá trị tương đối nhỏ. Khi đó việc phân mảnh trở nên không cần thiết. Vì phần lớn các giao thức liên kết dữ liệu hỗ trợ IP có MTU tối thiểu là 536 byte, có thể loại bỏ hoàn toàn việc phân mảnh nếu đặt giá trị MSS là 536 byte với 20 byte tiêu đề của gói TCP và 20 byte tiêu đề của gói IP. Đây chính là lý do hầu hết các gói TCP khi truyền khối lượng lớn dữ liệu (chẳng hạn FTP) có độ dài từ 512 đến 536 byte (Khi duyệt WEB, bạn sẽ thấy thường khoảng 500 byte dữ liệu đến cùng một lần).

Nếu gói TCP được lồng trong gói IP và cả hai gói TCP và IP đều không có trường tùy chọn (option) thì gói dữ liệu IP sẽ có 40 byte tiêu đề, phần còn lại là dữ liệu ứng dụng.

4.4.5 Giao thức kiểm soát lỗi ICMP (Internet Control Message Protocol)

ICMP được các máy tính đầu cuối, router và các cổng (gateway) sử dụng để trao đổi các thông tin tầng mạng với nhau. ICMP được đặc tả trong RFC 792. ICMP được sử dụng chủ yếu cho việc báo lỗi. Ví dụ khi chạy một phiên Telnet, FTP, hoặc HTTP, bạn có thể gặp một thông điệp như "Destination network unreachable" (*Không đến được mạng đích).* Thông điệp này có do thực thể ICMP ở router tạo ra. Khi không tìm được đường dẫn đến máy tính đích, router sẽ tạo ra và gửi thông báo ICMP kiểu 3 tới máy tính của bạn với mục đích thông báo lỗi. Máy tính nhận được thông báo lỗi ICMP sẽ trả lại mã lỗi cho thực thể TCP đang cố gắng kết nối tới máy tính đích. Đến lượt lượt mình, TCP trả lại mã lỗi cho ứng dụng (là phiên làm việc FTP, HTTP...).

ICMP thường được coi là một phần của IP, nhưng về mặt kiến trúc lại nằm trên IP, bởi vì thông báo ICMP được đặt trong gói IP, giống như TCP hay UDP segment nằm trong trường dữ liệu (payload) của gói dữ liệu IP. Tương tự, khi nhận được một gói tin IP với trường *protocol* xác định giao thức ICMP, tầng mạng của máy tính nhận sẽ chuyển phần dữ liệu (là thông điệp ICMP) lên thực thể ICMP, giống như đã làm với TCP hay UDP.

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). Do đó phía gửi có thể xác định được gói tin nào gây ra lỗi. Một số kiểu thông điệp ICMP tiêu biểu được minh họa trên Hình 4.22. Chú ý rằng thông báo ICMP không chỉ được sử dụng để báo lỗi. Chương trình ping rất thông dụng gửi thông báo ICMP kiểu 8, mã 0 tới máy tính nào đó, máy tính nhận được yêu cầu ICMP sẽ gửi lại một thông báo ICMP phản hồi với kiểu 0, mã 0.

ICMP Type	Code	Description
0	0	echo reply (to ping)
3	0	destination network unreachable
3	1	destination host unreachable
3	2	destination protocol unreachable
3	3	destination port unreachable
3	6	destination network unknown
3	7	destination host unknown
4	0	source quench (congestion control)
8	0	echo request
9	0	router advertisement
10	0	router discovery
11	0	TTL expired
12	0	IP header bad

Hình 4.22 Các kiểu thông điệp ICMP hay gặp

Chương trình ***Traceroute*** cho phép bạn xác định tất cả các router trên một tuyến đường giữa bất kỳ hai thiết bị đầu cuối nào. Chương trình Traceroute cũng sử dụng các thông báo của ICMP để xác định tên và địa chỉ của các router giữa nguồn và đích. Chương trình Traceroute trong máy tính nguồn sẽ gửi đi một loạt các gói dữ liệu IP tới máy tính đích. Gói IP đầu tiên có trường TTL nhận giá trị 1, gói thứ hai là 2, gói thứ ba là 3,... Máy tính nguồn đặt timer cho mỗi gói IP gửi đi. Khi gói IP thứ n đến router thứ n, router này thấy trường TTL của gói dữ liệu nhận giá trị 0, nên theo nguyên tắc của giao thức IP, router sẽ loại bỏ gói dữ liệu và gửi thông điệp cảnh báo ICMP (kiểu 11 mã 0). Trong thông điệp cảnh báo này có tên và địa chỉ IP của router. Khi nhận được thông báo ICMP, máy tính nguồn xác định được thời gian khứ hồi đến router thứ n (nhờ timer) cũng như tên, và địa chỉ IP của router đó.

4.5 ĐỊNH TUYẾN TRÊN INTERNET

Sau khi đã nghiên cứu về địa chỉ Internet và giao thức IP, bây giờ chúng ta nói tới các giao thức định tuyến của Internet – là các giao thức xác

định tuyến đường đi từ nguồn tới đích. Chúng ta sẽ thấy rằng các giao thức định tuyến của Internet được triển khai dựa trên những nguyên tắc mà chúng ta đã nói tới: link state và distance vector trong phần 4.2, miền tự trị (AS) trong phần 4.3.

Mạng Internet toàn cầu ngày nay là sự kết hợp lỏng lẻo của nhiều mạng bao gồm các ISP khu vực, quốc gia và quốc tế. Trong phần 4.3 chúng ta thấy rằng tập hợp các router cùng nằm dưới một một sự quản trị - ít nhất về mặt kỹ thuật - tạo thành miền tự trị (AS). Mỗi AS lại có thể bao gồm nhiều mạng (ở đây chúng ta sẽ sử dụng thuật ngữ "mạng" với ý nghĩa giống trong phần 4.4). Điểm phân biệt quan trọng nhất giữa các giao thức định tuyến của Internet là liệu chúng được sử dụng để định tuyến trong một miền hay giữa các miền với nhau. Phần 4.5.1 giới thiệu giao thức định tuyến trong một miền và phần 4.5.2 giới thiệu các giao thức định tuyến giữa các miền.

4.5.1 Định tuyến trong một miền (Intra-AS routing) (Định tuyến nội miền)

Giao thức định tuyến Intra-AS được sử dụng để cấu hình và duy trì bảng định tuyến trong tất cả các router thuộc cùng một miền. Những giao thức định tuyến kiểu này được gọi là **giao thức định tuyến nội miền (interior gateway protocol)**. Trên Internet có 3 giao thức định tuyến nội miền được sử dụng rộng rãi: **RIP** (Routing Information Protocol), **OSPF** (Open Shortest Path First) và **EIGRP** (Cisco's propriety Enhanced Interior Gateway Routing Protocol).

RIP (Routing Information Protocol)

RIP là một trong những giao thức định tuyến nội miền đầu tiên. Nó được triển khai trong một chương trình được gọi là ***routed*** trong phần lớn các hệ thống UNIX. RIP có một số đặc điểm sau:

Định tuyến nội miền: Cho phép các router trong một miền trao đổi thông tin với nhau.

Đo khoảng cách bằng chặng: Giá đường đi giữa hai thiết bị đầu cuối được xác định bằng số lượng các router trung gian trên đường đi đó. Độ

dài tối đa của một tuyến đường là 15, nghĩa là đường kính tối đa của một miền là 15 router.

Truyền thông không tin cậy: RIP sử dụng UDP để chuyển thông điệp.

Gửi quảng bá (broadcast) và multicast: RIP được sử dụng chủ yếu trên mạng cục bộ (LAN) hỗ trợ công nghệ truyền quảng bá (mạng Ethernet). RIP v1 sử dụng cách truyền quảng bá khi truyền giữa hai router. RIP v2 cho phép truyền theo chế độ multicast.

Thuật toán distance vector. RIP sử dụng thuật toán distance vector. Các router hàng xóm trao đổi bảng định tuyến cho nhau 30s một lần trong các thông điệp RIP (RIP response message, RIP advertisement), mỗi thông điệp chứa tối đa 25 địa chỉ đích tới.

Các máy tính có thể thụ động nhận thông tin từ các router. RIP cho phép các thiết bị đầu cuối (chủ yếu là máy tính) lắng nghe và cập nhật bảng định tuyến. Điều này đặc biệt hữu dụng với các mạng có nhiều router. Khi đó máy tính trong mạng có thể dễ dàng xác định được router cần chuyển tới.

Chú ý rằng router gửi một thông điệp RIP liệt kê các mạng mà nó có thể kết nối tới. Khi nhận được một quảng cáo như vậy, thực thể RIP (phần mềm) trên router sử dụng những thông tin này để cập nhật lại bảng định tuyến của mình. Mỗi một trường trong thông điệp quảng cáo là một cặp:

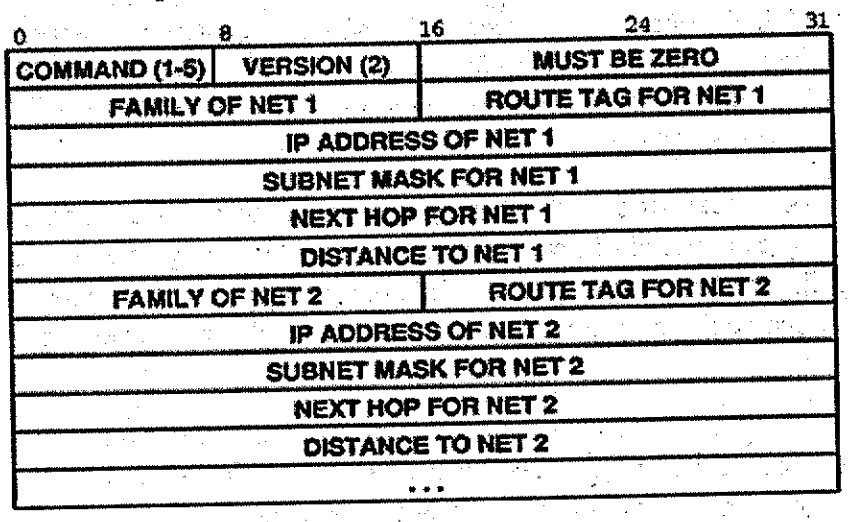
(địa chỉ mạng đích **n**, khoảng cách **r**)

trong đó khoảng cách **r** là số lượng các router trung gian từ router gửi thông điệp tới đích có địa chỉ mạng là **n**. Khi nhận được một thông điệp, giả sử router nhận không có đường đi tới đích được quảng cáo trong thông điệp hoặc có đường đi đến đích nhưng giá lớn hơn, router sẽ cập nhật bảng định tuyến để sử dụng tuyến đường vừa mới nhận được quảng cáo (điểm đầu tiên trên tuyến đường này chính là router gửi quảng cáo).

Ưu điểm chính của RIP là tính đơn giản. RIP không đòi hỏi cấu hình nhiều. Người quản trị chỉ cần bật máy lên, cho phép router trao đổi thông tin với nhau, sau một thời gian ngắn, router sẽ tự xây dựng được bảng định tuyến cho mình.

Tổ chức có thể lựa chọn một router trong miền làm router ngầm định, thường là router nối với ISP. Sau đó RIP sẽ thực hiện việc quảng cáo cho router ngầm định này. Sau đó các gói tin gửi ra phía ngoài sẽ được gửi qua router ngầm định tới ISP.

Hình 4.23 minh họa khuôn dạng thông điệp cập nhật RIP. Mỗi trường trong thông điệp ứng với một địa chỉ đích, mặt nạ mạng của địa chỉ đích (do đó có thể sử dụng địa chỉ không phân lớp CIDR), khoảng cách tới đích và nút kế tiếp trên đường tới đích.



Hình 4.23 Khuôn dạng gói dữ liệu RIP

OSPF – Open Shartest Path First

RIP có một số nhược điểm của thuật toán distance vector. Độ dài của thông điệp có thể lớn do phải liệt kê toàn bộ danh sách các địa chỉ đích và khoảng cách tới đó. Khi nhận được thông điệp, router nhận phải lấy ra từng trường, so sánh trong bảng định tuyến, như vậy thời gian xử lý thông điệp trong mỗi router lớn, gây ra một độ trễ nào đó. Do vậy RIP chỉ phù hợp với các mạng có kích cỡ nhỏ.

Khi một tổ chức mạng tương đối lớn, người ta cần phải đưa ra giao thức phù hợp hơn. IETF đưa ra OSPF với các đặc điểm sau:

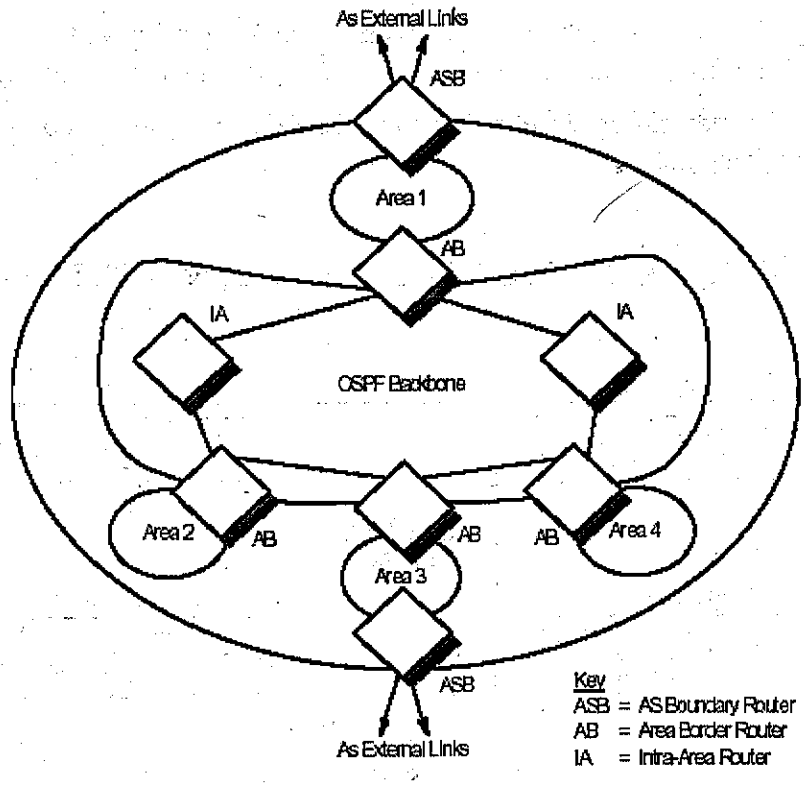
Định tuyến nội miền: Cho phép trao đổi thông tin giữa các router trong một miền.

Hỗ trợ phân mạng và CIDR. Bên cạnh địa chỉ IP 32 bit là mặt nạ 32 bit. Do đó OSPF hỗ trợ việc phân mạng, chia mạng to ra các mạng con.

Trao đổi các thông tin đã được kiểm chứng. Hai router trao đổi thông điệp OSPF với nhau có thể tiến hành thủ tục kiểm tra để xác định mình nhận được thông điệp từ đúng phía bên kia. Điều này ngăn ngừa được tin tặc tiến hành các cuộc tấn công bằng phương pháp giả mạo.

Sử dụng thuật toán Link state.

Hỗ trợ phân cấp trong miền. Ưu điểm chính của OSPF là cho phép tiếp tục phân một miền thành nhiều miền con.



Hình 4.24 OSPF cho phép chia miền con

4.5.2 Định tuyến giữa các miền (Inter-AS routing) (Định tuyến liên miền)

Giao thức BGP (Border Gateway Protocol v4) (xem đặc tả RFC 1771, 1772, 1773) được xem là một chuẩn ngầm định *de facto* trong định tuyến liên miền trên Internet ngày nay. Nhiệm vụ của nó là định tuyến giữa các miền được quản trị độc lập với nhau.

BGP (Border Gateway Protocol)

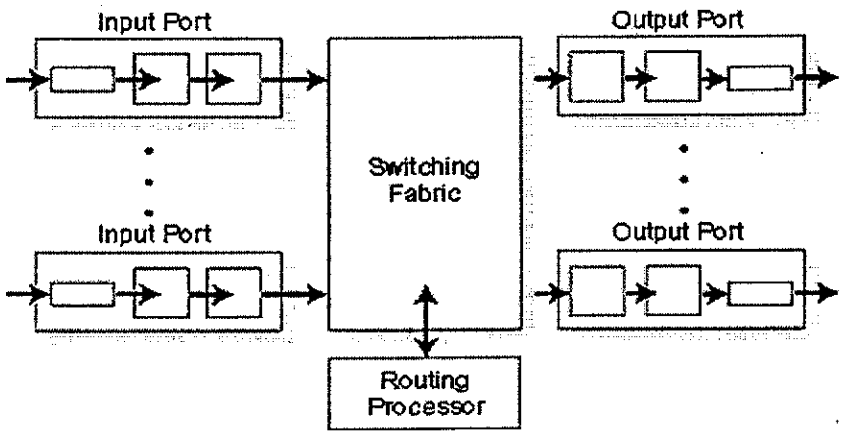
BGP là giao thức liên miền chủ yếu được sử dụng hiện nay. BGP có những đặc điểm sau:

Định tuyến liên miền. BGP cho phép cung cấp các thông tin định tuyến giữa các miền. Mỗi tuyến đường được xem là một chuỗi các AS liên tiếp nhau.

Hỗ trợ việc thiết lập chính sách (policy). Người quản trị có thể áp dụng những chính sách nào đó, ví dụ hạn chế việc quảng cáo ra phía ngoài.

Truyền thông tin cậy. Hai thực thể BGP sử dụng kết nối TCP để trao đổi thông điệp.

4.6 CẤU TẠO CỦA THIẾT BỊ ĐỊNH TUYẾN (ROUTER)



Hình 4.25 Kiến trúc bộ định tuyến

Các phần trước trình bầy về các mô hình dịch vụ của tầng mạng, các thuật toán định tuyến xác định đường đi cho các gói tin trên mạng cũng như các giao thức gắn với các thuật toán ấy. Trong phần này, chúng ta sẽ nói đến một chủ đề quan trọng khác - chức năng chuyển mạch của bộ định tuyến – công việc thực sự để chuyển một datagram từ liên kết này tới liên kết kia. Chỉ nghiên cứu các khía cạnh về mặt kiểm soát và dịch vụ của tầng mạng cũng giống như nghiên cứu một công ty mà chỉ tìm hiểu cơ chế quản lý của công ty và các quan hệ với bên ngoài. Để hiểu rõ về công ty, người ta phải xem xét đến công nhân. Trong tầng mạng, công việc thực sự của việc truyền gói tin chính là việc chuyển gói tin từ liên kết này tới liên kết kia của router. Trong mục này chúng ta sẽ nghiên cứu router thực hiện công việc này như thế nào.

Một cách tổng thể, kiến trúc chung của router được minh họa trong Hình 4.25. Bốn thành phần chính của router được xác định như sau:

Cổng vào (Input port). Cổng vào của router thực hiện một số chức năng: chức năng của tầng vật lý (hộp ngoài cùng cổng vào và cổng ra); chức năng của tầng liên kết dữ liệu (là các hộp ở giữa đối với cả đường vào và đường ra) cần thiết để làm việc được với tầng liên kết dữ liệu ở đầu bên kia của kết nối; chức năng tìm kiếm và chuyển (hộp trong cùng của cổng vào và cổng ra). Gói tin từ cổng vào sẽ đi qua kết cấu chuyển để tới cổng ra phù hợp. Các gói tin chứa thông tin điều khiển (chứa thông tin điều khiển của các giao thức RIP, OSPF, BGP) sẽ được chuyển từ cổng vào đến bộ xử lý của router.

Kết cấu chuyển (Switching fabric). Kết cấu chuyển nối cổng vào của router tới cổng ra. Kết cấu chuyển nằm hoàn toàn trong router – là một mạng chuyển mạch nằm bên trong router mạng.

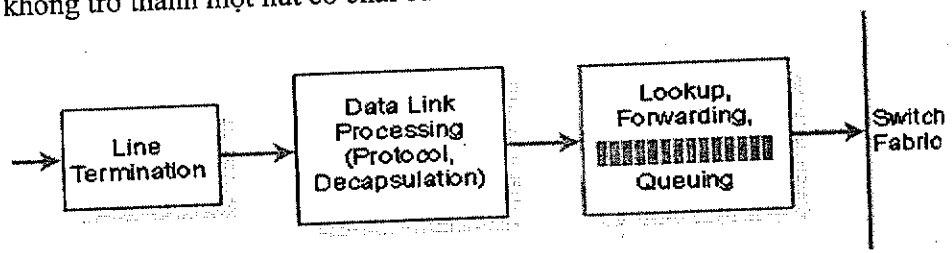
Cổng ra (output port) Cổng ra nhận những gói dữ liệu gửi tới nó qua kết cấu chuyển và sau đó truyền gói dữ liệu này trên đường nối ra ngoài. Nó cũng thực hiện chức năng của tầng liên kết dữ liệu và tầng vật lý.

Bộ xử lý (Routing Processor) Bộ xử lý router thực hiện các giao thức định tuyến, (ví dụ các giao thức đã nói trong phần 4.5), duy trì bảng định tuyến, và thực hiện một số chức năng quản trị mạng.

Dưới đây chúng ta sẽ nghiên cứu các thành phần này một cách kỹ lưỡng hơn.

4.6.1 Cổng vào (Input port)

Hình 4.26 minh họa chi tiết các chức năng của cổng vào. Như đã nói ở trên, chức năng kết thúc tín hiệu trên đường truyền (line termination) và xử lý liên kết dữ liệu ứng với tầng vật lý và tầng liên kết dữ liệu của một đường truyền vật lý thực sự với router. Chức năng tìm kiếm và chuyển tiếp của cổng vào đóng vai trò trung tâm trong việc chuyển của bộ định tuyến. Trong nhiều router, cổng vào chính là nơi xác định cổng ra cho một gói dữ liệu. Cổng ra được xác định nhờ các thông tin lưu trong bảng định tuyến. Mặc dù bảng định tuyến được bộ xử lý tạo ra, song mỗi cổng vào đều có một bảng sao chép bảng định tuyến và cập nhật khi cần thiết. Nhờ vậy, quyết định chuyển đến cổng ra nào có thể được thực hiện cục bộ ở tại cổng vào, mà không cần đến bộ xử lý trung tâm. Điều này khiến bộ xử lý sẽ không trở thành một nút cổ chai của router.



Hình 4.26 Cấu trúc cổng vào của Router

Với những router mà khả năng xử lý ở cổng vào còn hạn chế, cổng vào sẽ chuyển gói dữ liệu tới bộ xử lí. Bộ xử lý sẽ tìm kiếm trên bảng định tuyến để xác định cổng ra thích hợp. Người ta thường áp dụng giải pháp này trong trường hợp router là một trạm làm việc hay một máy tính. Khi đó, bộ xử lí của router là bộ xử lý của trạm làm việc hay của máy tính. Cổng vào sẽ là card mạng (ví dụ card Ethernet).

Với bảng định tuyến xác định trước, tìm kiếm trên bảng định tuyến tương đối đơn giản. Duyệt toàn bộ bảng định tuyến để xác định hàng nào có địa chỉ phù hợp nhất với địa chỉ đích của gói dữ liệu, trong trường hợp không tìm thấy thì sử dụng cổng mặc định (Trong phần 4.4.1, chúng ta thấy

rằng địa chỉ phù hợp nhất là địa chỉ có tiền tố mạng dài nhất trùng với phần mạng của địa chỉ đích). Tuy nhiên triển khai trong thực tế lại không đơn giản như thế. Điều phức tạp nhất là router trên các trục chính (backbone router) phải hoạt động ở tốc độ cao, có khả năng thực hiện hàng triệu phép tra cứu mỗi giây. Thực sự người ta mong muốn tốc độ xử lý ở cổng vào phải ngang với tốc độ đường truyền (line speed), có nghĩa là tốc độ xử lý phải nhỏ hơn tốc độ đến của các gói tin. Như vậy gói tin có thể được xử lý xong trước khi gói tin kế tiếp đến.

Để hoạt động ở tốc độ cao không thể sử dụng phương pháp tìm kiếm tuyến tính trên bảng định tuyến, hợp lý hơn là lưu giữ giá trị của bảng định tuyến trong cấu trúc dữ liệu dạng cây. Mỗi mức trong cây ứng với vị trí một bit trong địa chỉ đích. Để tìm kiếm một địa chỉ, bắt đầu từ "gốc" của cây. Nếu bit địa chỉ đầu tiên là 0 thì địa chỉ cần tìm nằm trong cây con trái, ngược lại nằm trong cây con phải. Tiếp tục duyệt cây con bằng cách sử dụng các bit còn lại. Bit kế tiếp bằng 0, tìm trên cây con trái, bit kế tiếp bằng 1 tìm trên cây con phải. Người ta sẽ tìm kiếm bảng định tuyến trong N bước, N là số lượng bit trong địa chỉ.

Thậm chí với N = 32 (địa chỉ IP 32 bit), tốc độ tìm kiếm bằng kỹ thuật duyệt nhị phân chưa đủ nhanh để đáp ứng yêu cầu định tuyến trên các đường trục chính của Internet. Ví dụ tốc độ truy cập bộ nhớ là 40ns. Đã có một số kỹ thuật nâng cao tốc độ này. Bộ nhớ đánh địa chỉ theo nội dung (CAM – Content Addressable Addressing) cho phép địa chỉ IP 32 bit được biểu diễn trong CAM, và các trường tương ứng với địa chỉ đó được xác định trong khoảng thời gian cố định. Dòng router CISCO 8500 [Cisco 8500 1999] có 64K CAM cho mỗi cổng vào. Kỹ thuật khác làm tăng tốc độ tra cứu là lưu giữ các địa chỉ vừa được truy cập trong bộ nhớ cache có tốc độ truy cập nhanh. Vấn đề cần quan tâm ở đây là độ lớn của cache.

Khi xác định được cổng ra, gói dữ liệu sẽ được chuyển đến qua kết cấu chuyển. Tuy nhiên, gói dữ liệu tạm thời có thể bị "phong tỏa" chưa được chuyển qua kết cấu (có thể do các gói dữ liệu khác đang sử dụng kết cấu chuyển). Một gói dữ liệu bị phong tỏa phải xếp hàng ở cổng vào và đợi được lên lịch chuyển qua kết cấu tại một thời điểm nào đó.

Trong lịch sử

Công ty Cisco: thống trị thị trường mạng

Tháng giêng năm 2000, Công ty Cisco có 23000 công nhân và trị giá 360 tỉ đôla. Cisco hiện thống trị thị trường router, và hiện nay đang nhanh chóng chiếm lĩnh thị trường điện thoại qua Internet cạnh tranh quyết liệt cùng với các công ty thiết bị điện thoại như Lucent, Alcatel, Nothern Telcom và Siemens. Cisco được khởi đầu như thế này ? Bắt đầu từ năm 1984 trong phòng tiếp khách của một căn hộ tại Silicon Valley.

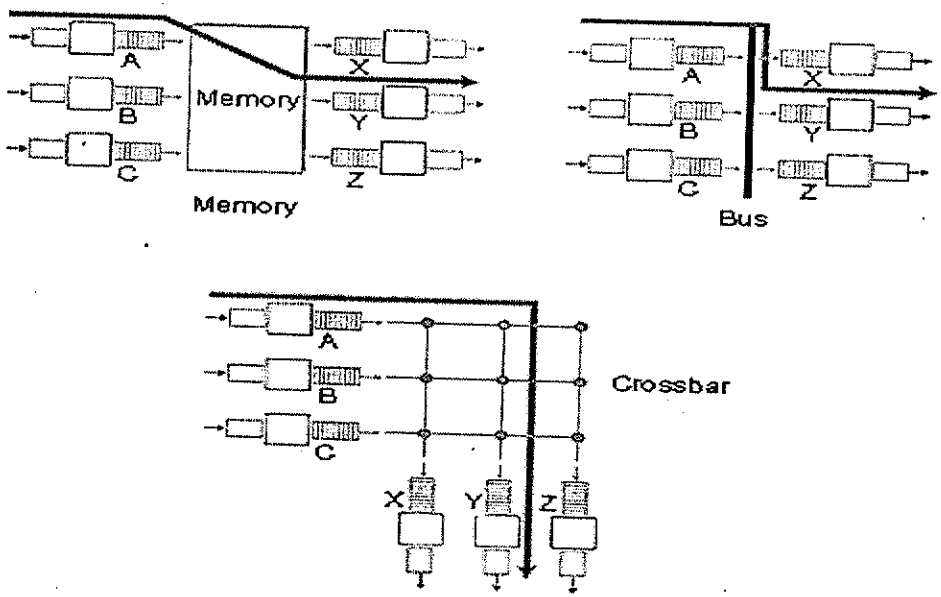
Khi làm việc ở đại học Stanford, Len Bosak cùng vợ là Sandy Lerner nẩy ra ý tưởng sản xuất và bán router Internet cho các viện nghiên cứu và trường đại học. Sandy Lerner đưa ra tên gọi của công ty "Cisco" (viết tắt của SanFrancisco) và bà cũng thiết kế biểu tượng cho công ty. Ban đầu tổng hành dinh của công ty đặt ngay tại phòng khách của họ. Hai vợ chồng phải sử dụng thẻ tín dụng trong công việc kinh doanh và phải làm thêm công việc tư vấn vào ban đêm. Cuối năm 1986, doanh thu của Cisco lên tới 250000USD/tháng - một kết quả không tồi với công ty được cấp vốn chỉ bằng thẻ tín dụng mà không có sự hùn vốn nào. Cuối năm 1987 Cisco huy động được 2 triệu đô từ Sequoia Capital đổi lấy việc kiểm soát 1/3 công ty. Vài năm tiếp theo Cisco tiếp tục phát triển và giành ngày càng nhiều thị phần. Cùng thời điểm đó, quan hệ giữa vợ chồng Bosak, Lener với đội ngũ quản lý của Cisco bắt đầu trục trặc. Cisco niêm yết trên thị trường cổ phiếu vào năm 1990. Cùng năm đó, Cisco sa thải Lerner và Bosak về hưu.

4.6.2 Kết cấu chuyển (Switching fabric)

Kết cấu chuyển nằm ở trung tâm của router. Gói dữ liệu chuyển từ cổng vào đến cổng ra qua kết cấu chuyển. Việc chuyển được thực hiện bằng nhiều cách như minh họa trên Hình 4.27

Chuyển qua bộ nhớ: Các router đơn giản nhất thuộc thế hệ đầu tiên thường chính là các máy tính truyền thống, việc chuyển từ cổng vào tới cổng ra được thực hiện dưới sự điều khiển trực tiếp của CPU. Cổng vào và cổng ra chỉ là các thiết bị vào ra truyền thống trong hệ điều hành. Khi nhận được một gói dữ liệu, cổng vào sẽ sử dụng ngắt để báo cho CPU. Sau đó gói dữ liệu được sao chép vào bộ nhớ của vi xử lý. Bộ vi xử lý lấy địa chỉ đích từ tiêu đề gói tin, tìm cổng ra trong bảng định tuyến và sao chép gói dữ liệu vào bộ đệm của cổng ra.

Nhiều router hiện đại cũng thực hiện việc chuyển qua bộ nhớ. Điểm khác biệt chính ở chỗ việc xác định địa chỉ đích và việc lưu trữ (chuyển) gói dữ liệu vào vị trí phù hợp trong bộ nhớ được thực hiện bởi bộ xử lý trên mạch cổng vào. Router chuyển qua bộ nhớ, theo một cách nào đó giống hệ thống nhiều bộ xử lý với bộ nhớ dùng chung, bộ xử lí trên cổng vào sẽ đặt gói dữ liệu vào bộ nhớ của một cổng ra thích hợp. Dòng router Cisco Catalyst 8500 và đòng Bay Network Accelar 1200 chuyển gói dữ liệu qua bộ nhớ dùng chung.



Hình 4.27 Kết cấu chuyển

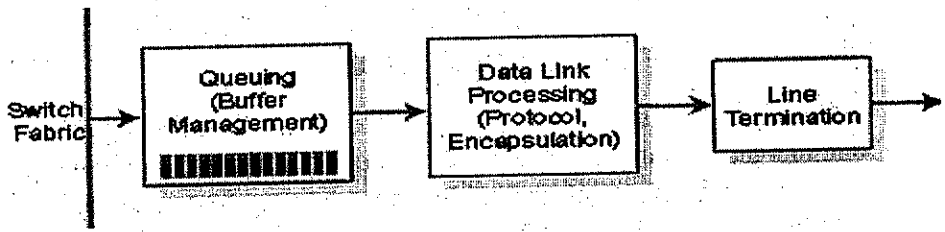
Chuyển qua bus. Cổng vào chuyển thẳng gói tin tới cổng ra qua một đường bus dùng chung mà không cần bộ xử lí của router can thiệp (chú ý là khi chuyển qua bộ nhớ, gói tin phải qua bus hệ thống để đến hay đi khỏi bộ nhớ). Mặc dù bộ xử lí của router không liên quan đến việc chuyển trên bus, song do bus dùng chung, tại một thời điểm chỉ cho phép một gói tin được truyền dẫn trên bus. Một gói tin đến cổng vào và thấy bus đang bị chiếm dụng bởi gói tin khác sẽ tạm thời bị chặn lại, đưa vào hàng đợi ở cổng vào. Vì tất cả các gói tin đều phải truyền qua một bus duy nhất, tốc độ chuyển của router bị giới hạn bởi tốc độ bus. Với công nghệ băng thông của bus vượt qua 1Gbit/s, chuyển mạch qua bus đủ hiệu quả với các router hoạt động ở mức tổ chức (ví dụ mạng cục bộ). Dòng Cissco 1900 chuyển mạch các gói qua bus 1Gbps.

Chuyển mạch qua một liên mạng. Một cách khắc phục hạn chế của bus dùng chung duy nhất là sử dụng một mạng liên kết phức tạp, giống các kĩ thuật được sử dụng để kết nối những bộ xử lí trong hệ thống đa bộ xử lý.

Hình 4.27 minh họa một mạng liên kết sử dụng 2N bus để nối N cổng vào với N cổng ra. Một gói tin đến từ một cổng vào sẽ được chuyển dọc theo bus nằm ngang gắn với cổng vào cho tới khi gặp giao điểm với bus nằm dọc gắn với cổng ra tương ứng. Nếu bus nằm dọc đó rỗi, gói tin sẽ được chuyển trên bus dọc đó tới cổng ra cần đến. Trong trường hợp ngược lại, gói tin tạm thời bị chặn lại và xếp hàng tại cổng vào. Dòng Cisco 12000 Family sử dụng công nghệ này.

4.6.3 Cổng ra (Output port)

Quá trình xử lí tại cổng ra được minh họa trên Hình 4.28: lấy gói dữ liệu đã được lưu trữ trong bộ đệm của cổng ra và truyền qua đường liên kết ra. Các chức năng xử lí giao thức liên kết dữ liệu và kết thúc đường truyền là chức năng tầng liên kết dữ liệu và tầng vật lí để làm việc với đầu vào bên kia của đường truyền vật lý. Chức năng quản lí vùng đệm và hàng đợi được sử dụng khi tốc độ dữ liệu mà kết cấu chuyển chuyển tới nhanh hơn tốc độ gửi đi của cổng ra.



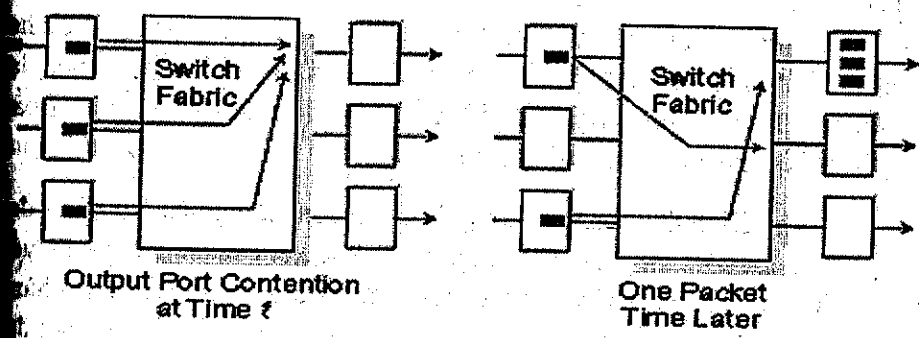
Hình 4.28 Cấu trúc cổng ra

4.6.4 Hàng đợi ở router

Nếu nhìn vào chức năng, cấu hình của cổng vào, cổng ra trong Hình 4.27, rõ ràng hàng đợi của các gói tin có thể được hình thành tại cả cổng vào và cổng ra. Chúng ta sẽ trình bày chi tiết về hàng đợi vì khi hàng đợi lớn, bộ

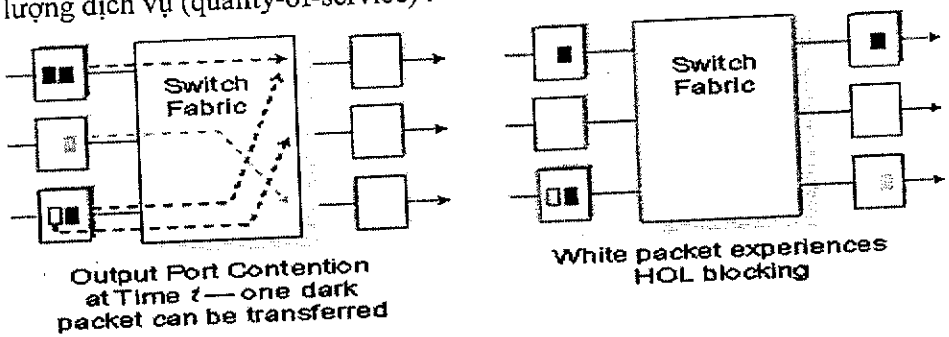
đệm của router sẽ đầy lên và xảy ra hiện tượng mất gói tin (tràn bộ đệm). Trong các phần trước, chúng ta đã nói mơ hồ rằng gói tin bị mất đâu đó "trong mạng" hay "tại router". Chính tại các hàng đợi của router, gói tin bị mất. Trên thực tế, vị trí gói tin bị mất (cổng vào hay cổng ra) phụ thuộc vào tải của mạng, tốc độ tương đối giữa kết cấu chuyển và đường truyền.

Giả sử tốc độ đường truyền vào và ra bằng nhau, và có n cổng vào, n cổng ra. Nếu tốc độ chuyển của kết cấu chuyển lớn hơn tốc độ đường truyền n lần thì chắc chắn không có hàng đợi tại cổng vào. Bởi trong tình huống xấu nhất, tất cả n đường vào cùng nhận được gói tin, kết cấu chuyển có khả năng truyền n gói tin từ cổng vào tới cổng ra. Nhưng điều gì có thể xảy ra tại cổng ra? Giả sử tốc độ kết cấu chuyển vẫn nhanh hơn tốc độ đường truyền n lần. Trong trường hợp xấu nhất, tất cả gói tin từ n cổng vào cùng đến một cổng ra. Mỗi lần cổng ra chỉ có thể gửi đi một gói tin duy nhất, do đó n gói tin sẽ phải xếp hàng tại cổng ra để đợi truyền. Khi số lượng gói tin xếp hàng vượt qua độ lớn bộ đệm, các gói tin đến sau sẽ bị mất. Hàng đợi tại cổng ra được minh hoạ trong Hình 4.29. Tại thời điểm t, tất cả các cổng vào đều nhận được một gói tin và phải chuyển tới cổng ra trên cùng bên phải. Giả sử tốc độ ba đường truyền là như nhau và tốc độ kết cấu chuyển nhanh hơn tốc độ đường truyền ba lần. Sau một đơn vị thời gian (thời gian cần thiết để nhận hay gửi một gói tin), cả 3 gói tin được chuyển tới cổng ra và xếp hàng để đợi truyền đi. Trong một đơn vị thời gian tiếp theo, một gói tin được truyền đi trên đường truyền ra. Cùng lúc đấy, lại có thêm hai gói tin khác chuyển tới cổng ra trên cùng và do đó sẽ phải xếp hàng.



Hình 4.29 Mất gói tin tại Router

Bộ điều phối (**scheduler**) tại cổng ra phải chọn một gói tin trong hàng đợi để truyền đi. Có thể sử dụng một cơ chế đơn giản như First-Come-First-Served (Người đến trước được phục vụ trước-FCFS) hay cơ chế hàng đợi có trọng số (WFQ – weighted fair queuing) phức tạp hơn, cho phép chia sẻ một cách tương đối công bằng đường truyền ra giữa các kết nối đầu cuối khác nhau. Bộ điều phối có vai trò quyết định trong trong việc bảo đảm chất lượng dịch vụ (quality-of-service) .



Hình 4.30 Ví dụ về tắc nghẽn

Nếu kết cấu chuyển không đủ nhanh (so với tốc độ cổng vào) để chuyển ngay lập tức tất cả các gói tin qua, hàng đợi sẽ xuất hiện tại cổng vào, vì khi đó các gói tin sẽ phải xếp hàng đợi đến lượt chuyển qua kết cấu chuyển tới cổng ra. Để minh hoạ xét chuyển mạch qua một liên mạng trong Hình 4.30 và giả sử (1) tốc độ của tất cả các liên kết bằng nhau, (2) thời gian gói tin chuyển từ cổng vào tới cổng ra bằng thời gian cổng vào nhận được một gói tin và (3) các gói tin được chuyển từ cổng vào tới cổng ra theo thứ tự đến (FCFS). Nhiều gói tin có thể được truyền đồng thời miễn là cổng ra của chúng khác nhau. Tuy nhiên nếu hai gói tin xếp đầu hàng đợi trên hai cổng vào khác nhau cùng hướng tới một cổng ra thì một gói tin sẽ bị chặn lại tại cổng vào (phải xếp hàng trong hàng đợi) vì tại một thời điểm kết cấu chuyển chỉ có thể chuyển đi một gói tin.

Trong Hình 4.30 ta thấy hai gói tin (tô đen) đứng đầu hai hàng đợi cùng hướng tới cổng ra phía trên bên phải. Giả sử kết cấu chuyển sẽ chuyển gói tin từ hàng đợi phía trên bên trái trước. Khi đó gói tin màu đen ở hàng đợi phía dưới bên trái sẽ phải đợi. Nhưng không chỉ có gói tin này phải đợi mà còn gói tin màu trắng xếp hàng sau nó cũng phải đợi - mặc dù các gói tin này hướng tới cổng ra khác. Đây là hiện tượng head-of-the-line (HOL)

blocking. [Karol 1997] đã chứng minh rằng [illegible]
hàng đợi tăng lên vô hạn ngay cả khi tốc độ đến của các gói tin trên cổng vào bằng 58% tốc độ tối đa. [McKeowwn 1997b] đưa ra nhiều giải pháp ngăn chặn HOL.

4.7 IPv6

Đầu những năm 1990, Internet Engineering Task Force bắt đầu nỗ lực phát triển giao thức mạng thay thế IPv4. Nguyên nhân đầu tiên cho nỗ lực này là không gian địa chỉ IP 32 bit đã bắt đầu cạn kiệt trong khi số lượng những mạng mới và những nút mạng được kết nối vào Internet (cần cấp phát một địa chỉ IP duy nhất) tăng lên đáng kể. Để giải quyết nhu cầu có không gian địa chỉ IP lớn hơn, giao thức mạng IPv6 đã được phát triển. Những người thiết kế IPv6 cũng chọn lọc các tính năng, cải tiến nhiều đặc điểm khác của IPv4 dựa trên cơ sở những kinh nghiệm thực tế của IPv4.

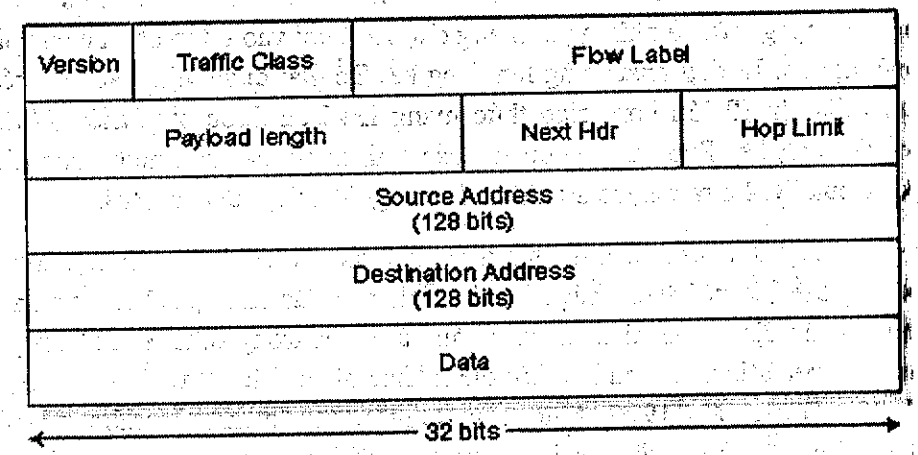
Người ta chưa thống nhất được khi nào địa chỉ IPv4 cạn kiệt (khi đó không thể kết nối thêm bất kỳ máy tính nào vào mạng). Căn cứ trên xu hướng cấp địa chỉ IP hiện tại hai nhóm làm việc trong IETF's Address Life time Expectations đưa ra hai thời điểm khác nhau là 2008 và 2018 [Solenky 1996]. Trong năm 1996, American Registry for Internet Number (ARIN) thông báo tất cả các địa chỉ IPv4 lớp A, 62% lớp B và 37% lớp C đã được phân phối [ARIN 1996]. Mặc dù những đánh giá và các con số dự đoán trên cho thấy còn có nhiều thời gian cho tới khi không gian địa chỉ IPv4 hết, song đã đến lúc triển khai một công nghệ mới trên một quy mô rộng lớn, và do đó "Next Generation IP" (thế hệ IP mới) [Brander 1996:RFC1752] đã bắt đầu triển khai.

4.7.1 Định dạng gói tin IPv6

Khuôn dạng gói dữ liệu IPv6 được minh họa trên hình Hình 4.31. Điểm thay đổi quan trọng nhất của IPv6 chính là khuôn dạng gói tin.

Mở rộng khả năng đánh địa chỉ. IPv6 tăng kích thước địa chỉ IP từ 32 bit lên 128 bit. Nó đảm bảo khả năng không bị thiếu địa chỉ IP. Với không gian 128 bit, có thể đánh địa chỉ cho đến từng hạt cát có trên trái đất. Bên cạnh địa chỉ duy nhất (unicast) và địa chỉ đa đích (multicast), IPv6 còn có một dạng địa chỉ mới gọi là "anycast address", cho phép một gói tin với địa chỉ đích thuộc kiểu "anycast address" có thể được chuyển tới một nhóm các máy tính (đặc điểm này sẽ được sử dụng ví dụ khi gửi thông điệp HTTP GET tới nhiều site phụ chứa cùng một tài liệu nào đấy).

Tiêu đề có độ dài cố định 40 byte. Một số trường IPv4 mang tính chất tùy chọn. Tổng độ dài tiêu đề cố định cho phép xử lí các gói dữ liệu IPv6 nhanh hơn.



Hình 4.31 Khuôn dạng địa chỉ IPv6

Gắn nhãn luồng (flow label) và độ ưu tiên (priority). IPv6 không có định nghĩa cho "flow" một cách rõ ràng. Các khuyến nghị RFC 1752 và RFC 2460 cho phép gắn nhãn cho các gói tin thuộc về cùng một "flow". Các gói tin này đòi hỏi được xử lý một cách đặc biệt, như các dịch vụ thời gian thực với chất lượng tốt hơn. Ví dụ, các dữ liệu đa phương tiện có thể xem như một luồng liên tục. Dữ liệu các ứng dụng truyền thống, như truyền file, E-mail không được xem như một luồng. Có thể dữ liệu của những người có độ ưu tiên cao (ví dụ người trả phí cao hơn) cũng có thể coi như một luồng. Rõ ràng ở đây những người thiết kế IPv6 đã dự đoán được nhu cầu phân biệt giữa các luồng dữ liệu ngay cả khi chưa định nghĩa chính xác được luồng là gì. Tiêu đề IPv6 cũng có trường Traffic Class 8 bit. Trường này giống

trường TOS (Tyte of Service) trong IPv4 có thể được sử dụng cho những gói n có quyền ưu tiên trong một luồng, hoặc cho những ứng dụng có độ ưu iên cao (ví dụ gói tin ICMP).

So sánh khuôn dạng gói dữ liệu IPv4 (Hình 4.20) và IPv6 (Hình 31), ta thấy gói IPv6 có cấu trúc đơn giản hơn. Sau đây là một số trường rong gói dữ liệu IPv6:

Phiên bản (version). Trường 4-bit này xác định phiên bản IP của ói dữ liệu. Rõ ràng gói IPv6 có giá trị "6" trong trường này. Chú ý không hải đặt giá trị "4" trong trường này thì gói dữ liệu là IPv4.

Traffic class. Trường 8-bit này giống trường TOS trong IPv4

Nhãn luồng (Flow label). Trường 20 bit này xác định một luồng hứa gói dữ liệu.

Độ lớn dữ liệu (Payload length). Độ lớn (tính theo byte) của phần ữ liệu không tính tiêu đề.

Next header. Trường này xác định giao thức ở tầng phía trên sẽ hận dữ liệu (ví dụ tới TCP hoặc UDP). Trường này giống trường Protocol ủa IPv4.

Hop limit. Giá trị của trường này sự giảm đi 1 khi đi qua mỗi router. Nếu giá trị này bằng 0, gói dữ liệu bị loại bỏ.

Địa chỉ nguồn và đích (source and destination addresss). Khuôn dạng 128-bit địa chỉ IPv6 được đặc tả trong RFC 2373.

Dữ liệu (data). Khi gói tin IPv6 tới đích, các tiêu đề sẽ bị loại bỏ và hần dữ liệu này sẽ được chuyển đến thực thể ở tầng phía trên.

Có một số trường trong IPv4 không xuất hiện trong IPv6 nữa.

Phân mảnh, Hợp nhất gói tin. IPv6 không cho phép phân mảnh và ợp nhất gói tin tại các router trung gian. Nếu một gói dữ liệu IPv6 quá lớn ể có thể gửi đi trên một đường liên kết ra của router, router sẽ loại bỏ gói in này và gửi một thông báo lỗi ICMP "Packet Too Big" tới bên gửi. Sau đó ên gửi gửi lại dữ liệu, sử dụng các gói dữ liệu có kích thước nhỏ hơn. Việc

phân mảnh và hợp nhất các gói tin IP chiếm nhiều thời gian xử lý của router. Thực hiện những công việc này tại các thiết bị đầu cuối sẽ làm tăng tốc độ truyền trên mạng.

Checksum. Do tầng giao vận (ví dụ, TCP và UDP) và các giao thức liên kết dữ liệu (ví dụ Ethenet) đã thực hiện kiểm tra lỗi, chức năng này không cần thiết trong tầng mạng nên được bỏ đi. Vấn đề xử lý nhanh các gói tin IP cực kỳ quan trọng. Khi nói về IPv4 trong phần 4.4.1, chúng ta đã thấy giá trị trường TTL trong tiêu đề của IPv4 giảm đi một khi đi qua mỗi router, nên giá trị trường checksum trong tiêu đề IPv4 cần phải được tính lại tại các router. Như vậy, giống như phân mảnh và hợp nhất, việc này khiến thời gian xử lý gói IPv4 lâu hơn.

ICMP cho IPv6

Trong phần 4.4, 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. Bên cạnh các kiểu và mã cũ, ICMPv6 cũng đưa thêm vào nhiều kiểu và mã mới. Ví dụ kiểu mã lỗi "Packet Too Big" hay "Unrecognized IPv6 option".

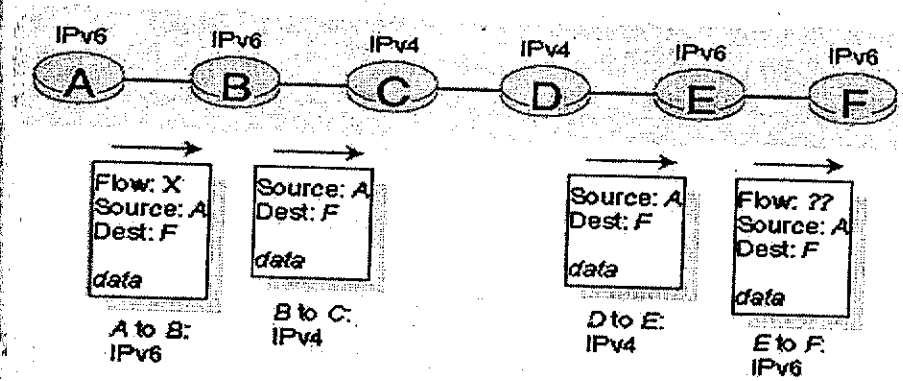
4.7.2 Chuyển từ IPv4 sang IPv6

Chúng ta đã xem xét các chi tiết kỹ thuật của IPv6, bây giờ chúng ta sẽ xét tới một vấn đề rất thực tiễn: làm thế nào để chuyển mạng toàn cầu Internet hiện tại – đang sử dụng IPv4 sang sử dụng IPv6? Vấn đề ở chỗ trong khi các hệ thống mới IPv6 có khả năng tương thích ngược tức là có thể gửi, chuyển, nhận các gói dữ liệu IPv4 thì các hệ thống cũ đang được sử dụng lại không có khả năng xử lý gói dữ liệu IPv6. Người ta đã đưa ra một số giải pháp.

Giải pháp thứ nhất là lựa chọn một thời điểm nào đó, tắt tất cả máy để nâng cấp lên IPv6. Việc chuyển đổi công nghệ quan trọng gần đây nhất (từ NCP sang TCP cho giao thức giao vận tin cậy) cách đây 20 năm. Ngay

cả thời điểm đó [RFC 801], khi Internet còn rất nhỏ và vẫn còn được quản trị bởi một nhóm nhỏ các chuyên gia, người ta cũng không thể chọn được một thời điểm như vậy. Giải pháp này ngày nay sẽ đòi hỏi sự tham gia của hàng trăm triệu máy tính và hàng triệu người quản trị mạng – rõ ràng là không thể. 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 (dĩ nhiên mục tiêu dài hạn vẫn là chuyển tất cả các thiết bị sử dụng IPv4 sang IPv6).

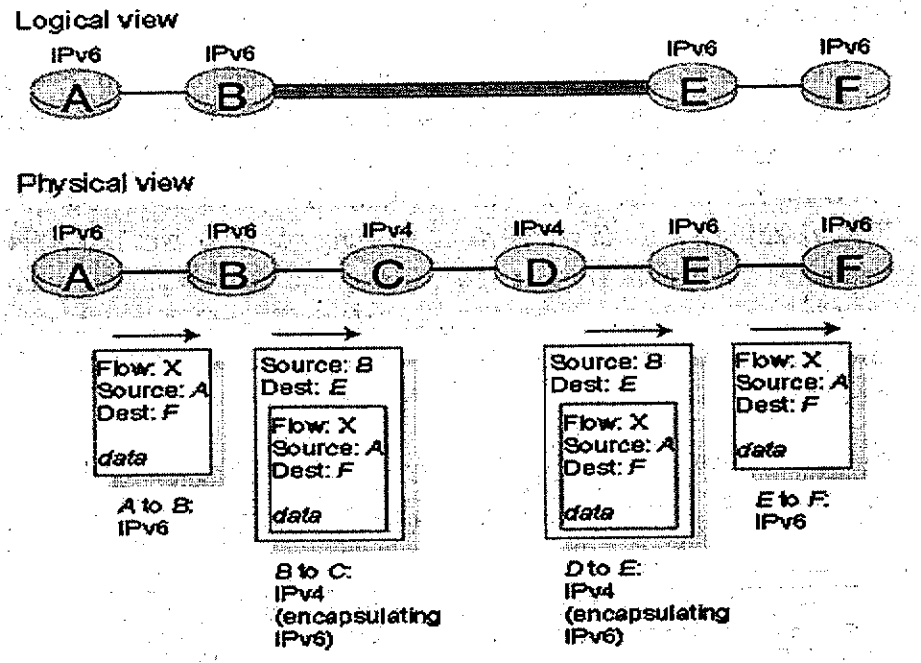
Có thể giải pháp đơn giản nhất để đưa vào các thiết bị hỗ trợ IPv6 là **dual-stack**, các thiết bị triển khai cả IPv4 và IPv6. Thiết bị IPv6/IPv4 như vậy được đặc tả trong RFC 1933 có khả năng nhận và gửi cả hai gói dữ liệu IPv4 và IPv6. Khi trao đổi với một nút IPv4, nút IPv6/IPv4 sử dụng gói dữ liệu IPv4 và khi trao đổi với nút IPv6, sẽ sử dụng gói IPv6. Nút IPv4/IPv6 cần phải có cả hai địa chỉ IPv6 và IPv4. Chúng cần phải có khả năng xác định được một nút có khả năng IPv6 hay chỉ hỗ trợ IPv4. Có thể giải quyết vấn đề này nhờ Hệ thống tên miền DNS (Domain Name Server) (xem trong chương 2).



Hình 4.32 Giải pháp dual-stack

Trong giải pháp trên, nếu cả bên gửi và bên nhận chỉ hỗ trợ IPv4 thì giữa chúng chỉ có thể trao đổi gói dữ liệu IPv4. Do đó có thể hai nút có khả năng IPv6 chỉ có thể trao đổi gói dữ liệu IPv4 với nhau. Trong Hình 4.32,giả sử nút A hỗ trợ IPv6 muốn gửi một gói dữ liệu IP tới nút F cũng hỗ trợ IPv6. Nút A và B có thể trao đổi gói tin IPv6 với nhau. Tuy nhiên, nút B phải tạo gói dữ liệu IPv4 để gửi tới C. Chắc chắn một số trường tiêu đề của gói IPv6

được sao chép vào trường tiêu đề của gói dữ liệu IPv4 và phải thực hiện việc đổi địa chỉ. Tuy nhiên, khi chuyển từ IPv6 sang IPv4, sẽ có một số trường đặc thù trong IPv6 sẽ không có trong IPv4 (ví dụ trường flow). Các thông tin trong các trường này sẽ bị mất. Cho dù E và F có thể trao đổi gói dữ liệu IPv6, gói dữ liệu IPv4 từ D tới E không có đầy đủ tất cả các thông tin trong gói dữ liệu IPv6 nguyên thủy được gửi từ A.



Hình 4.33 Giải pháp đường ống

Giải pháp thứ hai được nêu ra trong khuyến nghị RFC 1933 là giải pháp **đường ống (tunneling)**. Giải pháp này cho phép khắc phục được vấn đề nêu trên, cho phép E nhận gói IPv6 nguyên bản từ A. Sau đây trình bày ý tưởng cơ bản của giải pháp này. Giả sử hai nút IPv6 (ví dụ B và E trong Hình 4.33) muốn gửi gói IPv6 cho nhau, nhưng giữa chúng là các router chỉ hỗ trợ IPv4. Các router IPv4 nằm giữa hai router IPv6 là một **đường ống**, như minh hoạ trong Hình 4.33. Hai nút IPv6 ở hai đầu của đường ống (B chẳng hạn) đặt toàn bộ gói dữ liệu IPv6 vào trong trường dữ liệu (payload) của gói dữ liệu IPv4. Gói dữ liệu IPv4 này có địa chỉ đích là nút IPv6 ở đầu

kia đường ống (ví dụ là E) và được gửi tới nút đầu tiên trong đường ống (C). Các router IPv4 trong đường ống sẽ định tuyến gói dữ liệu IPv4 giống như bất kỳ gói dữ liệu IPv4 nào mà không biết được gói dữ liệu IPv4 đó chứa gói dữ liệu IPv6. Nút IPv6 phía bên kia đường ống nhận được gói IPv4, xác định trong đó có gói IPv6, lấy gói IPv6 ra, và sau đó chuyển tiếp gói IPv6 như thể nó nhận thẳng gói IPv6 từ phía bên kia.

Vậy khi nào IPv6 có bước đột phá thực sự vào mạng Internet toàn cầu. Thực tế hiện nay các ISP Bắc Mỹ không có dự định mua thiết bị có hỗ trợ IPv6 vì ít khách hàng đòi hỏi điều này. Nhu cầu sử dụng IPv6 chủ yếu xuất phát từ châu Âu và châu Á.

Rõ ràng cực kỳ khó khăn để thay đổi các giao thức của tầng mạng. Từ đầu những năm 90, rất nhiều giao thức mới của tầng mạng đã được đưa ra nhưng vẫn chưa được áp dụng thực sự trong thực tế. Thực vậy, thay đổi giao thức tầng mạng giống như thay đổi móng của một ngôi nhà. Sẽ rất khó thực hiện nếu không phá hủy hoàn toàn ngôi nhà hoặc ít nhất tạm thời bố trí những người sinh sống trong đó đến các địa điểm khác. Mặt khác, Internet chứng kiến sự phát triển nhanh chóng các giao thức tầng ứng dụng. Một ví dụ hiển nhiên là HTTP và WEB. Đưa ra những giao thức ứng dụng mới giống như việc thêm một lớp sơn mới cho ngôi nhà - nó thực sự dễ dàng, và nếu bạn chọn một màu hấp dẫn, những người xung quanh sẽ bắt chước làm theo bạn. Tóm lại, trong tương lai chúng ta có thể kỳ vọng thấy được sự thay đổi tầng mạng của Internet, nhưng những thay đổi này sẽ diễn ra rất chậm so với những thay đổi tại tầng ứng dụng.

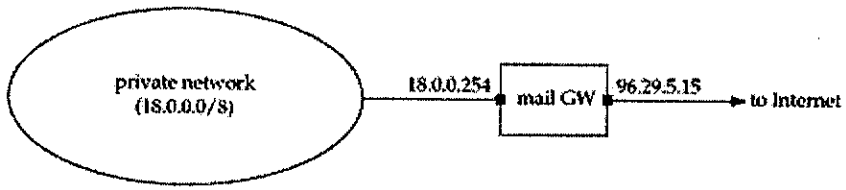
4.8 CƠ CHẾ DỊCH CHUYỂN ĐỊA CHỈ (NAT)

Trong trường hợp tổ chức có nhu cầu kết nối nhiều máy tính vào mạng nhưng lại chỉ được cấp phát một lượng nhỏ địa chỉ IP chính thức, vậy làm thế nào để tất cả các máy tính trong mạng có thể truy cập Internet ? NAT là một trong các giải pháp để thực hiện điều này. RFC 1918 xác định một số địa chỉ IP đặc biệt gọi là địa chỉ riêng (Hình 4.34). Gói tin có địa chỉ trong dải địa chỉ này sẽ không được router chuyển ra phía ngoài.

Khối	Dải địa chỉ
10.0.0.0/8	10.0.0.1 – 10.255.255.255
172.16.0.0/12	172.16.0.0 – 172.31.255.255
192.168.0.0/16	192.168.0.0 – 192.168.255.255

Hình 4.34 Các địa chỉ riêng

Dĩ nhiên nếu tổ chức không có nhu cầu kết nối ra Internet thì sử dụng bất kỳ địa chỉ IP nào cũng được. Trước khi có RFC 1918, mọi tổ chức đều có khuynh hướng chọn một lớp địa chỉ bất kỳ. Trong Hình 4.35, tổ chức chọn địa chỉ 18.0.0.0/8. Tuy nhiên giả sử sau đó tổ chức muốn cài đặt Mail server để máy tính ở trong mạng nội bộ có thể gửi và nhận thư với bên ngoài. Để nối với các máy tính bên trong mạng nội bộ, mail server sử dụng địa chỉ 18.0.0.254. Để kết nối ra ngoài Internet, mail server sử dụng địa chỉ 96.29.5.15 thực xác định duy nhất trên Internet.



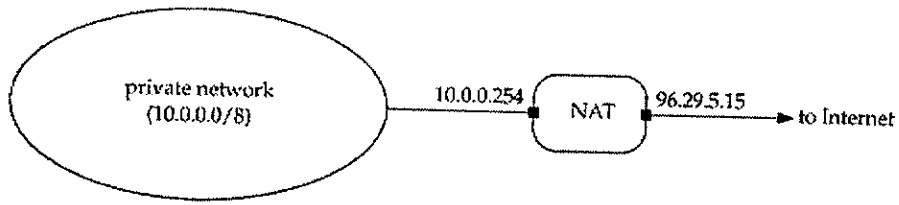
Hình 4.35 Ví dụ một tổ chức dùng Gateway Mail Server

Bây giờ giả sử người trong cơ quan muốn gửi email đến ai đó ở MIT. Vấn đề ở đây là địa chỉ thật của MIT là 18.0.0.0/8. Do đó mail gateway sẽ chuyển ngược thư vào lại mạng nội bộ, chứ không chuyển tới miền mit.edu. Vấn đề này sẽ không còn nữa nếu cơ quan sử dụng địa chỉ riêng.

Một ưu điểm khác của NAT là cho phép cấu hình mạng của cơ quan không phụ thuộc vào bất kỳ ISP nào cả.

Trước khi đi sâu vào chi tiết, chúng ta sẽ tìm hiểu qua cơ chế Dịch chuyển địa chỉ mạng NAT (Network Address Translation). Cơ chế này được cài đặt trên router nối ra ngoài của mạng. Nếu chúng ta thay thế mail gateway trên Hình 4.35 bằng một router có hỗ trợ cơ chế NAT và đánh lại

địa chỉ cho các máy tính bên trong là địa chỉ 10.0.0.0/8, chúng ta được Hình 4.36. Chúng ta quay lại ví dụ trên, giả sử máy 10.0.0.1 bên trong mạng riêng muốn gửi email đến địa chỉ 18.7.7.76. Kết nối TCP này không thể được thiết lập do mail server ở MIT không thể gửi lại gói tin phản hồi đến địa chỉ 10.0.0.1



Hình 4.36 Mạng riêng với Router có hỗ trợ cơ chế NAT

Về mặt lý thuyết, gói tin gửi từ 10.0.0.1 sẽ đến được MIT server. Nhưng trên thực tế, các router và mail server thường loại bỏ các gói tin có địa chỉ gửi là địa chỉ riêng với mục đích phòng ngừa tấn công từ chối dịch vụ (DoS) hay các thư rác.

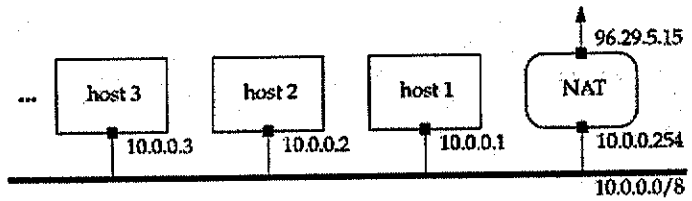
Chính vì vậy gói tin trước khi rời mạng riêng, địa chỉ gửi trong gói tin phải được thay đổi để bên kia có thể gửi thông điệp phản hồi.

NAT hoạt động theo một trong ba chế độ sau:

1. **Cơ chế tĩnh**. Mỗi máy tính trong mạng riêng khi truy cập đến Internet sẽ có một địa chỉ thật. NAT chỉ thực hiện việc ánh xạ từ địa chỉ ảo vào địa chỉ thực và ngược lại.
2. **Cơ chế vòng**. Tất cả các máy tính trong mạng riêng sử dụng chung một nhóm địa chỉ chính thức. Ví dụ mạng riêng với 300 máy tính được cấp phát một dải 32 địa chỉ thật. Khi một máy tính trong mạng riêng muốn kết nối với một máy tính bên ngoài, nó sẽ được gán cho một địa chỉ thực tạm thời. Sau khi kết nối kết thúc, địa chỉ vừa được cấp phát sẽ được thu hồi để cấp cho máy tính có nhu cầu khác. Hiển nhiên khi hết các địa chỉ thật, các máy khác có nhu cầu cũng không được kết nối.

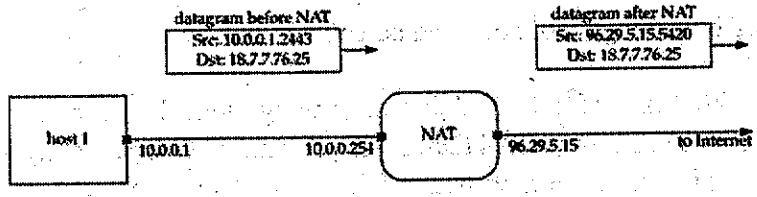
Cơ chế chuyển địa chỉ theo cổng (PAT). Đây là cơ chế được sử dụng rộng rãi nhất. Cơ chế này được sử dụng khi có một địa chỉ thật duy

nhất dùng chung cho cả tổ chức. Khi đó địa chỉ port gửi của mỗi gói tin được thay bằng một giá trị xác định duy nhất. Giá trị này sẽ được sử dụng để khi nhận một gói tin, router biết phải chuyển tiếp nó cho máy tính nào trong mạng riêng.Để hiểu rõ cơ chế hoạt động của PAT, chúng ta minh họa mạng riêng ở Hình 4.36 thành Hình 4.37.



Hình 4.37 Ví dụ về mạng sử dụng cơ chế PAT

Giả sử Host 1 muốn thiết lập kết nối với email server của MIT có địa chỉ 18.7.7.76 thông qua proxy 2443 Hình 4.38 minh họa đường đi của gói tin IP chứa TCP SYN segment gửi tới MIT email server qua hai chặng: chặng thứ nhất từ Host 1 đến Router và chặng thứ hai từ router đến MIT server. Cơ chế PAT được sử dụng để biến đổi gói tin này giữa hai chặng



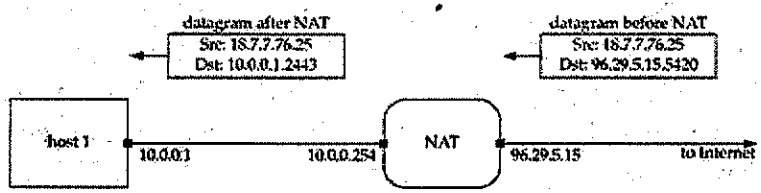
Hình 4.38 Gói tin từ Host 1 đến router và được chuyển tiếp

Đầu tiên gói tin được gửi từ Host 1 có địa chỉ gửi là địa chỉ riêng (**10.0.0.1**) và port gửi là **2443**. Sau khi đi qua router, địa chỉ gửi của gói tin là **96.29.5.15** và port gửi là **5420**.

MIT's mail server thấy kết nối này đến từ **96.29.5.15:5420** và sẽ gửi thông điệp trả lời đến socket này.

Chúng ta thấy gói tin trả lời trong Hình 4.39. Chú ý rằng, khi đến router gói tin này có địa chỉ nhận là **96.29.5.15:5420** còn địa chỉ nguồn là

phía mail server của MIT. Router sẽ kiểm tra địa chỉ **96.29.5.15:5420** trong bảng PAT và thấy gói tin này phải được chuyển tiếp tới **10.0.0.1:2443**. Phía bên trái của Hình 4.39 minh họa gói tin sau khi đã được router biến đổi địa chỉ gửi theo cơ chế PAT.



Hình 4.39 Gói tin trả lời được NAT biến đổi

Giả sử tại cùng thời điểm Host 1 gửi, một máy tính khác trong mạng riêng, chẳng hạn Host 2 với địa chỉ **10.0.0.2** cũng gửi email tới MIT, khi đó NAT sẽ ánh xạ địa chỉ gửi của Host 2 vào **96.29.5.15** nhưng sẽ sử dụng một địa chỉ cổng khác, ví dụ **7322**. Do vậy khi nhận được gói tin có địa chỉ đích **96.29.5.15:7322**, NAT biết rằng cần chuyển gói tin này đến Host 2 (**10.0.0.2**).

Ưu điểm của NAT là hoàn toàn rõ ràng. Thứ nhất, nó làm tiết kiệm không gian địa chỉ IP đã sắp cạn kiệt, một mạng riêng khá lớn cũng chỉ cần một địa chỉ thật. Thứ hai là tổ chức có thể cấp phát địa chỉ riêng cho các máy tính trong cơ quan của mình và thực hiện ánh xạ địa chỉ tại router kết nối ra ngoài, do đó có sự độc lập với ISP. Nếu tổ chức muốn chuyển sang nhà cung cấp dịch vụ ISP khác thì chỉ cần cấu hình lại bảng NAT mà không cần gán lại địa chỉ IP cho các máy tính trong tổ chức. Như vậy, NAT là giải pháp rất phù hợp với tổ chức quy mô nhỏ.

Tuy nhiên, NAT cũng có nhiều nhược điểm. Trong phần trên, chúng ta thấy rằng dường như NAT chỉ là sự ánh xạ rất đơn giản giữa địa chỉ riêng và địa chỉ thật. Tuy nhiên có khá nhiều công việc trong quá trình ánh xạ. Thứ nhất địa chỉ IP gửi bị thay đổi trong trường tiêu đề nên giá trị trong trường checksum cũng phải tính lại. Thứ hai, router phải kiểm tra xem gói tin chứa TCP segment hay UDP segment, và sau đó phải thay đổi số hiệu cổng và checksum trong tiêu đề của TCP (hay UDP) segment. Như vậy,

router hỗ trợ NAT đã "vi phạm" cơ chế phân tầng do đã kiểm tra bên trong gói dữ liệu IP.

Có rất nhiều ứng dụng, chẳng hạn ftp sử dụng nhiều kết nối. FTP client gửi lệnh **PORT x y** tới FTP server để yêu cầu FTP server mở một kết nối tới socket địa chỉ IP **x** và cổng **y**. Địa chỉ và số hiệu cổng này được chuyển dưới dạng mã ASCII bên trong thông điệp ở tầng ứng dụng. Như vậy NAT phải biết điều này vì hai lí do. Thứ nhất, NAT phải ánh xạ lại địa chỉ riêng của client gửi đến server và NAT phải ghi nhớ (để có thể ánh xạ lại trong trường hợp có xung đột với cổng NAT đã gán khác). Thứ hai NAT phải xác định chính xác socket (IP/port) của máy tính trong mạng riêng đang đợi kết nối từ phía FTP server ở bên ngoài.

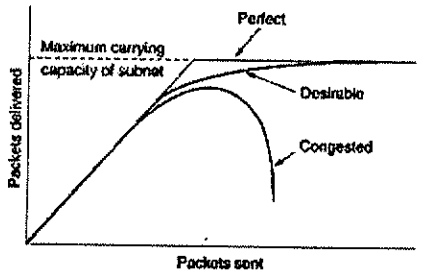
Để thực hiện điều này, NAT router phải kiểm tra gói tin TCP để xem nó có chứa lệnh **PORT** của ftp không. Nếu có, không những chúng ta phải thay đổi tiêu đề TCP segment mà còn phải thay đổi cả độ dài gói tin (đổi **10.0.0.1** thành **96.29.5.15** khiến độ dài trường dữ liệu trong TCP segment tăng thêm 2 byte). Khi độ dài trường dữ liệu thay đổi, các giá trị biên nhận (ACK) cũng phải thay đổi theo. Chẳng hạn như trong ví dụ trên, nếu byte cuối cùng mà client (bên trong mạng riêng) gửi có số thứ tự là n. Tuy nhiên do NAT router bổ sung thêm dữ liệu ở trong TCP segment nên byte cuối cùng của gói tin này có số thứ tự là n+2. Phía server sẽ trả lại giá trị biên nhận ACK là n+3 (hy vọng byte kế tiếp nhận được có số thứ tự là n+3). Khi đó NAT router phải đổi giá trị biên nhận ACK này thành n+1 khi chuyển tiếp cho client. Không chỉ thế NAT router phải ghi nhớ sự biến đổi này trong suốt phiên kết nối.

Chế độ thụ động của FTP có thể khắc phục vấn đề này. Trong chế độ này, phía client sẽ khởi tạo cả kết nối điều khiển lẫn kết nối dữ liệu. Khi đó, NAT router không cần biến đổi dữ liệu bên trong gói tin. Khi client muốn mở một kết nối dữ liệu tới server, nó sẽ gửi lệnh PASV yêu cầu server lắng nghe trên một cổng nào đó mà client chỉ định.

NAT có thể xử lý với ftp và một số rất ít các giao thức thông dụng khác nhưng có thể NAT làm hỏng các giao thức ở tầng ứng dụng do người dùng đặt ra nếu những giao thức này đặt địa chỉ cổng bên trong trường dữ liệu ứng dụng.

4.9 KIỂM SOÁT TẮC NGHẼN

Như chúng ta đã nói trong chương 3, hiện tượng tắc nghẽn là tại một thời điểm có quá nhiều gói tin đến cùng một phân đoạn mạng (hay một router) và hệ quả của tắc nghẽn là hiệu suất của hệ thống giảm nghiêm trọng. Trong Hình 4.40, chúng ta thấy rằng nếu tổng số gói tin trong mạng nhỏ hơn năng lực chuyển của mạng thì tất cả các gói tin đều sẽ được chuyển đến đích và số lượng gói tin truyền thành công bằng số lượng gói tin được gửi. Tuy nhiên khi lưu lượng gửi tăng nhanh, các router không kịp chuyển tiếp, các gói tin bắt đầu bị mất (do thiếu bộ đệm). Đến một giới hạn nào đó, hiệu suất hệ thống sụt giảm nghiêm trọng.



Hình 4.40 Đồ thị minh họa tắc nghẽn

Có nhiều tình huống dẫn đến tắc nghẽn. Chẳng hạn tại router, khi tất cả các gói tin đến từ nhiều cổng vào khác nhau nhưng lại cùng được chuyển đến một cổng ra, xuất hiện hàng đợi tại cổng ra này, do bộ đệm hữu hạn nên các gói tin đến sau có thể bị mất. Một số nghiên cứu cũng đã chứng minh được rằng, cho dù kích thước bộ đệm của router lớn vô cùng, tình trạng tắc nghẽn cũng không thể được giải quyết mà còn có xu hướng ngày càng xấu đi vì các gói tin nằm ở cuối hàng đợi chưa được chuyển đi, tại phía gửi do không nhận được ACK (sự kiện timeout) nên sau một thời gian sẽ tự động truyền lại gói tin (gửi trùng lặp). Các gói tin này sau đó lại "nằm xếp hàng" tại các router trên tuyến đường tới đích.

Tốc độ xử lý của CPU chậm cũng có thể là nguyên nhân gây tắc nghẽn. Nếu router không xử lý đủ nhanh, hàng đợi có thể xuất hiện tại router

(ngay cả khi công suất đường truyền vẫn chưa đạt tới cực đại). Tương tự, đường truyền có băng thông nhỏ cũng có thể gây ra tắc nghẽn. Nâng cấp đường truyền mà không nâng cấp router (hoặc ngược lại) không thể khắc phục được toàn bộ vấn đề, mà chỉ dịch chuyển "nút cổ chai" từ vị trí này trên mạng sang vị trí khác. Vấn đề ở đây là sự không cân đối giữa các phần khác nhau của hệ thống. Vấn đề chỉ được khắc phục hoàn toàn nếu đảm bảo được tính cân bằng trong hệ thống.

Ở đây chúng ta lại nhấn mạnh đến sự khác biệt giữa kiểm soát tắc nghẽn và điều khiển lưu lượng. Kiểm soát tắc nghẽn đảm bảo mạng máy tính có khả năng trung chuyển được khối lượng thông tin định trước. Điều này là vấn đề toàn cục, liên quan đến hành vi của tất cả các máy tính, các router. Ngược lại, điều khiển lưu lượng tập trung vào đường kết nối giữa hai đầu mút của truyền thông với mục đích kiểm soát để bên gửi không gửi quá tốc độ xử lý của bên nhận. Thông thường, phía nhận sẽ gửi thông báo phản hồi về tình trạng phía bên mình cho bên gửi.

Đôi khi hai khái niệm này không được phân biệt rõ ràng vì thuật toán kiểm soát tắc nghẽn cũng vận hành bằng cách gửi các thông báo phản hồi đến các nút trong mạng thông báo tình trạng tắc nghẽn. Do thế, nút gửi sẽ được yêu cầu gửi chậm lại khi hoặc mạng bị tắc nghẽn, hoặc khi nút nhận không kịp xử lý lượng dữ liệu gửi tới.

Chúng ta sẽ trình bày mô hình tổng quát xử lý tắc nghẽn, sau đó sẽ giới thiệu cách thức ngăn chặn để không xảy ra tắc nghẽn và sau đó là các thuật toán động – được sử dụng để khắc phục khi tắc nghẽn xuất hiện.

4.9.1 Các nguyên lý Kiểm soát tắc nghẽn

Đối với vấn đề tắc nghẽn, có hai chiến lược xử lý: Đóng và Mở. Giải pháp Mở là đưa ra một thiết kế đảm bảo tắc nghẽn không thể xảy ra. Quan trọng nhất trong giải pháp này là quyết định có chấp nhận một phiên truyền thông mới hay không ? Quyết định khi nào loại bỏ và loại bỏ gói tin nào ? Tuy nhiên tất cả các quyết định này được đưa ra không phụ thuộc vào trạng thái hiện thời của mạng.

Ngược lại, giải pháp Đóng dựa trên ý tưởng phản hồi và gồm ba giai đoạn:

1. Kiểm soát hệ thống để phát hiện thời điểm và vị trí xảy ra tắc nghẽn.
2. Chuyển thông tin cảnh báo về nơi có thể xử lý.
3. Điều chỉnh hệ thống để xử lý vấn đề.

Có rất nhiều tiêu chí khác nhau để kiểm soát tắc nghẽn. Thông dụng nhất là tỷ lệ phần trăm các gói tin bị loại bỏ do thiếu bộ đệm; độ lớn trung bình của hàng đợi; số lượng các gói tin không nhận được phản hồi và bị truyền lại; độ trễ trung bình của gói tin. Trong những trường hợp trên, những độ đo này tăng nghĩa là tắc nghẽn bắt đầu xuất hiện.

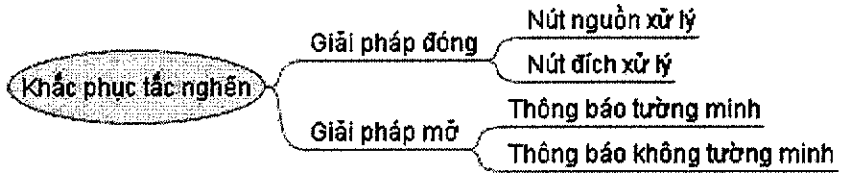
Bước phản hồi thứ hai là chuyển thông điệp cảnh báo từ nơi phát hiện tắc nghẽn đến nơi có thể đưa ra các hành động khắc phục. Một cách đơn giản là các router phát hiện ra tắc nghẽn gửi các gói tin đặc biệt đến tất cả các máy tính trên mạng thông báo về tình trạng tắc nghẽn. Rõ ràng nhược điểm của giải pháp này là phát sinh thêm các gói tin tại thời điểm đã có quá nhiều gói tin trên mạng (do tắc nghẽn).

Tuy nhiên người ta có thể đặt một trường nào đó bên trong mỗi gói tin và router sẽ thay đổi giá trị của trường này khi tắc nghẽn vượt qua một ngưỡng nào đó. Khi phát hiện tắc nghẽn, router sẽ thiết lập giá trị ở trường báo hiệu tắc nghẽn trong mỗi gói tin đi qua nó để cảnh báo các "hàng xóm".

Một phương pháp khác là các máy tính và router định kỳ gửi các thông điệp thăm dò với mục đích hỏi về tình trạng tắc nghẽn. Sau đó những thông tin nhận được có thể được router sử dụng để chuyển các gói tin vòng qua những khu vực đang bị tắc nghẽn. Điều này tương tự hệ thống cảnh báo tắc đường ở Mỹ. Trên một số trục xa lộ của nước Mỹ, có máy bay trực thăng bay để phát hiện xem có tắc đường không và những thông tin này sẽ được phát qua radio trên một số tần số cố định. Người lái xe có thể nghe được những thông tin này và chuyển hướng đi (không đi qua khu vực bị tắc).

Trong tất cả các phương pháp phản hồi, người ta hy vọng khi nhận được các thông tin báo hiệu tắc nghẽn, các máy tính trên mạng giảm bớt việc gửi tin để làm giảm tắc nghẽn. Có rất nhiều thuật toán kiểm soát tắc

nghẽn, và được phân loại như trong Hình 4.41.Giải pháp đóng được chia thành hai loại: Nút gửi thông tin sẽ xử lý tắc nghẽn và Nút nhận thông tin sẽ xử lý tắc nghẽn. Giải pháp mở cũng được chia thành hai loại: Tường minh (có thông báo tường minh gửi tới các nút trên mạng thông báo về tình trạng tắc nghẽn) và Không tường minh (các nút tự rút ra tình trạng tắc nghẽn qua các trạng thái cục bộ của mình – chẳng hạn như có quá nhiều gói tin phải gửi lại).

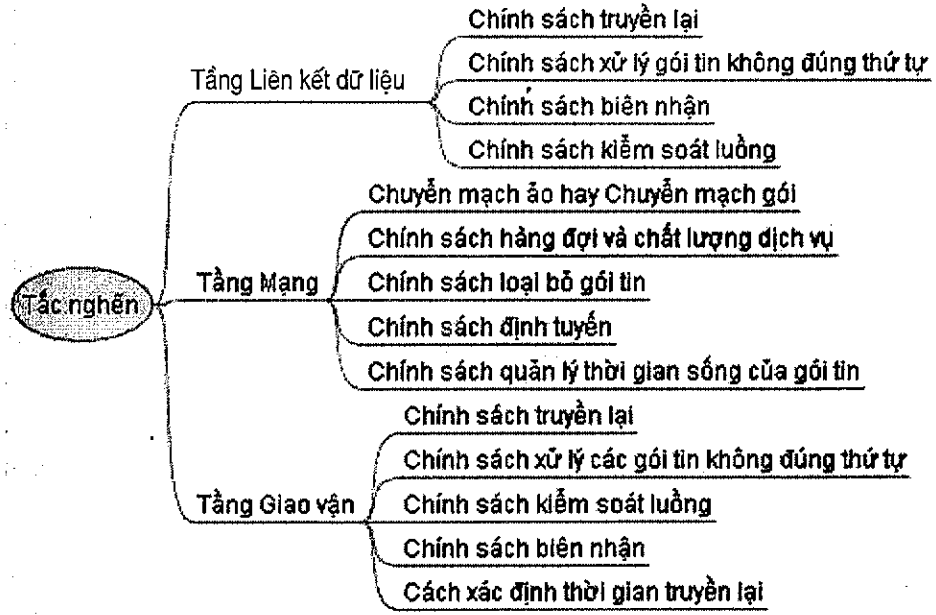


Hình 4.41 Phân loại các giải pháp xử lý tắc nghẽn

Nguyên nhân của tắc nghẽn là tại một thời điểm, tải lớn hơn năng lực xử lý của các tài nguyên mạng. Như vậy sẽ có hai giải pháp: Tăng tài nguyên cho mạng hoặc giảm tải. Có nhiều cách thức làm tăng tài nguyên, chẳng hạn nâng cấp đường truyền, sử dụng thêm các kết nối dự trữ... ; tuy nhiên không phải lúc nào cũng làm được điều này. Do vậy giải pháp khắc phục duy nhất là giảm tải. Có khá nhiều phương pháp để thực hiện điều này: có thể từ chối dịch vụ đối với một số người dùng (không đưa thêm dữ liệu vào mạng nữa), hay yêu cầu tất cả người dùng giảm lượng dữ liệu gửi vào mạng...

Những phương thức kể trên sẽ được nghiên cứu trong các phần sau. Đối với chuyển mạch ảo, những phương thức này được áp dụng ở tầng mạng, còn đối với chuyển mạch gói, những phương thức này lại có thể được cài đặt ở tầng giao vận.

4.9.2 Chính sách ngăn chặn tắc nghẽn



Hình 4.42 Các chính sách ngăn chặn tắc nghẽn ở các tầng khác nhau

Chúng ta bắt đầu xét giải pháp mở. Tư tưởng của chiến lược này là ngăn chặn không cho tắc nghẽn xuất hiện chứ không đợi đến lúc tắc nghẽn xuất hiện mới khắc phục. Mục tiêu này đạt được bằng các áp dụng nhiều chính sách khác nhau ở các tầng khác nhau. Trong Hình 4.42 chúng ta thấy các chính sách ở các tầng liên kết dữ liệu, mạng và giao vận ảnh hưởng đến tắc nghẽn.

Đầu tiên chúng ta xét đến tầng liên kết dữ liệu. Chính sách truyền lại xác định phía gửi sau bao lâu sẽ gửi lại gói tin chưa nhận được biên nhận. Rõ ràng nếu đặt thời gian timeout quá bé và sử dụng thuật toán Go-Back-N thì số lượng gói tin phía gửi đưa vào mạng lớn hơn rất nhiều so với hệ thống có timeout lớn và sử dụng thuật toán Selective Repeat. Một vấn đề liên quan mật thiết là chính sách lưu tạm thời dữ liệu ở phía nhận. Nếu phía nhận loại bỏ tất cả các gói dữ liệu đến không đúng thứ tự, các gói này sẽ được gửi lại

lần nữa và do đó làm tăng tổng số lượng gói tin truyền qua mạng. Rõ ràng thuật toán Selective Repeat có hiệu quả hơn Go-Back-N.

Chính sách biên nhận cũng ảnh hưởng đến tắc nghẽn. Nếu phía nhận gửi biên nhận cho mỗi gói tin nhận được thì số lượng gói tin biên nhận sẽ rất lớn. Tuy nhiên nếu biên nhận được dồn lại và gửi kèm cùng với các thông tin khác, hệ thống cũng tiết kiệm được đường truyền và góp phần làm giảm tắc nghẽn.

Ở tầng mạng, quyết định sử dụng chuyển mạch gói hay chuyển mạch ảo cũng ảnh hưởng đến việc kiểm soát tắc nghẽn vì một vài thuật toán chỉ có thể sử dụng trên mạch ảo. Chính sách hàng đợi và chất lượng dịch vụ liên quan đến việc tại mỗi đường ra (hay đường vào) của router có thể có một hay nhiều hàng đợi. Điều này cũng liên quan đến thứ tự ưu tiên khi xử lý các gói tin. Chính sách loại bỏ gói tin là quy tắc xác định gói tin nào sẽ bị loại bỏ trong trường hợp không đủ bộ đệm. Một chính sách tốt sẽ giúp làm giảm tắc nghẽn.

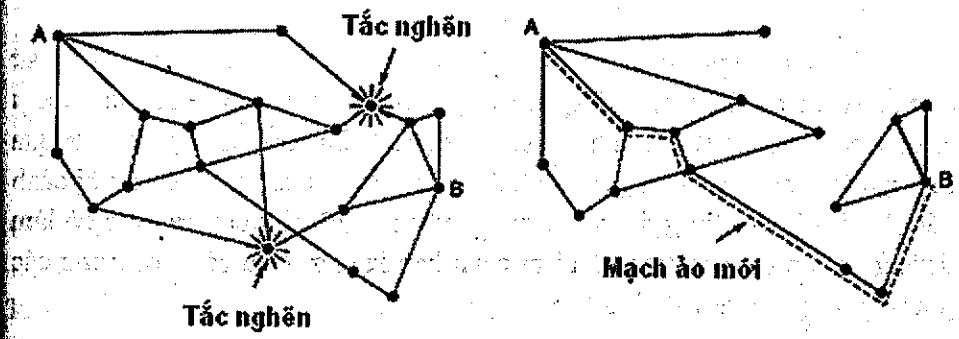
Một thuật toán định tuyến tốt giúp làm giảm tình trạng tắc nghẽn trên mạng bằng cách phân chia đều các gói tin trên tất cả các đường truyền. Một thuật toán không tốt có thể gửi quá nhiều gói tin qua một đường truyền đã bị tắc nghẽn. Cuối cùng, thời gian sống của gói tin (từ khi gửi cho đến khi bị xóa) cũng đóng vai trò quan trọng. Nếu thời gian này quá dài, các gói tin có thể lưu chuyển trên mạng quá lâu, nhưng nếu thời gian này quá dài, gói tin sẽ bị xóa trước khi đến đích, và do đó sẽ phải truyền lại.

Vấn đề ở tầng giao vận cũng tương tự như ở tầng liên kết dữ liệu. Tuy nhiên xác định khoảng thời gian timeout phức tạp hơn rất nhiều vì giữa bên gửi và bên nhận có thể có nhiều router trung gian.

4.9.3 Kiểm soát tắc nghẽn trong mạch ảo

Trong mạch ảo thường sử dụng chiến lược mở: ngăn ngừa không cho tắc nghẽn xảy ra. Ở đây chúng ta sẽ mô tả một vài phương pháp kiểm soát động được sử dụng trong các mạng ảo.

Một kỹ thuật được sử dụng khá phổ biến là kiểm soát việc thiết lập kết nối mới. Khi bắt đầu tắc nghẽn, không mạch ảo nào được thiết lập cho đến khi hết tắc nghẽn. Như vậy, việc thiết lập kết nối ở tầng giao vận sẽ không thể thực hiện được. Ưu điểm của giải pháp này là tính đơn giản và dễ cài đặt. Trong mạng điện thoại truyền thống, khi một tổng đài rơi vào tình trạng quá tải, hệ thống thực hiện kiểm soát việc thiếp lập kết nối mới không cho phép kết nối qua tổng đài này.



Hình 4.43 (a) Mạng bị tắc nghẽn (b) Mạng không có tắc nghẽn do mạch ảo từ A đến B vòng qua khu vực bị tắc nghẽn

Một giải pháp khác là cho phép thiết lập mạch ảo, nhưng định tuyến mạch này không đi qua khu vực bị tắc nghẽn, như ví dụ trong Hình 4.43 Giả sử một nút gắn với router A muốn thiết lập kết nối tới nút gắn với router B. Bình thường, kết nối này sẽ được thực hiện qua một vài router đang bị tắc nghẽn. Để tránh trường hợp này, chúng ta vẽ lại hình trạng mạng như trong Hình 4.43b, loại bỏ các router bị tắc nghẽn và các đường truyền liên quan. Đường đứt nét là tuyến mạch ảo mới tránh khỏi khu vực tắc nghẽn.

Một chiến lược khác liên quan đến mạch ảo là nút sẽ "thương lượng" với hệ thống mạng để đạt được một "thỏa thuận" trước khi thiết lập mạch ảo. Thỏa thuận này bao gồm khối lượng, khuôn dạng và chất lượng dịch vụ của mạch ảo. Để đảm bảo thực hiện đúng thỏa thuận, hệ thống mạng thường thực hiện việc dự trữ tài nguyên dọc theo tuyến đường mạch ảo được thiết lập. Tài nguyên có thể là bảng định tuyến, bộ đệm, băng thông đường truyền của router. Do mỗi mạch ảo đều được đảm bảo những tài nguyên cần thiết, tắc nghẽn không bao giờ xuất hiện.

Quá trình dự trữ tài nguyên này có thể chỉ được thực hiện khi mạng bị tắc nghẽn hay là một thủ tục trong hoạt động bình thường của hệ thống. Nhược điểm của hệ thống này là lãng phí tài nguyên. Đó là cái giá cho việc thực hiện kiểm soát tắc nghẽn.

4.9.4 Kiểm soát tắc nghẽn trong mạng chuyển mạch gói

Trong mạng chuyển mạch gói, các router kiểm soát hiệu suất sử dụng của các đường truyền đi qua. Ví dụ mỗi đường truyền sẽ có một biến u (nằm trong khoảng [0,1]) là hiệu suất sử dụng đường truyền. Nếu u vượt qua một ngưỡng nào đó, đường truyền tương ứng chuyển sang trạng thái bị cảnh báo. Mỗi một gói tin gửi qua đường truyền này sẽ bị kiểm tra xem có làm đường truyền ở trong trạng thái bị cảnh báo không. Nếu có, một trong các phương pháp sau đây sẽ được áp dụng.

Bit cảnh báo

Kiến trúc DECNET thực hiện việc cảnh báo bằng cách thiết lập một bit đặc biệt trong tiêu đề gói tin (giống kiến trúc Frame Relay). Khi gói tin đến đích, thực thể giao vận ở nút đích sẽ sao giá trị bit này vào gói tin biên nhận. Khi nhận được, phía gửi sẽ giảm tốc độ truyền.

Chừng nào còn ở trong trạng thái cảnh báo, router tiếp tục thiết lập giá trị bit cảnh báo. Sau một thời gian, các nút gửi nhận được các gói tin biên nhận có thiết lập bit cảnh báo và phải giảm tốc độ gửi dữ liệu. Quá trình giảm này tiếp tục cho đến khi không nhận được các gói tin có chứa dấu hiệu cảnh báo.

Gói tin báo tắc nghẽn

Thuật toán trình bày trên nhờ phía nhận phản hồi cho phía gửi giảm tốc độ truyền. Tại sao router không thông báo trực tiếp tình trạng tắc nghẽn cho các nút ? Trong giải pháp này, router sẽ gửi một gói tin báo nghẽn cho phía gửi (địa chỉ nhận của gói báo nghẽn được xác định nhờ trường địa chỉ gửi của gói tin bình thường). Gói tin bình thường được đánh dấu (bằng cách

thiết lập một trường nào đó trong tiêu đề gói tin) và chuyển tiếp bình thường tới đích. Mục đích của việc đánh dấu là không phát sinh thêm các gói tin báo nghẽn do gói tin bình thường sinh ra khi đến các router khác.

Khi nhận được gói tin báo nghẽn, nút gửi phải giảm tốc độ truyền tới địa chỉ đích cụ thể đi X%. Có thể các gói tin gửi đến cùng địa chỉ đích này cũng khiến phát sinh ra các gói tin báo nghẽn, nên phía gửi sẽ bỏ qua các gói tin báo nghẽn trong một khoảng thời gian nào đó. Sau đó, phía gửi tiếp tục kiểm tra xem có nhận được gói tin báo nghẽn nào không. Nếu có, nó lại tiếp tục giảm tốc độ gửi vì trên đường truyền đến đích chắc chắn xuất hiện tắc nghẽn. Nút gửi có thể tăng tốc độ gửi trong trường hợp không nhận được gói tin báo nghẽn trong một khoảng thời gian đủ lâu.

Bằng cách điều chỉnh các tham số truyền (chẳng hạn độ lớn cửa sổ), nút có thể làm giảm tốc độ truyền. Thông thường, khi nhận được gói tin báo nghẽn đầu tiên, tốc độ truyền giảm 50%, lần nhận gói tin báo nghẽn thứ hai khiến tốc độ truyền giảm xuống còn 25%... Tuy nhiên tốc độ truyền tăng từ từ để tránh sự tái xuất hiện của tắc nghẽn.

Có khá nhiều phiên bản của thuật toán này. Chẳng hạn một router có thể có nhiều mức ngưỡng khác nhau. Phụ thuộc vào mức ngưỡng nào, router có thể gửi các gói tin báo nghẽn ở các mức độ khác nhau.

4.9.5 Cắt tải

Nếu không phương pháp nào trình bày trên khắc phục hoàn toàn tắc nghẽn, router có thể sử dụng một phương pháp rất mạnh: cắt tải. Khi có quá nhiều gói tin đến router và router không thể xử lý hết, router sẽ xóa bỏ một số gói tin. Điều này cũng tương tự để đảm bảo an toàn cho mạng lưới điện vào những ngày hè (khi nhu cầu điện lớn hơn lượng điện có thể cung cấp), một vài khu vực sẽ bị cắt điện.

Router có thể loại bỏ ngẫu nhiên các gói tin, nhưng thông thường router loại bỏ theo một chiến lược nào đó. Chiến lược này phụ thuộc vào ứng dụng đang chạy. Ví dụ đối với ứng dụng truyền file (FTP), gói tin cũ có

ích hơn gói tin mới, vì chẳng hạn nếu loại bỏ gói tin 6 và giữ lại các gói tin từ 7 đến 10 có thể sẽ khiến tất cả gói tin từ 6-10 phải gửi lại nếu phía nhận không chấp nhận các gói tin đến không theo thứ tự. Như vậy nếu file gồm 12 gói tin, loại bỏ gói 6 sẽ khiến phải truyền lại gói 7 đến 12 nhưng loại bỏ gói 10 thì chỉ cần truyền lại gói 10 đến 12. Tuy nhiên với ứng dụng đa phương tiện thì gói tin mới có ích hơn gói tin cũ.

Đối với nhiều ứng dụng, một số gói tin có tầm quan trọng hơn các gói tin khác. Chẳng hạn với một số thuật toán nén phim đôi khi gửi toàn bộ frame (chứa hình ảnh) trong một gói tin và các gói tin sau chỉ lưu lại các khác biệt so với frame ban đầu. Do vậy, ở đây nếu phải loại bỏ một gói tin thì nên loại bỏ gói tin chứa các gói tin biến đổi chứ không nên loại bỏ gói tin chứa một frame hoàn chỉnh.

Để cài đặt chiến lược loại bỏ thông minh, ứng dụng phải đánh dấu các gói tin ở các độ ưu tiên (thể hiện sự quan trọng) khác nhau. Khi đó, tại các router, các gói tin ít quan trọng sẽ bị loại bỏ trước.

Phát hiện sớm ngẫu nhiên (Random Early Detection)

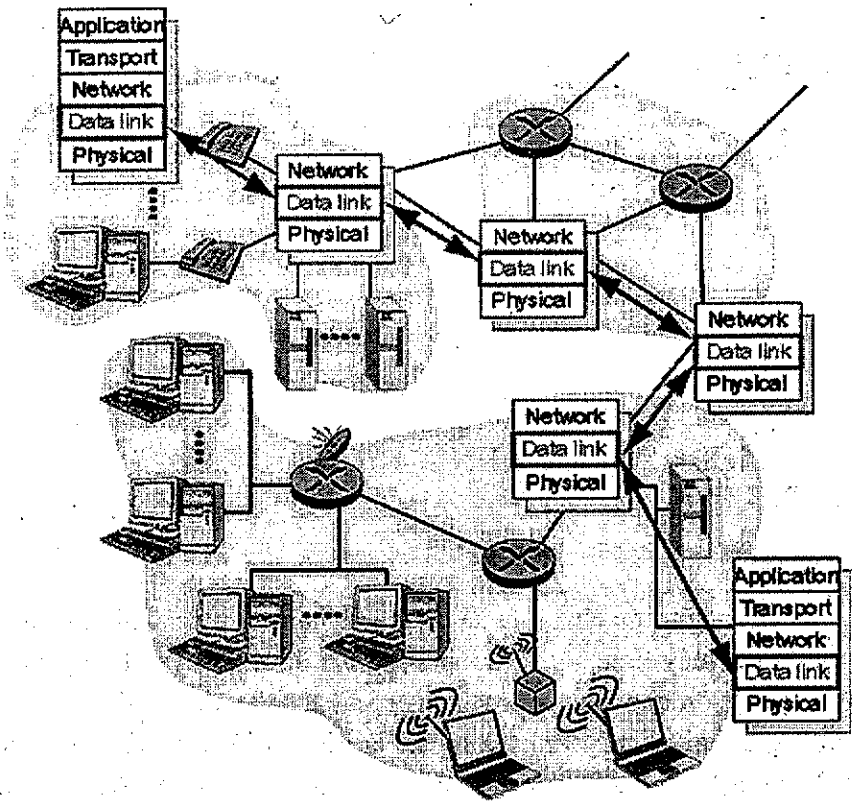
Rõ ràng xử lý tắc nghẽn ngay khi phát hiện ra sẽ có hiệu quả hơn nhiều nếu trì hoãn xử lý rồi sau đó mới khắc phục. Thuật toán thực hiện điều này – RED (Random Early Detection) – thực hiện loại bỏ các gói tin trước khi router sử dụng hết các bộ đệm. Trong một số giao thức ở tầng giao vận (chẳng hạn TCP), phía gửi khi bắt đầu thấy mất gói tin (không nhận được biên nhận) sẽ bắt đầu giảm tốc độ gửi. Ý tưởng của điều này là TCP được sử dụng trên mạng có dây (tỷ lệ lỗi thấp) nên nếu gói tin bị mất thì nguyên nhân lớn nhất là thiếu bộ đệm ở router. Điều này có thể được sử dụng để tránh tắc nghẽn.

Nếu router ngăn ngừa trước khi tắc nghẽn thực sự xảy ra thì tắc nghẽn sẽ không xảy ra ở mức độ nghiêm trọng. Để xác định khi nào bắt đầu xảy ra tắc nghẽn, độ dài hàng đợi trung bình của mỗi đường truyền được gán một mức ngưỡng nào đó. Nếu vượt quá ngưỡng này, chắc chắn tắc nghẽn bắt đầu xuất hiện và router thực hiện loại bỏ gói tin.

Chương 5
TẦNG LIÊN KẾT DỮ LIỆU

5.1 CÁC KHÁI NIỆM CHUNG, DỊCH VỤ CỦA TẦNG DATALINK

Ở chương trước chúng ta đã biết tầng mạng cung cấp dịch vụ truyền thông giữa hai máy tính. Ví dụ trong Hình 5.1 đường truyền thông này bắt đầu từ máy nguồn qua lần lượt các router và kết thúc ở máy đích. Để thuận tiện, chúng ta coi cả máy tính và router là các **nút (node)** vì ở đây nút là router hay máy tính không phải là vấn đề quan trọng và kênh truyền thông kết nối giữa hai nút liền kề trên toàn bộ đường truyền thông được gọi là **đường liên kết,** hay **đường truyền (link)**. Để gói dữ liệu (datagram) đi từ máy tính nguồn tới máy tính đích, gói dữ liệu phải được chuyển trên mỗi đường liên kết. Chương này tập trung vào tầng liên kết dữ liệu với nhiệm vụ truyền gói dữ liệu trên một đường liên kết (đường truyền vật lý). Đầu tiên chúng ta sẽ phân loại và nghiên cứu những dịch vụ của tầng liên kết dữ liệu. Từ phần 5.2 đến phần 5.4 là các nguyên lý quan trọng của những giao thức cung cấp các dịch vụ này (ví dụ nhận lỗi và sửa lỗi, giao thức đa truy cập được sử dụng để chia sẻ đường truyền vật lý duy nhất giữa các nút và địa chỉ mức liên kết dữ liệu). Chúng ta sẽ thấy nhiều kiểu công nghệ kết nối khác nhau được sử dụng để nối hai nút, từ phần 5.5 đến phần 5.10 chúng ta sẽ nghiên cứu chi tiết kiến trúc và giao thức của các kiểu kết nối.



Hình 5.1 Vị trí của tầng liên kết dữ liệu

5.1.1 Những dịch vụ của tầng liên kết dữ liệu

Giao thức tầng liên kết dữ liệu được sử dụng để truyền gói dữ liệu trên một môi trường vật lý. Giao thức tầng liên kết dữ liệu định nghĩa khuôn dạng đơn vị dữ liệu trao đổi giữa các nút ở mỗi đầu của đường truyền, cũng như những công việc các nút thực hiện khi nhận và gửi những đơn vị dữ liệu này. Trong chương 1 chúng ta đã biết rằng đơn vị dữ liệu của tầng liên kết dữ liệu là frame và mỗi frame tầng liên kết dữ liệu chứa một gói dữ liệu tầng mạng. Công việc của giao thức tầng liên kết dữ liệu khi gửi và nhận frame gồm: phát hiện lỗi, truyền lại, điều khiển lưu lượng và truy cập ngẫu nhiên.

Giao thức tầng liên kết dữ liệu gồm: Ethernet, token ring, FDDI và PPP; đôi khi ATM và frame relay có thể cũng được coi là giao thức tầng liên kết dữ liệu. Chúng ta sẽ nghiên cứu chi tiết những giao thức này trong nửa chương sau.

Nếu nhiệm vụ của tầng mạng là chuyển gói dữ liệu của tầng giao vận từ máy gửi tới 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. Một đặc điểm quan trọng của tầng liên kết dữ liệu là gói dữ liệu tầng mạng có thể được xử lí bởi các giao thức liên kết dữ liệu khác nhau trên đường truyền. Ví dụ, gói dữ liệu này có thể được chuyển bởi giao thức Ethernet trên đường truyền đầu tiên, bởi PPP ở đường truyền cuối cùng và frame relay trên các đường truyền ở giữa. Chú ý quan trọng là các giao thức liên kết dữ liệu khác nhau có thể cung cấp những dịch vụ khác nhau. Giao thức tầng liên kết dữ liệu không nhất thiết phải cung cấp dịch vụ truyền tin cậy. Vì vậy tầng mạng phải tính đến khả năng hoạt động trên nhiều kiểu dịch vụ liên kết dữ liệu khác nhau.

Để hiểu rõ quan hệ của tầng liên kết dữ liệu với tầng mạng như thế nào xét ví dụ sau. Giả sử một đại lý du lịch phải sắp xếp cho khách du lịch đi từ KS Hilton, Hà Nội đến Ngọ Môn, Huế. Đầu tiên khách du lịch sẽ đi xe buýt đến sân bay Nội Bài, bay bằng máy bay của Hàng không Việt Nam đến Huế và đi taxi từ sân bay Phú Bài, Huế đến Ngọ Môn. Sau khi đại lý du lịch tiến hành đặt chỗ, thì chính công ty xe buýt chịu trách nhiệm đưa khách du lịch từ Hilton đến Nội Bài, Hàng không Việt Nam chịu trách nhiệm từ Nội Bài đến sân bay Phú Bài và hãng taxi từ sân bay Huế đến Ngọ Môn. Mỗi một trong ba chặng này được xem là một lần chuyển trực tiếp giữa các "nút" kế tiếp. Rõ ràng ba chặng đường sẽ được các công ty khác nhau phụ trách. Trong ví dụ trên có thể xem khách du lịch là datagram (gói tin của tầng mạng), chặng đường tương ứng với một đường truyền vật lý, phương tiện đi lại trên chặng đường là giao thức cho đường truyền vật lý và đại lý du lịch chính là giao thức định tuyến ở tầng mạng.

Dịch vụ cơ bản của bất kì tầng liên kết dữ liệu nào là chuyển gói dữ liệu của tầng mạng giữa hai nút kế tiếp, song cụ thể dịch vụ này được thực hiện như thế nào lại phụ thuộc vào giao thức tầng liên kết dữ liệu sử dụng

trên đường truyền đó. Nói chung giao thức tầng liên kết dữ liệu có thể cung cấp những dịch vụ sau:

Đóng gói dữ liệu (frame) và truy cập đường truyền (link access): Phần lớn các giao thức tầng liên kết dữ liệu đặt gói dữ liệu tầng mạng vào trong gói dữ liệu tầng liên kết dữ liệu (frame) trước khi truyền lên trên đường truyền. Frame gồm trường dữ liệu là gói dữ liệu của tầng mạng cùng với một số trường tiêu đề khác. Chú ý frame có thể có cả trường tiêu đề đầu và cuối (header và trailer). Giao thức tầng liên kết dữ liệu xác định khuôn dạng của frame cũng như giao thức truy cập kênh truyền (cách thức truyền). Với đường truyền kiểu point-to-point có một phía gửi và một phía nhận thì giao thức truy cập đường truyền khá đơn giản (gần như không tồn tại) - bên gửi có thể gửi frame bất cứ lúc nào đường truyền rỗi. Trường hợp phức tạp hơn xảy ra khi nhiều nút cùng chia sẻ đường truyền duy nhất: đây là vấn đề đa truy cập. Giao thức đa truy cập được nghiên cứu trong phần 5.3. Chúng ta sẽ thấy các giao thức liên kết dữ liệu khác nhau sẽ có những khuôn dạng dữ liệu khác nhau (chú ý gói dữ liệu của tầng liên kết dữ liệu được gọi là frame). Trong phần 5.3, chúng ta sẽ thấy tiêu đề frame thường chứa trường **địa chỉ vật lí** của nút, (địa chỉ này khác địa chỉ IP).

Dịch vụ truyền tin cậy. Nếu cung cấp dịch vụ truyền tin cậy, giao thức tầng liên kết dữ liệu bảo đảm chuyển chính xác gói dữ liệu tầng mạng trên một đường truyền. Trong chương 3 chúng ta thấy rằng giao thức tầng giao vận (TCP) cũng cung cấp dịch vụ truyền tin cậy. Tương tự như trong tầng giao vận, truyền tin cậy ở tầng liên kết dữ liệu được thực hiện qua cơ chế báo nhận và truyền lại (xem phần 3.4). Dịch vụ truyền tin cậy ở tầng liên kết dữ liệu thường được sử dụng trên đường truyền có tỉ lệ lỗi cao (ví dụ trên đường truyền không dây). Mục đích là sửa lỗi ngay trên đường truyền bị lỗi chứ không phải truyền lại dữ liệu từ thiết bị gửi tới thiết bị nhận bởi giao thức tầng giao vận hoặc tầng ứng dụng. Tuy nhiên tầng liên kết dữ liệu không cần cung cấp dịch vụ truyền tin cậy cho các đường truyền ít lỗi (ví dụ cáp quang). Vì vậy phần lớn các giao thức tầng liên kết dữ liệu phổ biến không cung cấp dịch vụ truyền tin cậy.

Kiểm soát lưu lượng. Khả năng lưu trữ tạm thời (buffer) các frame tại các nút trên mỗi phía của đường truyền không phải là vô hạn. Đây sẽ là

vấn đề khi tốc độ tới của các frame nhanh hơn tốc độ nút nhận có thể xử lí được. Nếu không kiểm soát lưu lượng, bộ đệm phía nhận có thể bị tràn và frame sẽ bị mất. Giống như tầng giao vận, tầng liên kết dữ liệu cung cấp cơ chế kiểm soát lưu lượng để ngăn chặn phía phát gửi quá khả năng nhận của phía thu.

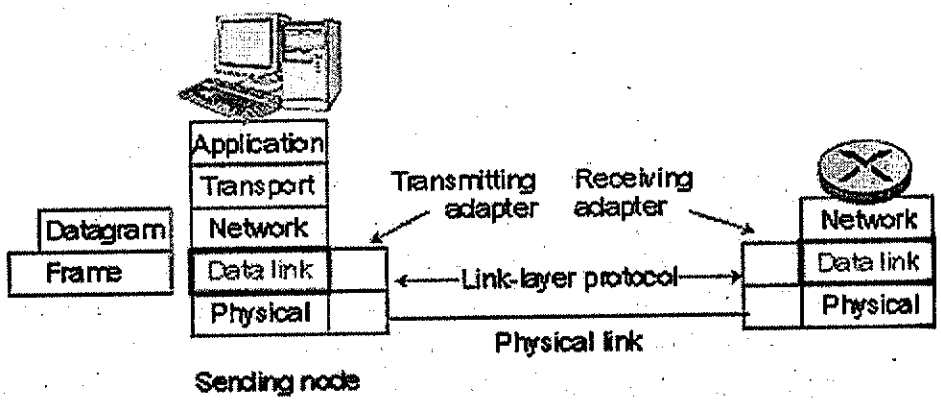
Phát hiện lỗi. Nút nhận có thể nhận bit 0 trong khi phía gửi gửi bit 1 hay ngược lại. Nguyên nhân bit bị lỗi có thể do tín hiệu bị suy hao hay nhiễu điện từ. Nhiều giao thức tầng liên kết dữ liệu cung cấp cơ chế phát hiện lỗi. Điều này được thực hiện bằng cách phía gửi sẽ thiết lập một số bit phát hiện lỗi trong frame và phía nhận thực hiện việc kiểm tra lỗi. Dịch vụ phát hiện lỗi rất phổ biến trong nhiều giao thức tầng liên kết dữ liệu. Trong chương 3 và chương 4 chúng ta thấy tầng giao vận và tầng mạng trên Internet cũng có khả năng phát hiện lỗi. Tuy nhiên phát hiện lỗi trong tầng liên kết dữ liệu phức tạp hơn rất nhiều và do vậy thường được triển khai bằng phần cứng.

Sửa lỗi. Sửa lỗi cũng tương tự phát hiện lỗi. Tuy nhiên không chỉ phát hiện được lỗi có khả năng mà phía nhận còn có khả năng xác định chính xác vị trí lỗi xuất hiện trong frame (và do đó có thể sửa được những lỗi này). Phát hiện và sửa lỗi được nghiên cứu trong phần 5.2.

Bán song công và song công (Half duplex, full duplex). Trong chế độ truyền song công, hai phía của đường truyền có thể đồng thời truyền dữ liệu. Trong chế độ truyền bán song công, tại một thời điểm thiết bị không thể cùng truyền và nhận.

Như nói trên, nhiều dịch vụ của tầng liên kết dữ liệu giống dịch vụ của tầng giao vận, chẳng hạn cả hai tầng đều có dịch vụ truyền tin cậy. Mặc dù cơ chế thực hiện dịch vụ truyền tin cậy ở cả hai tầng giống nhau (xem phần 3.4), nhưng hai dịch vụ truyền tin cậy này không giống nhau. Giao thức tầng giao vận cung cấp dịch vụ truyền tin cậy giữa hai tiến trình trên cơ sở đầu cuối (end-to-end). Giao thức tầng liên kết dữ liệu cung cấp dịch vụ truyền tin cậy giữa hai nút có một đường truyền vật lý trực tiếp. Tương tự với dịch vụ kiểm soát lưu lượng và phát hiện lỗi.

5.1.2 Bộ điều hợp (Adapter)

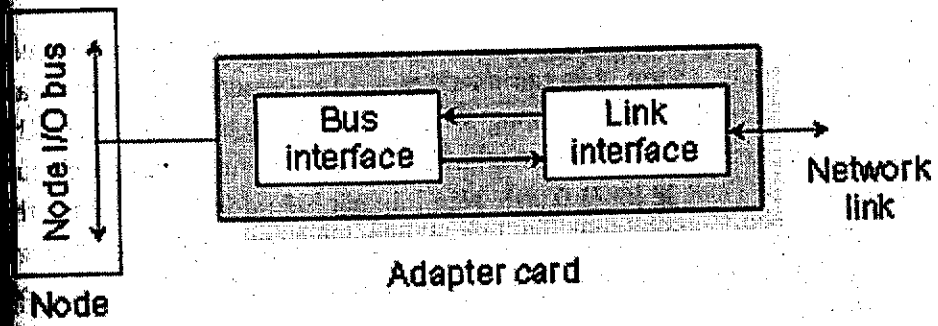


Hình 5.2 Tầng liên kết dữ liệu được triển khai tại adapter

Với phần lớn đường truyền, giao thức tầng liên kết dữ liệu được triển khai trên **adapter**. Adapter là bo mạch (hoặc card PCMCIA) có RAM, chip DSP, giao diện ghép nối (interface) với bus máy tính và giao diện ghép nối với đường truyền. Adapter cũng thường được coi là card giao tiếp mạng hay gọi tắt là card mạng (**NIC - Network Interface Card**). Như minh họa trên Hình 5.2, tầng mạng trên nút gửi (máy tính hoặc router) chuyển gói dữ liệu tầng mạng (datagram) xuống adapter để gửi sang phía bên kia đường truyền. Adapter đặt datagram trong frame, sau đó truyền frame qua đường truyền. Adapter phía bên kia (phía nhận) nhận frame, lấy và chuyển datagram lên tầng mạng. Nếu giao thức tầng liên kết dữ liệu cung cấp dịch vụ phát hiện lỗi thì adapter gửi đặt thêm một số bit phát hiện lỗi và adapter nhận thực hiện việc kiểm tra lỗi. Nếu giao thức tầng liên kết dữ liệu cung cấp dịch vụ truyền tin cậy thì những kĩ thuật cho dịch vụ truyền tin cậy (ví dụ số thứ tự, bộ định thời, biên nhận) được cài đặt ngay trên adapter. Nếu giao thức tầng liên kết dữ liệu cung cấp dịch vụ truy cập ngẫu nhiên (phần 5.3) thì giao thức này cũng được triển khai trên adapter.

Adapter là đơn vị bán tự trị. Ví dụ, adapter có thể nhận frame, xác định liệu frame có bị lỗi không và nếu có thì loại bỏ frame mà không thông

ão cho thiết bị. Khi nhận được frame đến từ môi trường vật lý (chẳng hạn rd mạng), adapter chỉ thông báo cho thiết bị (chính xác là CPU của thiết) khi adapter muốn chuyển gói dữ liệu datagram trong frame tới tầng mạng ủa thiết bị. Việc này sẽ được thực hiện thông qua ngắt phần cứng nterrupt). Tương tự, khi nút gửi gói dữ liệu đến adapter, có thể coi adapter ược uỷ quyền để chuyển gói dữ liệu sang nút kế tiếp. Mặt khác, adapter hông là đơn vị tự trị hoàn toàn. Mặc dù chúng ta coi adapter là "hộp đen" ư minh họa trên Hình 5.3, adapter vẫn nằm trong máy tính, dùng chung guồn điện và bus - do đó vẫn nằm dưới sự điều khiển của máy tính.



Hình 5.3 Kiến trúc Adapter

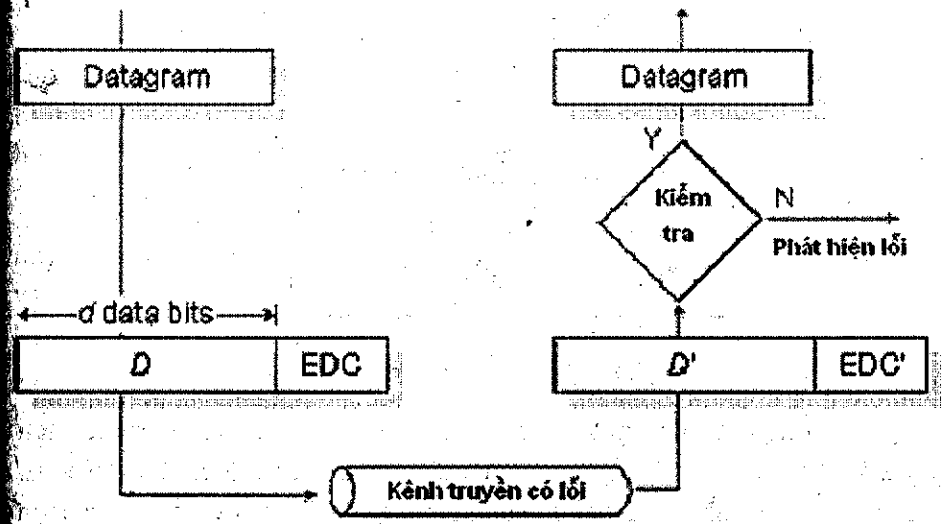
Thành phần chính của adapter là giao diện ghép nối bus và giao diện hép nối đường truyền. Giao diện bus chịu trách nhiệm truyền thông với nút hứa adapter (chính xác hơn là với CPU). Nó truyền dữ liệu cùng thông tin iều khiển giữa nút và card mạng. Giao diện đường truyền (link interface) ó trách nhiệm triển khai giao thức tầng liên kết dữ liệu. Bên cạnh chức năng đóng gói (framing) và bóc tách (de-framing) gói dữ liệu, card mạng có hể cung cấp dịch vụ phát hiện lỗi, truy cập ngẫu nhiên và các chức năng khác của tầng liên kết dữ liệu. Nó chứa các mạch truyền và nhận (circuitry). Với những công nghệ phổ biến ở tầng liên kết dữ liệu – ví dụ như Ethernet, giao diện đường truyền được triển khai trên các chip. Chip được sản xuất đại trà và bán rộng rãi, do đó, adapter Ethernet rẻ - thường không quá 10 USD.

5.2 KỸ THUẬT PHÁT HIỆN VÀ SỬA LỖI

Các đường điện thoại bình thường (truyền tín hiệu tương tự trên đường dây đồng) có tỷ lệ lỗi tương đối cao. Việc thay thế hệ thống này rất khó thực thi vì quá tốn kém. Bên cạnh đó mạng không dây có tỷ lệ lỗi cao hơn rất nhiều so với mạng có dây. Do vậy nếu cơ sở truyền thông của chúng ta vẫn dựa trên mạng điện thoại bình thường cũng như việc mạng không dây ngày càng phổ biến thì các nhà nghiên cứu vẫn phải nghiên cứu các phương thức để xử lý lỗi trên đường truyền.

Về đặc điểm xuất hiện lỗi, người ta thấy rằng lỗi có xu hướng xuất hiện theo cụm chứ không phải các lỗi riêng lẻ. Tại sao đặc điểm này quan trọng khi tìm giải pháp phòng chống lỗi ? Chúng ta xét thử ví dụ sau. Khi truyền, dữ liệu thường được gửi theo khối. Giả sử kích thước trung bình của khối là 1000 bit và tỷ lệ lỗi là 0.001. Nếu lỗi xuất hiện độc lập với nhau thì trung bình mỗi khối đều có lỗi. Nhưng nếu lỗi xuất hiện thành cụm 100 bit thì trung bình trong 100 khối sẽ có từ 1 đến 2 khối bị lỗi. Nói chung lỗi theo cụm khó sửa hơn lỗi riêng lẻ.

Một trong các dịch vụ mà tầng liên kết dữ liệu cung cấp là **phát hiện (detection) và sửa (correct) lỗi** ở mức bit – cho phép phát hiện và trong một số trường hợp có thể sửa các bit bị lỗi trong frame của tầng liên kết dữ liệu gửi giữa hai nút kế tiếp. Trong chương 3 chúng ta đã thấy dịch vụ phát hiện lỗi cũng được triển khai ở tầng giao vận. Trong phần này, chúng ta sẽ nghiên cứu một vài kỹ thuật đơn giản nhất trong việc phát hiện và sửa lỗi bit. Ở đây chúng ta chỉ trình bày hết sức sơ lược với mục tiêu giúp độc giả hình dung ra một vài kỹ thuật được áp dụng trong một số giao thức tầng liên kết dữ liệu thông dụng.Hình 5.4 minh hoạ ý tưởng của việc triển khai dịch vụ phát hiện lỗi. Tại nút gửi, dữ liệu D được sử dụng để xác định các bit phát hiện và sửa lỗi EDC. Dữ liệu cần bảo vệ không chỉ gồm gói dữ liệu datagram tầng mạng chuyển xuống mà còn các trường khác trong tiêu đề frame như trường địa chỉ, trường số thứ tự... Cả D và EDC cùng được gửi đến nút nhận trong cùng một frame. Bên kia sẽ nhận được chuỗi bit là D' và EDC'. Chú ý rằng, D' và EDC' có thể không giống D và EDC vì trên kênh truyền có thể có lỗi.

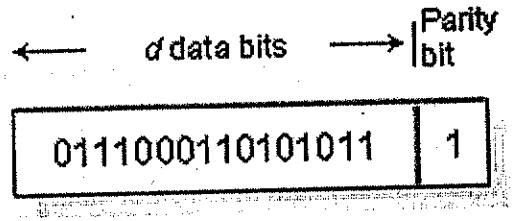


Hình 5.4 Kiểm tra và phát hiện lỗi

Phía nhận phải xác định liệu D' có chính là D được phía bên kia gửi hay không trong trường hợp nhận được cả D' và EDC'. Tại phía nhận, kết quả kiểm tra frame có lỗi không rất quan trọng (xem Hình 5.4). Chú ý rằng ở đây chúng ta xác định có ***phát hiện được lỗi*** không chứ không phải là ***có lỗi hay không***). Kỹ thuật phát hiện và sửa lỗi không phải luôn luôn cho phép phía nhận xác định được có lỗi hay không: sẽ có những trường hợp sử dụng bit phát hiện lỗi nhưng không có khả năng phát hiện khi xuất hiện lỗi, nghĩa là phía nhận sẽ không biết rằng thông tin nhận được bị lỗi. Hậu quả là phía nhận có thể chuyển gói dữ liệu bị lỗi lên tầng mạng, hoặc không biết rằng nội dung của một vài trường nào đó trong tiêu đề frame bị lỗi. Vì thế chúng ta mong muốn lựa chọn phương pháp có xác suất không phát hiện được lỗi nhỏ nhất có thể. Nói chung, các kĩ thuật phát hiện và sửa lỗi tinh vi (xác suất không phát hiện được lỗi khi có lỗi rất nhỏ) thường đòi hỏi nhiều thời gian tính toán (xác định EDC từ D), số lượng bit dư thừa lớn (EDC lớn) và tại phía nhận việc kiểm tra mất nhiều thời gian.

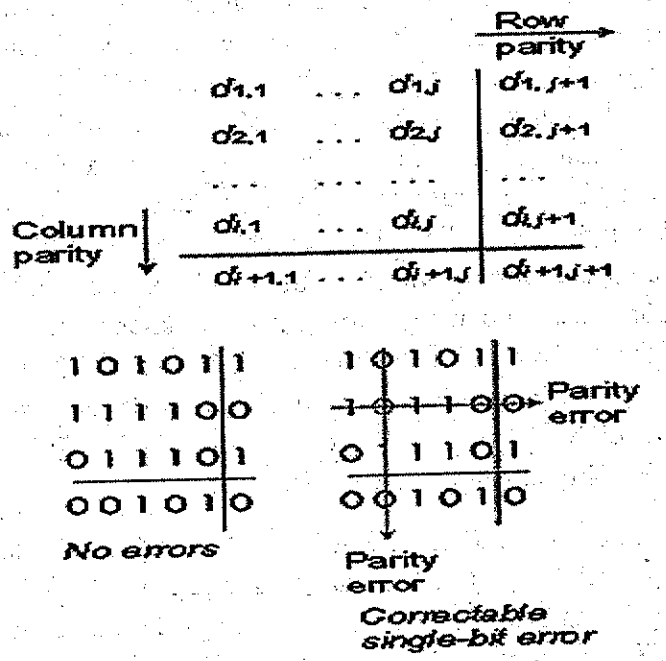
Bây giờ chúng ta sẽ xem xét một số kỹ thuật phát hiện lỗi đơn giản: bit chẵn lẻ (để minh họa ý tưởng của công việc); tính tổng (checksum) - được sử dụng ở tầng giao vận và bit dư thừa vòng - được sử dụng ở tầng liên kết dữ liệu. Kế tiếp chúng ta sẽ trình bày về mã Hamming, một phương pháp đơn giản nhưng có khả năng sửa được lỗi.

5.2.1 Kiểm tra tính chẵn lẻ



Hình 5.5 Kiểm tra tính chẵn lẻ

Phương pháp đơn giản nhất để phát hiện lỗi là sử dụng một **bit chẵn lẻ (parity bit)**. Giả sử thông tin D được gửi trong Hình 5.5 có d bit. Nếu sử dụng bit chẵn lẻ chẵn, bên gửi bổ sung một bit và giá trị bit này được chọn sao cho tổng số số bit 1 trong d+1 bit (d bit thông tin gốc D và một bit chẵn lẻ) là chẵn. Với bit chẵn lẻ lẻ, giá trị bit chẵn lẻ được chọn sao cho tổng số số bit 1 là một số lẻ. Hình 5.5 minh hoạ ví dụ bit chẵn lẻ chẵn, và bit chẵn lẻ (EDC) được đặt trong một trường khác với trường dữ liệu.



Hình 5.6 Kiểm tra chẵn lẻ hai chiều

Việc kiểm tra tại phía nhận cũng tương đối đơn giản. Phía nhận chỉ cần đếm số bit 1 trong d+1 bit nhận được. Nếu tổng số đếm được là một số lẻ trong khi sử dụng số chẵn lẻ chẵn thì phía nhận biết rằng ít nhất đã xuất hiện một bit bị lỗi. Chính xác hơn, tổng số các bit bị lỗi là một số lẻ.

Điều gì sẽ xảy nếu số bit bị lỗi là một số chẵn? Trong trường hợp này rõ ràng bên nhận không phát hiện được lỗi. Nếu xác suất bit bị lỗi thấp và giả định các bit bị lỗi xuất hiện độc lập với nhau thì khả năng một gói dữ liệu có nhiều bit bị lỗi sẽ cực kỳ thấp. Trong trường hợp này, có thể chỉ cần sử dụng một bit chẵn lẻ. Tuy nhiên, các thực nghiệm đã chỉ ra rằng lỗi không xuất hiện độc lập mà thường theo từng "cụm" (burst). Khi đó xác suất lỗi không bị phát hiện trong frame sử dụng một bit chẵn lẻ là 50%. Rõ ràng cần phải có phương pháp khác mạnh hơn. Nhưng trước khi nghiên cứu các phương pháp phát hiện lỗi được sử dụng trong thực tế, chúng ta hãy xét một phương pháp khác dựa trên bit chẵn lẻ có khả năng sửa được lỗi.

Hình 5.6 minh họa việc cải tiến phương pháp bit chẵn lẻ thông qua mảng hai chiều. Ở đây d bit trong D được sắp xếp vào bảng i dòng và j cột. Sau đó bên gửi xác định giá trị bit chẵn lẻ cho tất cả các dòng và cột. (i+j+1) bit chẵn lẻ được tạo ra này sẽ là các bit phát hiện và sửa lỗi của frame.

Bây giờ, giả sử rằng trên đường truyền có một bit duy nhất trong d bit thông tin gốc bị lỗi. Với phương pháp chẵn lẻ hai chiều, sẽ xuất hiện mâu thuẫn trong cả hàng và cột chứa bit bị lỗi. Khi đó phía nhận không những chỉ phát hiện có lỗi mà còn có thể xác định được vị trí bit bị lỗi (và do đó sửa được) thông qua chỉ số hàng và cột của bit đó.

Hình 5.6 chỉ ra ví dụ, bit có giá trị 1 tại vị trí (1,1) bị lỗi và bị chuyển thành 0. Lỗi mà sẽ được phía nhận phát hiện và sửa lại. Mặc dù, ở đây chúng ta chỉ quan tâm đến d bit thông tin, nhưng một bit lỗi trong các bit chẵn lẻ kiểm tra cũng sẽ bị phát hiện. Phương pháp này cũng có thể phát hiện được (nhưng không sửa được) khi có 2 bit bị lỗi trong gói dữ liệu.

Khả năng phía nhận vừa phát hiện vừa sửa được lỗi được gọi là FEC (Forward Error Correction). Những kĩ thuật này thường được sử dụng phổ biến trong các thiết bị lưu trữ âm thanh như đĩa CD nhạc. Trong môi trường mạng, kĩ thuật FEC có thể được sử dụng một mình, hoặc cùng với kĩ thuật

ARQ đã được nghiên cứu trong chương 3. Ưu điểm của kỹ thuật FEC là cho phép phía gửi không phải truyền lại (do phía nhận sửa được các thông tin bị lỗi). Có lẽ quan trọng hơn là phía nhận sửa được lỗi ngay khi nhận được. Điều này tránh thời gian đợi khi phía gửi phải truyền lại gói dữ liệu. Ưu điểm này được áp dụng trong các ứng dụng thời gian thực.

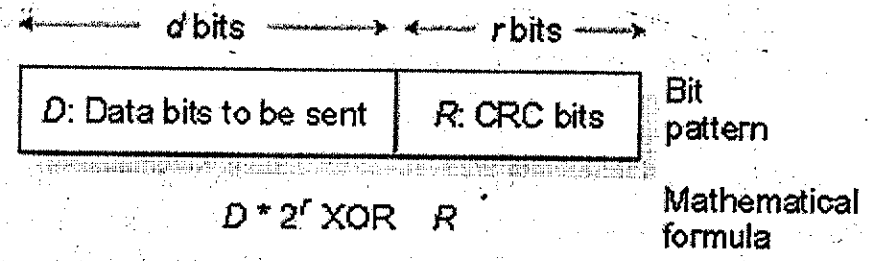
5.2.2 Phương pháp tính tổng kiểm tra (checksum)

Trong kỹ thuật checksum, d bit dữ liệu trong Hình 5.4 được xem là dãy liên tiếp các số nguyên có độ dài k bit. Trong kỹ thuật checksum đơn giản, người ta tính tổng tất cả các số nguyên k bit này và sử dụng kết quả tính được làm các bit phát hiện lỗi. Phương pháp Internet checksum dựa trên hướng tiếp cận này - luồng dữ liệu được coi dãy liên tiếp các số nguyên 16 bit và Internet checksum là giá trị bù một của tổng các số nguyên 16 bit. Như đã trình bày trong phần 3.3.2, phía nhận tính lại checksum trên dữ liệu nhận được và kiểm tra xem nó có trùng với checksum trong gói dữ liệu nhận được hay không. RFC 1071 nêu chi tiết thuật toán Internet checksum cũng như phương thức triển khai. Trong giao thức TCP/IP, Internet checksum được tính toán trên tất cả các trường (kể cả trường tiêu đề và dữ liệu). Trong những giao thức khác, ví dụ như XTP [Strayer 1992], có cả checksum cho các trường tiêu đề và checksum cho toàn bộ gói dữ liệu.

McAuley [McAuley 1994] và Feldmeier [Feldmeier 1995] mô tả cách triển khai bằng phần mềm việc tính toán checksum.

5.2.3 Kiểm tra dư thừa vòng (CRC)

Kĩ thuật phát hiện lỗi được sử dụng rộng rãi trên mạng máy tính ngày nay dựa trên mã CRC. Mã CRC còn được gọi là mã đa thức vì có thể xem dãy các bit được gửi như một đa thức với hệ số nhận nhận giá trị 0 hoặc 1 và thao tác trên dãy bit này giống như thực hiện phép toán trên đa thức.



Hình 5.7 Mã dư thừa vòng

Mã CRC hoạt động như sau. Giả sử phía gửi muốn gửi d bit dữ liệu D. Đầu tiên hai bên phải thống nhất trước **đa thức sinh (generator)** ký hiệu là G có (r+1) bit. Bit có trọng số cao nhất của G phải nhận giá trị 1. Ý tưởng chính của mã CRC được minh họa trên Hình 5.7. Căn cứ vào dữ liệu nguyên thủy D, bên gửi sẽ xác định dữ liệu dư thừa R gồm r bit. Sau đó ghép R với D thu được (d+r) bit. R được chọn sao cho đa thức ứng với (d+r) bit này chia hết cho G. Phép chia ở đây được thực hiện theo module 2. Phía nhận thực hiện quá trình kiểm tra lỗi CRC khá đơn giản: chia d+r bit nhận được cho G. Nếu phần dư khác 0 thì phía nhận xác định xuất hiện lỗi, nếu không dữ liệu được chấp nhận là đúng.

Tất cả các tính toán trên CRC được thực hiện theo module 2 nhưng không nhớ trong phép cộng và không mượn trong phép trừ. Điều này có nghĩa là phép cộng giống phép trừ và cả hai tương đương với việc thực hiện phép XOR trên các toán hạng. Ví dụ:

1011 XOR 0101 =1110

1001 XOR 1101 =0100

Tương tự, chúng ta cũng có:

1011 - 0101 = 1110

1001 - 1101 = 0100

Phép nhân và phép chia cũng giống như thuật toán trên cơ số 2 nhưng các phép cộng và trừ đều không nhớ. Giống như khi thao tác trên các số nhị phân, phép nhân với 2^k là dịch sang trái k vị trí. Do vậy với D và R, kết quả $D*2^r$ XOR R là d+r bit minh họa trên Hình 5.7. Chúng ta sẽ sử dụng các đặc điểm đại số của mẫu d+r bit này trong phần trình bày dưới đây.

Vấn đề quan trọng là làm sao bên gửi xác định được R. Chúng ta muốn xác định R sao cho tồn tại n thỏa mãn:

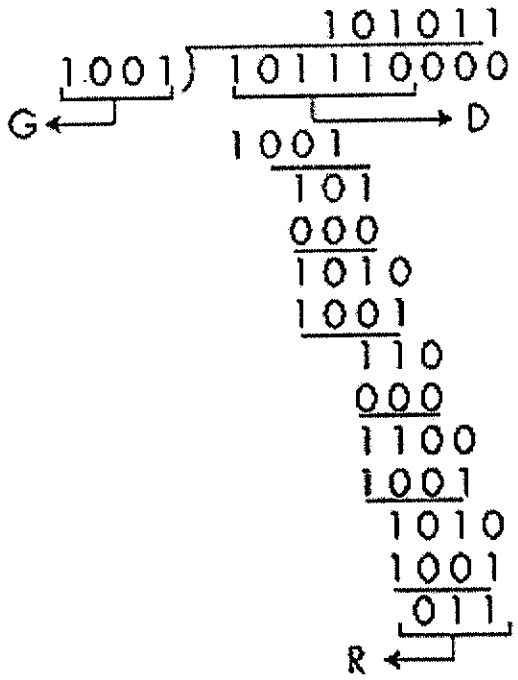
$$D * 2^r \text{ XOR } R = nG$$

Nghĩa là, chúng ta muốn chọn R sao cho G chia cho $D*2^r$ XOR R không có số dư. Nếu chúng ta thực hiện phép XOR R (là phép cộng theo module 2 không nhớ) cả hai vế của phương trình trên, chúng ta nhận được:

$$D * 2^r = nG \text{ XOR } R$$

Đẳng thức này cho chúng ta biết nếu chúng ta chia $D*2^r$ cho G, giá trị phần dư chính là R. Nói cách khác, có thể xác định R như sau:

$$R = \text{số dư } D * 2^r / G$$



Hình 5.8 Ví dụ tính toán CRC

Hình 5.8 minh hoạ kết quả tính toán cho trường hợp D=101110, d=6, G=1001, r=3. Chín bit được truyền trong trường hợp này là 101110 011.

Các chuẩn quốc tế định nghĩa các đa thức sinh (G) 8, 12, 16, 32 bit. CRC 8 bit được sử dụng trong 5 byte tiêu đề của tế bào ATM. CRC-32 trong nhiều giao thức sử dụng đa thức sinh sau:

$$G_{CRC-32} = 100000100110000010001110110110111$$

Các chuẩn CRC có thể phát hiện lỗi cụm với độ lớn nhỏ hơn r+1 bit hay số các bit bị lỗi là một số lẻ. Hơn nữa với một số giả định nào đó, lỗi cụm lớn hơn r+1 bit có thể được phát hiện với xác suất $1-0.5^r$. Lí thuyết mã CRC và thậm chí những mã có tính năng mạnh hơn vượt quá phạm vi của cuốn sách này.

5.2.4 Sửa lỗi bằng mã Hamming

Với môi trường truyền có tỷ lệ lỗi thấp, thì phát hiện được lỗi và sau đó yêu cầu truyền lại sẽ đem lại hiệu quả cao hơn. Để hiểu nguyên nhân điều này, chúng ta xét ví dụ sau. Giả sử trên môi trường truyền các lỗi xuất hiện riêng lẻ và tỷ lệ lỗi là 10^{-6} bit. Để sửa được lỗi cho các khối dữ liệu 1000 bit, chúng ta phải cung cấp thêm 10 bit kiểm tra. Như vậy tổng số lượng bit dư thừa là 10000 bit. Còn nếu chỉ để phát hiện khối có lỗi hay không, trong mỗi khối chỉ cần thêm 1 bit chẵn lẻ, và khi phát hiện được lỗi thì phía nhận yêu cầu bên kia gửi lại (gửi thêm 1 khối 1001 bit), nên tổng lượng bit dư thừa chỉ là 2001 - nhỏ hơn rất nhiều so với 10000. Do đó trong môi trường kiểu này, phát hiện lỗi và yêu cầu truyền lại hiệu quả hơn là trực tiếp sửa được lỗi.

Các phương pháp sửa lỗi được sử dụng chủ yếu trên kênh truyền không dây – là môi trường có tỷ lệ lỗi rất cao so với môi trường cáp đồng trục hay cáp quang.

Gửi lặp

Gửi lặp mỗi bit nhiều lần cũng là một cách thức để khắc phục lỗi. Giả sử với cách gửi lặp 3 lần, để gửi đi bit 1, phía gửi sẽ gửi đi "111". Nếu nhận được 3 bit không hoàn toàn giống nhau thì chắc chắn có lỗi. Nếu kênh truyền có độ tin cậy rất cao có thể đảm bảo tối đa 1 bit lỗi trong bộ ba mã hóa, thì khi nhận được bộ "001", "010" hay "100" ta có thể xem chúng ứng

với bit 0; trong khi các bộ ba "110", "101" và "011" ứng với bit 1. Phương pháp mã hóa kiểu như trên cho phép phía nhận có thể khôi phục lại thông tin đúng, cho dù trên kênh truyền xảy ra lỗi – đây là khả năng sửa lỗi.

Tuy nhiên hiệu suất của phương pháp gửi lặp rất thấp do làm giảm băng thông đường truyền đi ba lần. Nếu số lần gửi lặp tăng (để có thể phát hiện và sửa được nhiều lỗi hơn) thì hiệu suất cũng giảm theo.

Mã Hamming

Nếu có nhiều bit sửa lỗi được đặt trong thông điệp và bố trí các bit này một cách hợp lý sao cho các vị trí bit lỗi khác nhau tạo ra những kết quả khác nhau thì chúng ta hoàn toàn xác định được các bit bị lỗi nằm ở đâu. Trong một thông điệp 7 bit, có thể có 7 vị trí có lỗi, do đó chỉ cần 3 bit để xác định thông điệp có lỗi hay không và nếu có thì vị trí của lỗi.

Hamming đã nghiên cứu các phương pháp mã hóa trước đó để tìm hiểu một khái niệm tổng quát. Ông bắt đầu với việc tính toán số bit dữ liệu và số bit dư thừa trong một thông điệp. Chẳng hạn với phương pháp bit chẵn lẻ, thông điệp là một ký tự ASCII 7 bit sẽ được mô tả là mã có dạng (8,7) nghĩa là trong 8 bit có 7 bit dữ liệu thực sự. Ví dụ trong phương pháp mã lặp nêu trên thì mã có dạng (3,1).

Hamming cũng chú ý đến vấn đề có nhiều bit bị lỗi, và đưa ra định nghĩa "khoảng cách"– sau này được gọi là khoảng cách Hamming để định nghĩa sự khác biệt giữa hai từ khóa. Mã chẵn lẻ có khoảng cách là 2 vì không thể phát hiện được 2 lỗi. Mã lặp (3,1) có khoảng cách là 3 vì nếu lỗi ở cả 3 bit thì kết quả là một từ mã hợp lệ khác (000 -> 111 hay 111 -> 000). Mã lặp (4,1) (mỗi bit được lặp 4 lần) có khoảng cách là 4, do đó hoàn toàn có thể phát hiện được tình huống có 2 bit trong từ bị lỗi.

Hamming quan tâm đến hai vấn đề: làm thế nào để tăng khoảng cách cho phương pháp mã nhưng giữ được lượng bit dư thừa thấp nhất có thể. Trong những năm 1940, ông đã đưa ra nhiều phương pháp khác nhau cải tiến các hệ mã hiện thời với ý tưởng chính là xen kẽ các bit chẵn lẻ sao cho chúng không chỉ kiểm soát dữ liệu mà còn kiểm soát lẫn nhau.

Ở đây chúng ta sẽ xét mã hóa Hamming (7,4) được đưa ra vào năm 1950. Cứ mỗi cụm 4 bit (D) trong thông điệp sẽ chèn thêm 3 bit kiểm tra chẵn lẻ (P). Phương pháp này sẽ sửa được bất kỳ lỗi 1 bit nào và có thể phát hiện tất cả lỗi 2 bit.

Giá trị mỗi bit kiểm tra phụ thuộc vào giá trị một số bit trong từ mã và chính vị trí của bit kiểm tra sẽ xác định nó phụ thuộc vào những bit nào.

Hình 5.9 minh họa cách xác định vị trí của bit kiểm tra và thuật toán tính những bit kiểm tra này.

Tất cả các bit ở vị trí là lũy thừa của 2 đều là bit kiểm tra (các bit ở vị trí 1,2,4,8..)

Tất cả các bit ở vị trí còn lại là dữ liệu thực sự cần gửi đi (các vị trí 3,5,6,7,9…

Vị trí 1: Bỏ qua 1 bit, kiểm tra 1 bit, bỏ qua 1 bit, kiểm tra 1 bit

Vị trí 2: Kiểm tra 1 bit, bỏ qua 2 bit, kiểm tra 2 bit, bỏ qua 2 bit, kiểm tra 2 bit

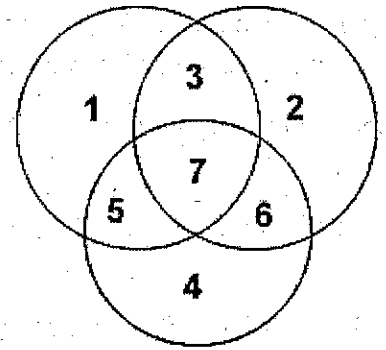
Vị trí 4: Kiểm tra 3 bit, bỏ qua 4 bit, kiểm tra 4 bit, bỏ qua 4 bit, kiểm tra 4 bit

Vị trí 8: Kiểm tra 7 bit, bỏ qua 8 bit, kiểm tra 8 bit, bỏ qua 8 bit, kiểm tra 8 bit

7	6	5	4	3	2	1	
D	**D**	**D**	**P**	**D**	**P**	**P**	Từ mã 7 bit
D	-	D	-	D	-	P	Bit kiểm tra
D	D	-	-	D	P	-	Bit kiểm tra
D	D	D	P	-	-	-	Bit kiểm tra

Hình 5.9 Vị trí các bit kiểm tra và những bit nó kiểm soát

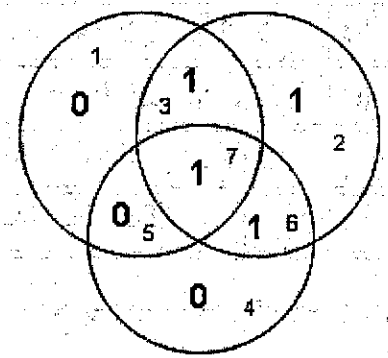
Chúng ta sẽ giải thích tại sao đặt bit kiểm tra vào các vị trí đó? Xét Hình 5.10, chúng ta thấy 3 bit kiểm tra (1,2,4) có liên quan đến các bit (3,5,6,7). Trong sơ đồ ven này, mỗi vòng tròn có một bit kiểm tra và 3 bit dữ liệu và tổng của 4 bit trong mỗi vòng tròn phải bằng 0. Với 4 bit dữ liệu, chúng ta có thể dễ dàng lựa chọn 3 bit kiểm tra thỏa mãn điều kiện này.



Hình 5. 10 Quan hệ giữa bit kiểm tra và các bit dữ liệu

Có thể chứng minh sự thay đổi của 1 trong 7 bit (1-7) sẽ làm thay đổi 3 bit kiểm tra một cách duy nhất. Ví dụ nếu bit 7 thay đổi thì cả 3 bit kiểm tra cũng thay đổi, nhưng nếu bit 6 thay đổi thì chỉ bit 2 và 4 mới thay đổi. Và nếu chính bit kiểm tra thay đổi thì chỉ có duy nhất bit này bị thay đổi, các bit còn lại không bị ảnh hưởng gì. Như vậy có thể dễ dàng xác định trực tiếp vị trí bit bị lỗi bằng cách kiểm tra ba vòng tròn.

Sự ảnh hưởng của lỗi duy nhất đến 3 bit kiểm tra phụ thuộc vào vị trí bit lỗi.



Hình 5.11 Cách xác định lỗi dựa vào sơ đồ ven

Giả sử bên gửi gửi chuỗi (1 1 0 0 1 1 0) và trên đường truyền, bit 5 bị lỗi (0 sang 1). Ở phía bên nhận, lỗi ở vị trí trên (bit 5) có thể được sửa bằng cách kiểm tra xem những bit kiểm tra nào bị tác động bởi bit lỗi.

Hình 5.12 minh họa điều này.

7	6	5	4	3	2	1			
1	1	1	0	1	1	0	Từ mã 7 bit		
1	-	1	-	1	-	0	Tổng chẵn	**Sai !**	1
1	1	-	-	1	1	-	Tổng chẵn	**Đúng !**	0
1	1	1	0	-	-	-	Tổng chẵn	**Sai !**	1

Hình 5.12 Cách xác định vị trí bit bị lỗi

Như vậy bit lỗi là 101 bằng 5. Qua ví dụ này, ta thấy bất kỳ lỗi nào cũng có thể được sửa bằng cách tương tự.

Mã Hamming tạo ra đúng 16 từ mã hợp lệ trong 128 từ có độ dài 7 bit có thể. 16 từ mã hợp lệ này được sắp xếp sao cho khoảng cách tối thiểu giữa 2 từ bất kỳ là 3 (Hình 5.13)

	7 6 5 4 3 2 1		7 6 5 4 3 2 1
0	0 0 0 **0** 0 **0** 0	8	1 0 0 **1** 0 **1** 1
1	0 0 0 **0** 1 **1** 1	9	1 0 0 **1** 1 **0** 0
2	0 0 1 **1** 0 **0** 1	A	1 0 1 **0** 0 **1** 0
3	0 0 1 **1** 1 **1** 0	B	1 0 1 **0** 1 **0** 1
4	0 1 0 **1** 0 **1** 0	C	1 1 0 **0** 0 **0** 1
5	0 1 0 **1** 1 **0** 1	D	1 1 0 **0** 1 **1** 0
6	0 1 1 **0** 0 **1** 1	E	1 1 1 **1** 0 **0** 0
7	0 1 1 **0** 1 **0** 0	F	1 1 1 **1** 1 **1** 1

Hình 5.13 Các từ mã Hamming hợp lệ

5.3. GIAO THỨC ĐA TRUY CẬP VÀ MẠNG CỤC BỘ

Trong phần mở đầu của chương, chúng ta đã thấy rằng có hai kiểu kết nối mạng: kiểu truyền điểm nối điểm (point-to-point) và kiểu truyền

quảng bá (broadcast). Trên đường truyền điểm nối điểm có duy nhất một bên gửi và một bên nhận. Nhiều giao thức tầng liên kết dữ liệu được thiết kế cho đường truyền điểm nối điểm như PPP (Point-to-Point Protocol) và HDLC. Kiểu truyền thứ hai - kiểu quảng bá cho phép có nhiều nút gửi và nút nhận cùng kết nối đến kênh truyền dùng chung duy nhất. Khi bất kỳ nút nào đó truyền đi một frame, kênh truyền sẽ quảng bá frame này và tất cả các nút khác đều nhận được bản sao của frame. Ethernet - công nghệ quảng bá được sử dụng rộng rãi nhất - sẽ được nghiên cứu chi tiết trong phần 5.5. Trong phần này, chúng ta sẽ nghiên cứu một trong những vấn đề quan trọng nhất của tầng liên kết dữ liệu: làm thế nào để điều phối việc truy cập vào kênh truyền chung của nhiều nút - vấn đề **đa truy cập (multiple access problem)**. Kênh truyền quảng bá thường được sử dụng trên mạng cục bộ - là mạng giới hạn trong một khu vực địa lý.

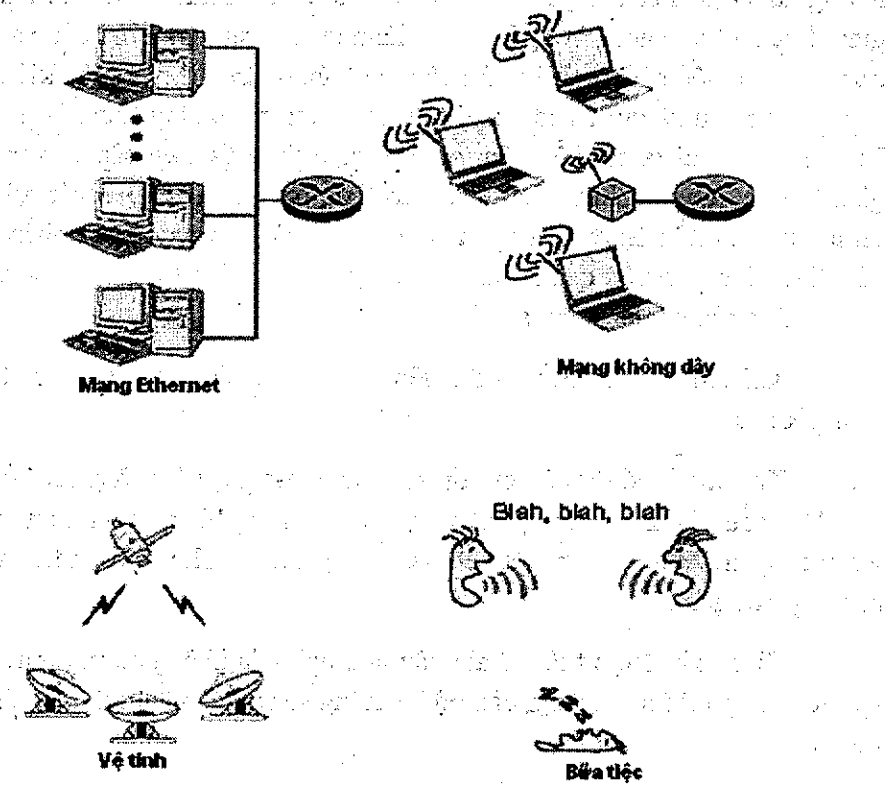
Chúng ta đã quá quen thuộc với công nghệ quảng bá trong hệ thống phát thanh truyền hình. Nhưng hệ thống này chỉ phát quảng bá một chiều (nghĩa là một nút cố định – là ăngten - truyền dữ liệu đến nhiều nút nhận), trong khi các nút trên kênh truyền quảng bá trong mạng máy tính có thể vừa gửi vừa nhận. Có thể xét ví dụ tương tự trong một bữa tiệc, mọi người cùng nhau tụ họp trong một đại sảnh (không khí cung cấp môi trường quảng bá) để nói chuyện với nhau. Ví dụ thứ hai là lớp học - nơi giáo viên và sinh viên cùng nhau chia sẻ môi trường quảng bá duy nhất.Vấn đề chính trong cả hai trường hợp này là việc quyết định ai sẽ là người được nói (nghĩa là được truyền trên kênh truyền). Với con người, chúng ta đã có những quy tắc giao tiếp theo phép lịch sự để chia sẻ kênh truyền chung:

"Mỗi người đều có cơ hội nói"
"Im lặng cho đến khi bạn được quyền nói"
"Không được quyền nói suốt"
"Giơ tay yêu cầu nếu muốn nói"
"Đừng ngắt lời ai đó đang nói "
"Đừng ngủ khi ai đó đang nói"

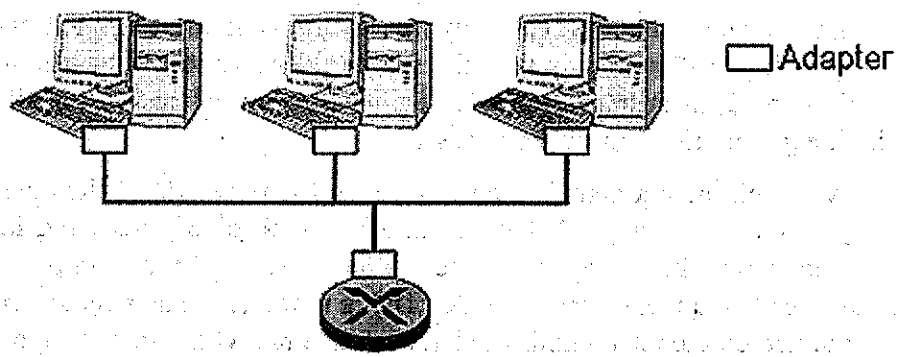
Tương tự như vậy, mạng máy tính cũng có những giao thức - gọi là giao thức đa truy cập **(multiple access protocol)** – cho phép các nút điều chỉnh việc truyền thông của mình trên kênh truyền quảng bá dùng chung. Như minh họa trên Hình 5.14, giao thức đa truy cập rất cần thiết trong nhiều kiểu môi trường mạng: mạng không dây, mạng có dây và mạng vệ tinh.

Hình 5.15 là ví dụ một kênh truyền quảng bá chia sẻ kênh truyền ùng chung. Mặc dù mỗi máy tính truy cập kênh truyền qua adapter, trong hần này chúng ta sẽ xem máy tính như một nút đóng vai trò vừa nhận lẫn ửi. Trên thực tế, hàng trăm hoặc thậm chí hàng nghìn nút có thể trực tiếp uyền thông trên một kênh truyền quảng bá.

Vì tất cả các nút đều có khả năng truyền frame nên rất có thể nhiều út truyền frame tại cùng một thời điểm. Khi đó, tất cả các nút cùng lúc hận được nhiều frame, nghĩa là các frame được truyền sẽ **xung đột collide)** với nhau tại tất cả các nút nhận. Thông thường khi xung đột xảy ra, hông nút nào có thể nhận chính xác bất kỳ frame nào vì tín hiệu trong các ame đan xen vào nhau hoàn toàn. Vì thế tất cả các frame liên quan đến ung đột đều bị mất, và có thể coi kênh truyền dùng chung không được sử ng trong khoảng thời gian xảy ra xung đột. Rõ ràng khi nhiều nút thường uyên muốn truyền frame, xác suất xảy ra xung đột sẽ tăng và phần lớn ăng thông của kênh truyền bị lãng phí.



Hình 5.14 Chia sẻ kênh truyền dùng chung



Hình 5.15 Mạng Ethernet quảng bá

Để bảo đảm hiệu suất của kênh truyền quảng bá đạt giá trị tối đa khi nhiều nút muốn gửi dữ liệu, bằng cách nào đó phải có cơ chế phối hợp giữa những nút có nhu cầu truyền. Cơ chế phối hợp này chính là trách nhiệm của giao thức đa truy cập. Trong ba mươi năm qua, hàng nghìn bài báo và hàng trăm luận án tiến sĩ đã nghiên cứu về giao thức đa truy cập. Nhiều kiểu giao thức khác nhau đã được triển khai trên các công nghệ tầng liên kết dữ liệu. Tuy nhiên người ta có thể phân loại các giao thức đa truy cập vào ba lớp: giao thức **phân chia kênh truyền**, giao thức **truy cập ngẫu nhiên** và giao thức **truy cập lần lượt**. Chúng ta sẽ xem xét các lớp này trong phần dưới đây. Trên kênh truyền quảng bá với tốc độ R b/s, giao thức đa truy cập lý tưởng sẽ có những đặc điểm sau:

Khi chỉ có một nút có dữ liệu gửi đi, nút đó được gửi với thông lượng R bps.

Khi M nút có dữ liệu gửi đi, mỗi nút được gửi với thông lượng R/M bps. Yêu cầu này không có nghĩa rằng mỗi nút trong M nút luôn luôn truyền với tốc độ tức thời R/M mà đây chỉ là tốc độ trung bình xác định trong một khoảng thời gian.

Giao thức được triển khai phân tán, nghĩa là không có một nút đóng vai trò điều phối (nếu không toàn bộ hệ thống sẽ sụp đổ nếu nút điều phối bị hỏng).

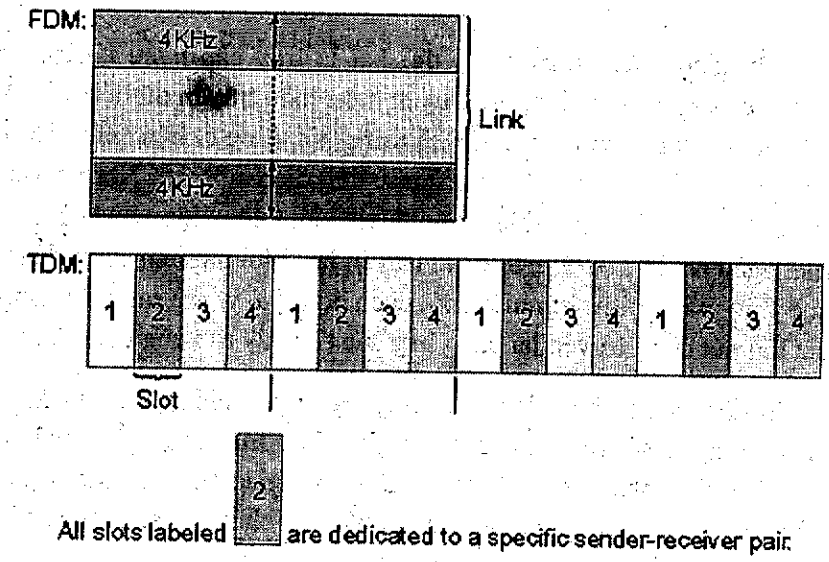
Giao thức phải đơn giản để chi phí cài đặt không cao.

5.3.1 Giao thức phân chia kênh truyền (channel partitioning)

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 kênh truyền cho các nút tham gia truyền thông. Giả sử kênh truyền có N nút và tốc độ truyền của kênh là Rbps. TMD chia thời gian thành các khoảng (time frame) (độc lập với đơn vị dữ liệu frame ở tầng data-link) và sau đó lại chia mỗi khoảng thời gian thành N khe thời gian (time slot). Mỗi khe thời gian được cấp phát cho một nút. Khi có dữ liệu cần gửi, nút truyền các bit dữ liệu của mình trong khe thời gian đã được cấp phát. Thường khoảng thời gian được chọn sao cho một frame dữ liệu có thể truyền trọn vẹn trong một khe thời gian.

Hình 5.16minh hoạ TDM đơn giản cho 4 nút. Trong bữa tiệc điều này tương tự như quy định mỗi người chỉ được nói trong từng khoảng thời gian quy định. Vì thế ai cũng có cơ hội để nói.

Ưu điểm chính của TDM là loại trừ xung đột và đảm bảo công bằng: mỗi nút có được tốc độ truyền riêng R/N bps trong mỗi khoảng thời gian. Tuy nhiên, xuất hiện hai nhược điểm: tốc độ truyền trung bình của mỗi nút bị giới hạn R/N bps và nút chỉ được truyền trong khoảng thời gian của mình ngay cả khi nó là nút duy nhất có nhu cầu gửi. (Trong ví dụ bữa tiệc của chúng ta mọi người đều muốn nghe một vị khách nào đó nói thì vị khách đó chỉ được nói trong khoảng thời gian được cấp phát của mình). Rõ ràng, TDM không phải là một giao thức đa truy cập tốt cho bữa tiệc kiểu này.



Hình 5.16 Ví dụ TDM cho 4 nút

Nếu TDM chia kênh truyền theo thời gian thì FDM chia kênh truyền R bps ra các tần số khác nhau (mỗi tần số có băng thông R/N) và mỗi nút được cấp phát một dải tần số. Vì thế, FDM tạo ra N kênh truyền nhỏ R/Nbps từ kênh truyền lớn R bps. Ưu nhược điểm của FDM giống TDM, tức là cũng loại bỏ được xung đột và phân chia công bằng dải tần giữa N nút. Tuy nhiên, nhược điểm là tốc độ gửi của nút bị giới hạn ngay cả khi chỉ có duy nhất một nút có nhu cầu gửi dữ liệu.

Giao thức phân chia kênh truyền thứ ba là **chia mã (CDMA – Code Division Multiple Access)**. Nếu TDM và FDM cấp phát khoảng thời gian và tần số cho các nút thì CDMA cấp phát cho mỗi nút một mã khác nhau. Sau đó 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). CDMA đã được sử dụng trong hệ thống quốc phòng nhờ đặc tính chống nhiễu và bây giờ bắt đầu được áp dụng phổ biến cho mục đích dân sự, đặc biệt trong đa truy cập kênh truyền không dây.

5.3.2 Giao thức truy cập ngẫu nhiên (random access)

Kiểu giao thức đa truy cập thứ hai là truy cập ngẫu nhiên. Trong giao thức truy cập ngẫu nhiên, nút truyền luôn luôn truyền dữ liệu với tốc độ cao nhất của kênh truyền R bps. Khi có xung đột, các nút liên quan đến xung đột sẽ truyền lại frame cho đến khi frame đến đích an toàn. Nhưng khi biết có xung đột, nút không cần thiết truyền lại frame ngay lập tức, mà đợi một thời gian ngẫu nhiên nào đó trước khi truyền lại. Mỗi nút liên quan đến xung đột chọn thời gian đợi ngẫu nhiên một cách độc lập, vì thế sau mỗi xung đột xác suất hai nút cùng truyền lại cùng một lúc (lại xảy ra xung đột) sẽ giảm.

Có đến hàng trăm giao thức truy cập ngẫu nhiên, tuy nhiên trong phần này chúng ta sẽ trình bầy một vài giao thức truy cập ngẫu nhiên được sử dụng phổ biến nhất - giao thức ALOHA và giao thức đa truy cập cảm nhận sóng mang (CSMA). Sau đó, trong phần 5.5 chúng ta sẽ nghiên cứu chi tiết về Ethernet - công nghệ sử dụng CSMA cực kỳ phổ biến.

Slotted ALOHA

Chúng ta bắt đầu nghiên cứu một trong số những giao thức đa truy cập đơn giản nhất: slotted ALOHA. Chúng ta giả định như sau:

- Tất cả frame có chính xác L bit.

- Thời gian được chia thành các khoảng L/R s (phải là khoảng thời gian đủ để truyền một frame).

- Nút bắt đầu truyền frame tại đầu mỗi khoảng thời gian.

- Tất cả các nút được đồng bộ hoá sao cho mỗi nút đều xác định được khi nào là đầu của khoảng thời gian.

- Nếu có nhiều frame xung đột trong khoảng thời gian nào đó thì tất cả các nút đều phát hiện sự kiện xung đột ngay trong khoảng thời gian đó.

Gọi p là xác suất ($0 \le p \le 1$). Hoạt động của slotted ALOHA trong mỗi nút như sau:

- Khi có frame mới cần gửi, nút sẽ đợi đến thời điểm đầu của khoảng thời gian kế tiếp và gửi toàn bộ frame trong khoảng thời gian đó.

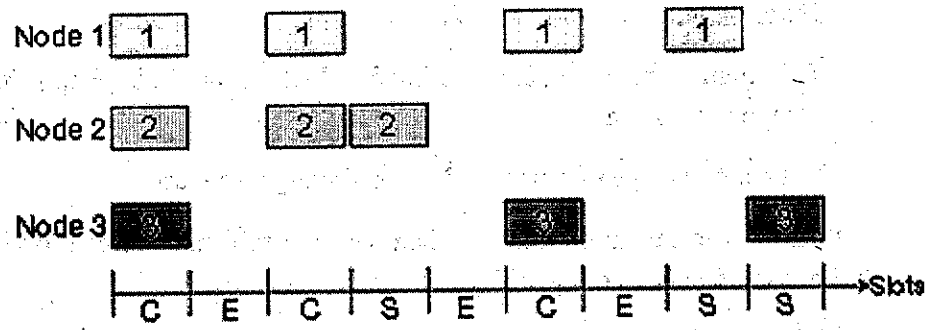
- Nếu không xảy ra xung đột, nút truyền thành công frame và vì vậy không cần thiết phải truyền lại (nút có thể chuẩn bị frame mới để truyền, nếu có).

- Nếu có xung đột, nút phát hiện xung đột ngay trong khoảng thời gian và sẽ truyền lại frame trong khoảng thời gian tiếp theo với xác suất p cho đến khi frame được truyền thành công.

Truyền lại với xác suất p giống như việc tung đồng xung: biến cố mặt ngửa ứng với việc truyền lại xảy ra với xác suất p. Biến cố mặt sấp ứng với việc "bỏ qua khoảng thời gian này và tung lại đồng xu trong khoảng thời gian kế tiếp" xảy ra với xác suất (1- p). Mỗi nút liên quan đến xung đột tung đồng xu độc lập với nhau.

Slotted ALOHA có nhiều ưu điểm. Không giống phân chia kênh truyền, nút tích cực duy nhất (nghĩa là nút có nhu cầu gửi dữ liệu) liên tục gửi frame ở tốc độ cao nhất của kênh truyền. Slotted ALOHA là một thuật

toán phân tán vì mỗi nút khi phát hiện ra xung đột sẽ quyết định khi nào truyền lại một cách độc lập. (Tuy nhiên slotted ALOHA đòi hỏi phải có cơ chế đồng bộ trên tất cả các nút).



Hình 5.17 Ví dụ giao thức Slotted ALOHA

Slotted ALOHA hoạt động tốt khi chỉ có một nút ở trạng thái tích cực, nhưng hiệu suất của nó bằng bao nhiêu khi có nhiều nút tích cực? Có hai yếu tố phải tính đến ở đây. Thứ nhất như trong Hình 5.17 khi có nhiều nút ở trạng thái tích cực sẽ xuất hiện nhiều khoảng thời gian xung đột và do đó kênh truyền bị lãng phí. Thứ hai sẽ có một số khoảng thời gian "rỗng" vì trong khoảng thời gian này tất cả các nút tích cực đều dừng lại đợi (kết quả của chính sách truyền theo xác suất). Chỉ trong những những khoảng thời gian "không bị lãng phí" sẽ có duy nhất một nút truyền thành công. Khoảng thời gian này gọi là khoảng thời gian thành công. Hiệu suất của giao thức được định nghĩa là tỷ lệ các khoảng thời gian truyền thành công trong trường hợp có nhiều nút tích cực, mỗi nút cần gửi đi nhiều frame. Rõ ràng rằng nếu không có cơ chế điều khiển truy cập và nút truyền lại ngay sau mỗi lần xung đột, hiệu suất sẽ bằng 0. Slotted ALOHA có hiệu suất lớn hơn 0 nhưng bằng bao nhiêu?

Bây giờ ta sẽ xác định hiệu suất tối đa của slotted ALOHA. Để đơn giản chúng ta thay đổi giao thức một chút và giả thiết rằng mỗi nút truyền frame trong mỗi khoảng thời gian với xác suất p (tức là mỗi nút luôn có một frame để gửi đi và frame này được gửi đi với xác suất p cho dù đây là frame mới hay frame phải gửi lại). Đầu tiên giả sử có N nút. Xác suất thành công của một khoảng thời gian nào đó là xác suất chỉ có một nút duy nhất truyền

và N-1 nút còn lại không truyền. Xác suất một nút nào đó truyền là p; xác suất mà các nút còn lại không truyền là $(1- p)^{N-1}$. Do vậy xác suất để một nút nào đó truyền trong khi các nút khác không truyền là $p(1-p)^{N-1}$. Vì có N nút, nên xác suất để có khoảng thời gian thành công bằng $Np(1-p)^{N-1}$.

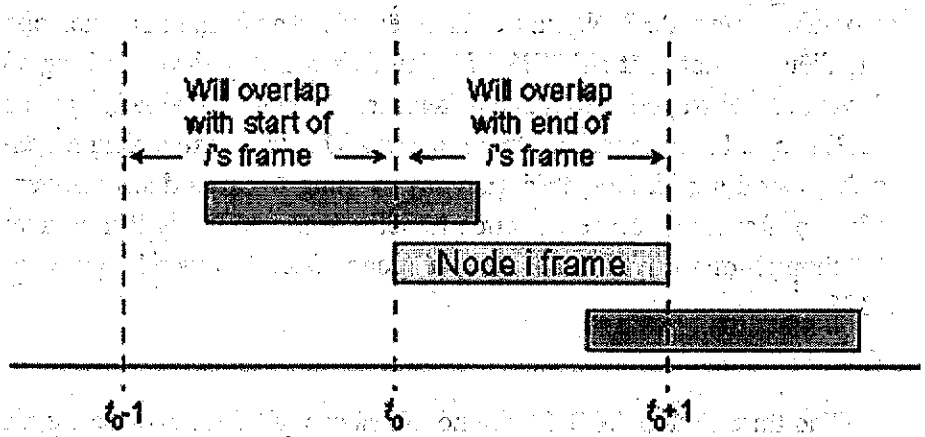
Do đó khi có N nút tích cực, hiệu suất của slotted ALOHA là $Np(1-p)^{N-1}$. Để đạt được hiệu suất lớn nhất, chúng ta phải xác định p* sao cho biểu thức này đạt giá trị lớn nhất. Và để đạt được hiệu suất lớn nhất khi có nhiều nút tích cực, chúng ta phải tính giới hạn của $Np^*(1-p^*)^{N-1}$ khi N tiến tới vô cùng. Áp dụng các công cụ toán học, chúng ta sẽ xác định được hiệu suất lớn nhất của giao thức là 1/e = 0.37. Nghĩa là, khi nhiều nút cùng ở trạng thái tích cực thì trong điều kiện tốt nhất chỉ 37% thời gian đường truyền được sử dụng có ích. Vì vậy, tốc độ truyền hiệu quả của kênh truyền không phải là R bps mà chỉ là 0,37R bps. Phân tích tương tự chỉ ra rằng 37% thời gian đường truyền không được sử dụng và 26% thời gian xảy ra xung đột trên đường truyền. Như vậy một frame nào đó có thể được truyền với tốc độ tối đa R nhưng về tổng thể thông lượng truyền thành công của toàn bộ kênh truyền không vượt qua 0.37R.

ALOHA

Giao thức slotted ALOHA đòi hỏi tất cả các nút đồng bộ việc truyền tại đầu mỗi khoảng thời gian. Giao thức ALOHA đầu tiên thực sự là giao thức không chia khoảng thời gian, hoàn toàn phân tán. Trong giao thức này, khi có dữ liệu cần gửi đi, ngay lập tức nút truyền toàn bộ frame vào kênh truyền dùng chung. Nếu frame được truyền xung đột với frame từ các nút khác, thì ngay sau khi truyền cho xong frame, nút sẽ ngay lập tức truyền lại frame với xác suất p. Ngược lại nút đợi trong một khoảng thời gian truyền frame. Sau quá trình chờ đợi, nút truyền frame với xác suất là p, hoặc đợi (không làm gì cả) trong khoảng thời gian truyền frame với xác suất (1-p).

Để xác định được hiệu suất cực đại của ALOHA xét trên một nút duy nhất, chúng ta cũng giả định như trong trường hợp slotted ALOHA thời gian truyền frame là một đơn vị thời gian. Tại bất kì thời gian nào, xác suất để nút truyền frame là p. Giả sử frame này bắt đầu truyền tại thời điểm t_0. Như minh họa trong Hình 5.18, để frame này được truyền thành công thì không nút nào được bắt đầu truyền trong khoảng thời gian $[t_0 - 1, t_0]$. Nếu

không tín hiệu của những frame này sẽ xung đột với các tín hiệu đầu tiên của frame đang xét. Xác suất để tất cả các nút khác không được bắt đầu truyền trong khoảng thời gian này là $(1-p)^{N-1}$. Tương tự không nút nào được bắt đầu truyền trong khi nút đang xét đang truyền. Xác suất của điều này cũng là $(1-p)^{N-1}$. Vì vậy xác suất nút nào đó truyền thành công là $p(1-p)^{2(N-1)}$. Bằng cách lấy giới hạn như trong trường hợp slotted ALOHA, chúng ta thấy rằng hiệu suất lớn nhất của giao thức ALOHA là 1/2e - bằng một nửa của slotted ALOHA. Đây là cái giá phải trả cho giao thức ALOHA hoàn toàn phân tán.



Hình 5.18 Các frame đan xen vào nhau trong ALOHA

CSMA - Đa truy cập cảm nhận sóng mang

Trong cả hai giao thức ALOHA và slotted ALOHA, quyết định truyền của nút được đưa ra độc lập với các nút khác. Cụ thể hơn, một nút không để ý tới việc liệu có nút khác đang truyền khi nó bắt đầu truyền hay không và nút cứ truyền kể cả khi có nút khác truyền (gây xung đột). Tương tự trong ví dụ buổi tiệc, giao thức ALOHA giống như hành vi của vị khách bất lịch sự cứ liên tục nói bất chấp việc có người đang nói hay không. Xã hội loài người có những quy tắc ứng xử cho phép xử sự một cách lịch sự và làm giảm "xung đột" giữa nhiều người nói chuyện. Đặc biệt, có hai quy tắc quan trọng cho một cuộc đối thoại của người lịch sự:

Nghe trước khi nói: nếu có ai đang nói, hãy đợi đến khi họ nói xong. Trong mạng máy tính, điều này được gọi là **cảm nhận sóng mang (carrier sense)** - nút lắng nghe kênh truyền trước khi truyền. Nếu có frame đang được truyền trên kênh truyền thì nút sẽ chờ (backs off) một khoảng thời gian ngẫu nhiên và lại tiếp tục lắng nghe kênh truyền. Lúc này nếu kênh truyền được cảm nhận là rỗi thì nút bắt đầu việc truyền frame. Trong trường hợp ngược lại, nút lại đợi một khoảng thời gian ngẫu nhiên khác và lặp lại quá trình này.

Nếu có ai đó bắt đầu nói cùng lúc thì hãy tạm ngừng (Nghe trong khi nói): Trong mạng máy tính điều này được gọi là **phát hiện xung đột (collision detection)** - nút đang truyền vẫn phải tiếp tục lắng nghe kênh truyền trong khi đang truyền. Nếu phát hiện có nút khác truyền xen vào, nút sẽ dừng truyền và sử dụng giao thức nào đó để quyết định khi nào nên thử truyền tiếp.

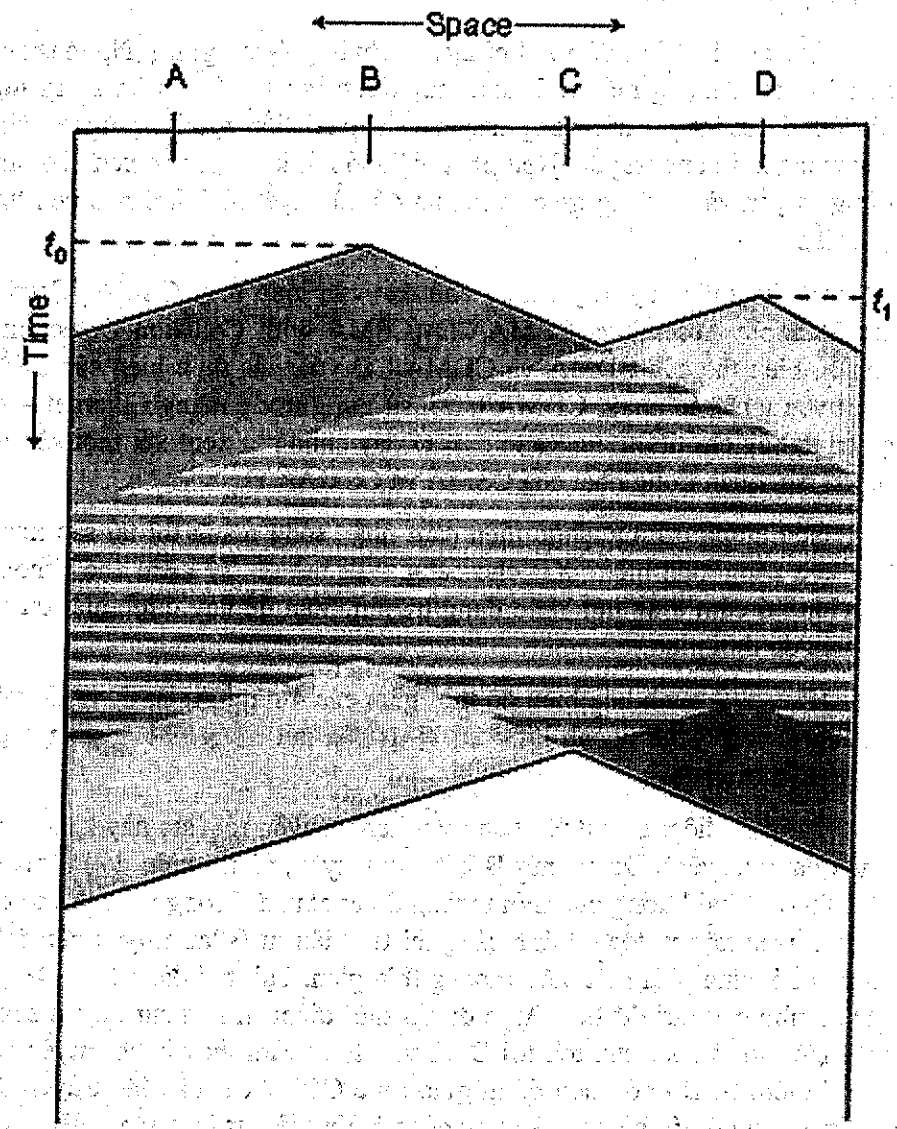
Hai quy tắc này là ý tưởng chủ đạo của giao thức CSMA (Carrier Sense Multiple Access) và CSMA/CD (CSMA with Collision Detection). Có nhiều biến thể của CDMA và CDMA/CD với việc thực hiện các chiến lược truyền lại khác nhau. CDMA/CD - sử dụng trong mạng Ethernet – sẽ được trình bày chi tiết trong phần 5.5. Ở đây chúng ta xem xét một số đặc trưng cơ bản quan trọng nhất của CSMA và CSMA/CD.

Nếu tất cả các nút thực hiện cảm nhận sóng mang thì tại sao xung đột có khả năng xuất hiện? Xét cho cùng, một nút sẽ "tự kiềm chế" không truyền khi nó cảm thấy nút khác đang truyền. Vấn đề này được giải quyết bằng biểu đồ thời gian.

Hình 519 minh họa biểu đồ thời gian của 4 nút (A, B, C, và D) với bus dùng chung. Trục hoành biểu thị vị trí của nút trong không gian, trục tung mô tả thời gian.

Tại thời điểm t_0, nút B nhận thấy kênh truyền rỗi (lúc này không có nút nào đang truyền). Do đó nút B bắt đầu truyền, và tín hiệu do B truyền lan tỏa theo cả hai hướng của môi trường dùng chung. Trong thực tế cho dù vận tốc truyền xấp xỉ tốc độ ánh sáng thì tín hiệu từ B lan truyền đến một điểm nào đó cũng phải mất một khoảng thời gian. Tại thời điểm t_1 ($t_1 > t_0$) nút D có nhu cầu gửi dữ liệu. Mặc dù tại thời điểm t_1, B đang truyền song các tín hiệu từ B chưa lan tỏa tới D và vì vậy D cảm thấy kênh truyền rỗi vào thời điểm t_1. Do đó, theo đúng giao thức CDMA, D bắt đầu truyền dữ liệu. Ngay sau đó, tín hiệu từ B xung đột với tín hiệu từ D. Hiển nhiên rằng độ trễ lan tỏa (propagation delay) giữa hai đầu mút của kênh truyền dùng

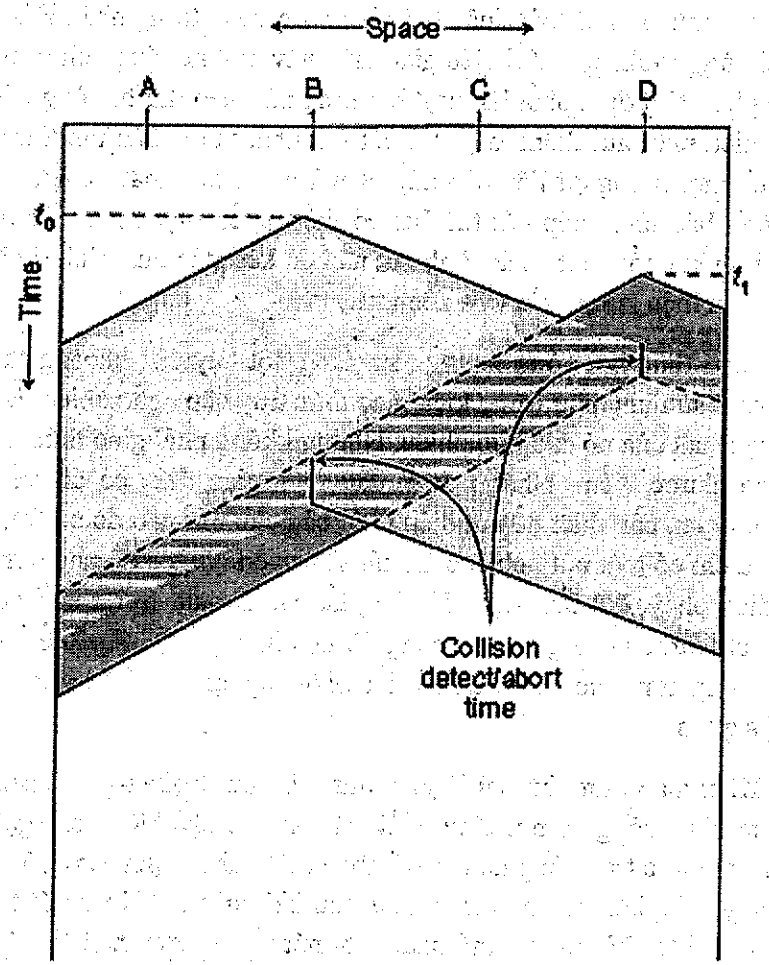
chung - thời gian để tín hiệu lan truyền từ đầu này đến đầu kia kênh truyền đóng vai trò quyết định trong hiệu suất hoạt động của kênh truyền. Thời gian trễ này càng lớn, xác suất một nút không phát hiện được có nút khác đang truyền cũng càng lớn.



Hình 5.19 Biểu đồ thời gian của 4 nút sử dụng CSMA

Trong Hình 5.19 các nút không thực hiện phát hiện xung đột, cả nút B và D tiếp tục truyền toàn bộ frame ngay cả khi có xung đột. Nếu thực hiện công việc phát hiện xung đột, nút sẽ ngừng truyền ngay khi phát hiện xung đột.

Hình 5.20 tương tự như Hình 5.19 chỉ khác là cả hai nút ngừng truyền ngay sau khi phát hiện có xung đột. Rõ ràng đưa khả năng phát hiện xung đột vào giao thức đa truy cập sẽ làm tăng hiệu suất của giao thức do các nút không cố gắng tiếp tục gửi frame đã bị lỗi. Giao thức Ethernet chúng ta sẽ nghiên cứu trong phần 5.5 là giao thức CSMA có phát hiện xung đột.



Hình 5.20 CSMA có phát hiện xung đột

5.3.3 Giao thức truy cập lần lượt (Taking – turns)

Hai tính chất mà tất cả các giao thức đa truy cập muốn có là (1) khi chỉ có một nút tích cực, nút này có thể chiếm toàn bộ đường truyền nghĩa là truyền với băng thông tối đa R bps và (2) khi M nút tích cực, mỗi nút có băng thông trung bình R/M bps.

Giao thức ALOHA và CSMA có tính chất đầu tiên nhưng không có tính chất thứ hai. Điều này đã thúc đẩy các nhà nghiên cứu xây dựng một lớp các giao thức khác – kiểu lần lượt. Giống như kiểu truy cập ngẫu nhiên, có rất nhiều giao thức kiểu lần lượt, và mỗi giao thức này lại có nhiều biến thể. Ở đây, chúng ta sẽ thảo luận hai giao thức quan trọng nhất. Đầu tiên là **kiểu hỏi vòng (polling)**. Với kiểu giao thức này một nút được chọn đóng vai trò điều phối. Nút điều phối lần lượt hỏi từng nút theo thứ tự vòng tròn. Đầu tiên nút điều phối gửi thông điệp tới nút thứ nhất thông báo nút thứ nhất có thể truyền một lượng dữ liệu nào đấy. Sau khi nút thứ nhất truyền, nút điều phối thông báo cho phép nút thứ hai có thể truyền một lượng dữ liệu nào đó... Nút điều phối có thể xác định nút nào đó kết thúc quá trình dữ liệu khi không có tín hiệu lan truyền trên kênh truyền.

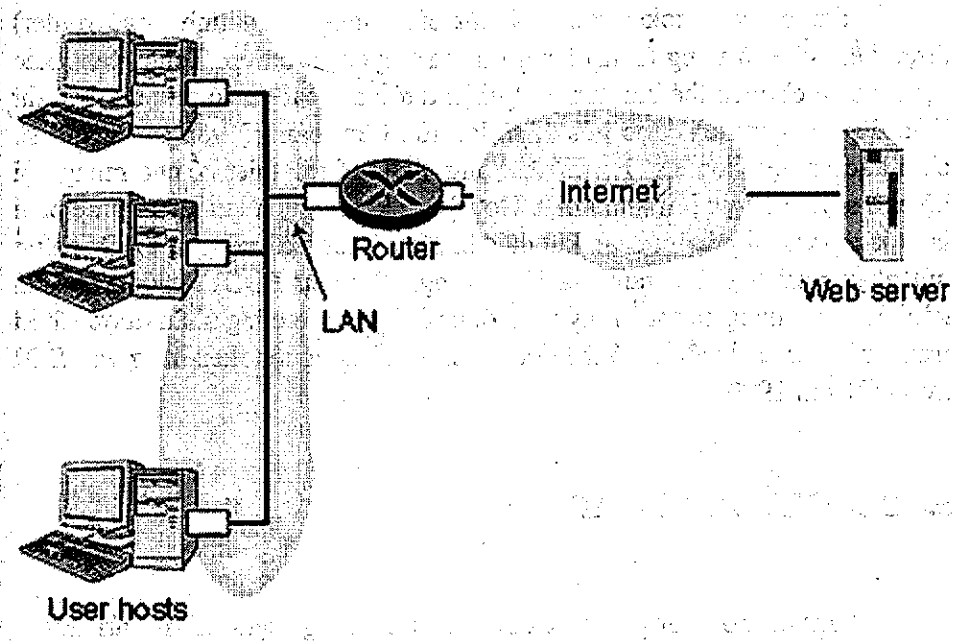
Giao thức hỏi vòng loại trừ sự xung đột hay các khoảng thời gian không được sử dụng như trong kiểu giao thức truy cập ngẫu nhiên. Điều này khiến hiệu suất của nó cao hơn nhiều. Nhưng không phải giao thức hỏi vòng không có nhược điểm. Nhược điểm đầu tiên, giao thức có độ trễ vòng - lượng thời gian cần thiết để nút điều phối báo cho nút nào đó có thể truyền. Ví dụ nếu chỉ có một nút tích cực thì nút sẽ truyền với tốc độ nhỏ hơn R bps, vì nút điều phối phải lần lượt hỏi vòng tất cả các nút, trong mỗi vòng nút tích cực chỉ được phép gửi một lượng dữ liệu hạn chế. Nhược điểm thứ hai, nghiêm trọng hơn nhiều, là nếu nút điều phối gặp sự cố thì toàn bộ hệ thống cũng bị sụp đổ.

Kiểu giao thức thứ hai là giao thức **thẻ bài (token–passing)**. Trong giao thức này không có nút điều phối. Một frame đặc biệt được gọi là **thẻ bài (token)** được trao đổi giữa các nút theo một thứ tự định trước. Ví dụ, nút thứ nhất gửi thẻ bài tới nút thứ hai, nút thứ hai gửi thẻ bài tới nút thứ ba... nút thứ N gửi thẻ bài tới nút thứ nhất. Khi nút nhận được thẻ bài, nó chỉ giữ thẻ bài khi có dữ liệu cần truyền, nếu không nó sẽ ngay lập tức chuyển thẻ

bài tới nút kế tiếp. Nếu nút có frame để truyền, khi nhận được thẻ bài, nó gửi đi lượng dữ liệu được phép và sau đó chuyển thẻ bài tới nút kế tiếp. Giao thức thẻ bài được triển khai phân tán và có hiệu suất cao. Nhưng nó cũng có nhiều vấn đề cần giải quyết. Ví dụ, một nút gặp sự cố có thể làm toàn bộ hệ thống sụp đổ. Hoặc nếu một nút tình cờ không chuyển tiếp hay làm mất thẻ bài thì cần có cơ chế đưa thẻ bài mới vào lưu thông.

5.3.4 Mạng cục bộ LAN (Local Area Network)

Những giao thức đa truy cập được sử dụng trên nhiều loại kênh truyền quảng bá khác nhau. Chúng được sử dụng cho kênh truyền vệ tinh hay môi trường không dây (các nút truyền trên cùng một dải tần số).



Hình 5.21 Máy tính người dùng truy cập máy dịch vụ Web Internet thông qua LAN. Kênh truyền quảng bá giữa máy tính người dùng và router gồm một "đường truyền"

Mạng cục bộ LAN là mạng máy tính giới hạn trong một khu vực địa lý, ví dụ trong một toà nhà hoặc trong khuôn viên trường đại học. Thông

thường khi truy cập Internet từ trường đại học hay cơ quan, hầu hết mọi người truy cập thông qua mạng LAN. Khi đó máy tính của người dùng là một nút trong mạng LAN và mạng LAN cung cấp khả năng truy cập tới Internet thông qua router, như minh hoạ trong Hình 5.21. Mạng LAN là kênh truyền duy nhất giữa tất cả các máy tính và router; do đó nó cần tới giao thức tầng liên kết dữ liệu và giao thức đa truy cập. Tốc độ truyền R của hầu hết các mạng LAN rất cao. Tốc độ mạng LAN trước năm 1980 là 10Mbps, ngày nay là 100Mbps và trong tương lai sẽ là 1Gbps.

Vào những năm 80 và đầu những năm 90, có hai kiểu công nghệ mạng LAN phổ biến trên thị trường. Công nghệ thứ nhất là mạng cục bộ Ethernet (được biết đến là 802.3 LAN [IEEE 802.3) sử dụng giao thức truy cập ngẫu nhiên. Công nghệ thứ hai dựa trên công nghệ thẻ bài gồm công nghệ token–ring và FDDI. Công nghệ Ethernet được trình bày chi tiết trong phần 5.4 vì đây là công nghệ chủ đạo ngày nay.

Trong mạng token–ring, N nút của mạng (máy tính hoặc router) được kết nối vào vòng (ring) bằng đường truyền trực tiếp. Topo mạng xác định thứ tự chuyển thẻ bài. Khi nút nhận được thẻ bài và có nhu cầu gửi dữ liệu, dữ liệu (frame) được gửi đi sẽ lan tỏa trên toàn bộ vòng. Nút gửi sẽ chịu trách nhiệm loại bỏ frame trên vòng. FDDI được thiết kế cho mạng nội bộ cho một khu vực lớn (vài km^2). Do vậy sẽ không hiệu quả nếu frame phải lan tỏa ngược lại phía gửi sau khi đã đến đích. Trong công nghệ FDDI chính nút nhận phải loại bỏ frame ra khỏi vòng. (thực sự FDDI không phải là kênh truyền quảng bá thuần tuý vì không phải nút nào cũng nhận được bất kì frame nào được truyền). Có thể đọc thêm về công nghệ token–ring và FDDI trong [3Com 1999].

5.4 ĐỊA CHỈ LAN VÀ ARP

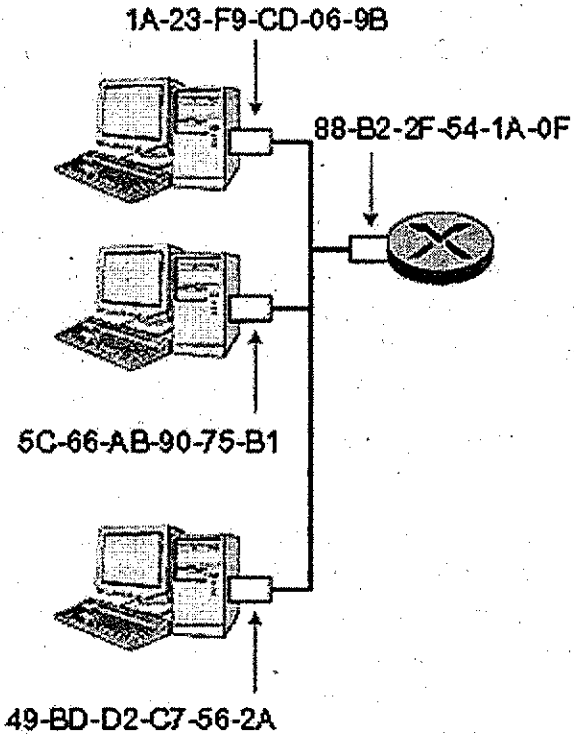
Như đã nói trong phần trước, các nút trong mạng LAN gửi frame cho nhau trên kênh truyền quảng bá dùng chung. Điều này nghĩa là khi một nút trong mạng LAN truyền frame, mọi nút khác kết nối tới mạng LAN đều có khả năng nhận được frame. Tuy nhiên mỗi nút trong mạng LAN không muốn gửi frame tới tất cả các nút khác mà chỉ muốn gửi tới một nút cụ thể nào đó trong mạng LAN. Để thực hiện điều này, các nút trong mạng LAN

phải có khả năng xác định địa chỉ của nhau khi gửi frame, nghĩa là mỗi nút cần có một địa chỉ trong mạng LAN và trong gói tin tầng liên kết dữ liệu (frame) cần có trường chứa địa chỉ nút đích. Như vậy khi nhận được frame, nút có thể xác định liệu frame đó có phải gửi cho mình không.

Nếu địa chỉ đích của frame trùng với địa chỉ LAN của mình, nút nhận sẽ lấy gói dữ liệu tầng mạng từ frame tầng liên kết dữ liệu và chuyển gói dữ liệu này lên tầng mạng phía trên.

Nếu địa chỉ đích không trùng với địa chỉ nút nhận, đơn giản nút sẽ loại bỏ frame nhận được.

5.4.1 Địa chỉ LAN



Hình 5.22 Mỗi adapter có một địa chỉ LAN duy nhất

Thực sự, không phải nút (máy tính) có địa chỉ LAN mà chính adapter mới có địa chỉ LAN. Điều này được minh hoạ trong Hình 5.22. Địa chỉ LAN có nhiều tên gọi khác nhau: địa chỉ vật lý (physical address), địa chỉ Ethernet, địa chỉ MAC (media access control). Với hầu hết các mạng LAN (kể cả mạng Ethernet và thẻ bài), địa chỉ LAN có độ dài 6 byte (có thể có 2^{48} địa chỉ). 6 byte địa chỉ này được biểu diễn dưới dạng thập lục phân, mỗi byte ứng với một cặp số thập lục phân. Địa chỉ mạng LAN của adapter mang giá trị cố định, được ghi cứng vào ROM của adapter trong quá trình sản xuất.

Một điểm thú vị là hai adapter bất kỳ có địa chỉ LAN khác nhau. Thoạt tiên điều này dường như khó có thể thực hiện được vì nhiều công ty khác nhau có thể sản xuất adapter. Làm thế nào để địa chỉ adapter sản xuất ở Đài Loan khác với địa chỉ adapter sản xuất tại Bỉ? Điều này có được là do IEEE quản lý không gian địa chỉ vật lý. Khi muốn sản xuất adapter, công ty phải mua một phần không gian địa chỉ gồm 2^{24} địa chỉ với một mức phí nào đó. IEEE cấp từng khối 2^{24} địa chỉ bằng cách cố định 24 bit đầu của địa chỉ vật lí và công ty có thể tùy ý gán 24 bit sau cho bất kỳ sản phẩm nào của mình.

Địa chỉ vật lý của adapter có cấu trúc phẳng (đối lập với cấu trúc phân cấp) và không thay đổi cho dù có mang adapter đi đâu chăng nữa. Mỗi máy tính xách tay với một card Ethernet luôn có cùng một địa chỉ vật lý cho dù ở bất kỳ ở đâu. Điều này ngược với địa chỉ IP có cấu trúc phân cấp (gồm địa chỉ mạng và địa chỉ máy tính). Địa chỉ IP của nút sẽ thay đổi khi chuyển sang mạng khác. Địa chỉ vật lý của adapter giống như số chứng minh thư nhân dân của một người – không phân cấp và luôn luôn không thay đổi bất kể người đó ở đâu.

Như đã nói, khi adapter muốn gửi frame đến adapter nào đó nằm trên cùng mạng LAN, adapter gửi sẽ đặt địa chỉ nhận vào trong frame. Khi nhận được frame, adapter đích loại bỏ các tiêu đề của frame và gửi dữ liệu lên tầng phía trên. Tất cả các adapter khác trong LAN cũng đều nhận được frame. Tuy nhiên các adapter này sẽ loại bỏ frame (không gửi gói dữ liệu lên trên). Như vậy, các adapter sẽ không làm gián đoạn CPU trung tâm khi chúng nhận được dữ liệu dành cho nút khác. Tuy nhiên, đôi khi adapter gửi

muốn gửi tới tất cả các adapter khác trên mạng LAN. Khi đó adapter sử dụng một địa chỉ đặc biệt: địa chỉ quảng bá trong frame gửi đi. Đối với mạng cục bộ sử dụng địa chỉ 6 byte (như là LAN Ethernet và token-passing) thì địa chỉ quảng bá là chuỗi 48 bit 1 (FF-FF-FF-FF-FF-FF trong hệ 16).

Nguyên lý: Giữ các tầng độc lập với nhau

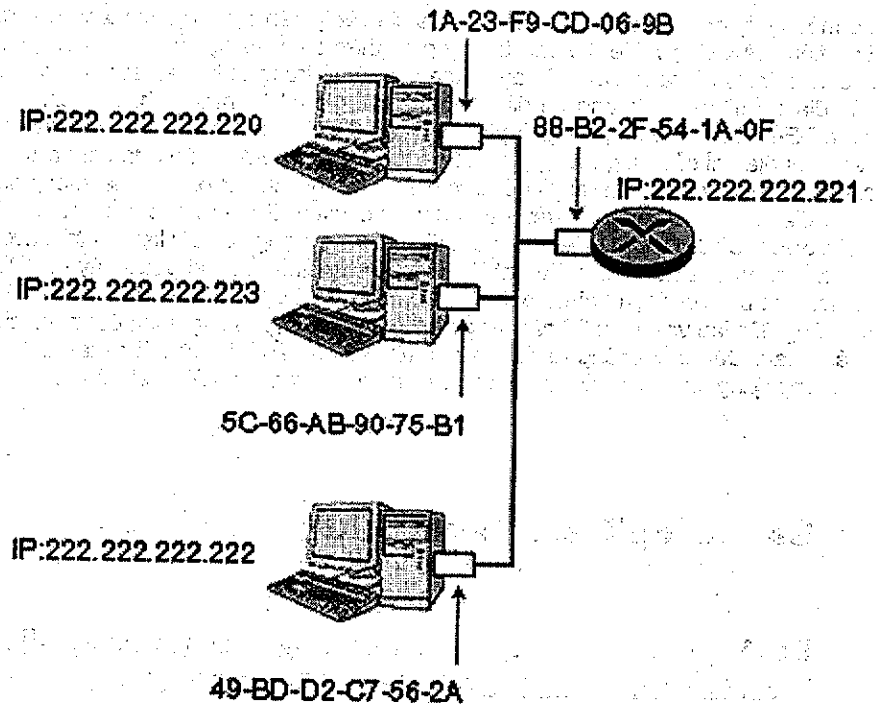
Có một vài lý do vì sao mỗi nút đều có địa chỉ vật lý bên cạnh địa chỉ tầng mạng. Đầu tiên, LAN được thiết kế cho bất kỳ giao thức tầng mạng nào chứ không chỉ cho giao thức IP và Internet. Thay vì địa chỉ vật lý trung tính, nếu adapter được gán địa chỉ IP thì nó không dễ dàng hỗ trợ các giao thức mạng khác (ví dụ IXP hoặc DECNET). Thứ hai, nếu adapter sử dụng địa chỉ tầng mạng thay cho địa chỉ vật lý thì địa chỉ của tầng mạng phải được lưu trữ trong RAM của adapter và phải cấu hình lại khi di chuyển (hay khởi động) adapter. Một lựa chọn khác là không sử dụng địa chỉ của adapter và yêu cầu adapter chuyển dữ liệu trong mỗi frame nó nhận được tới CPU. CPU kiểm tra xem địa chỉ tầng mạng của dữ liệu có trùng với địa chỉ mạng của nó không. Tuy nhiên vấn đề nảy sinh với giải pháp này là CPU sẽ liên tục bị gián đoạn khi có bất kỳ frame nào lan truyền trên LAN. Tóm lại, để các tầng độc lập với nhau thì các tầng có thể có những phương pháp đánh địa chỉ khác nhau. Đến đây chúng ta đã thấy có tới 3 kiểu đánh địa chỉ: tên máy tính ở tầng ứng dụng, địa chỉ IP ở tầng mạng và địa chỉ vật lý ở tầng liên kết dữ liệu

5.4.2. Giao thức giải mã địa chỉ (ARP)

Do tồn tại cả hai kiểu địa chỉ: địa chỉ tầng mạng (chẳng hạn địa chỉ IP) và địa chỉ tầng liên kết dữ liệu (địa chỉ vật lý) nên chắc chắn cần phải có một phương thức biến đổi giữa chúng. Đối với Internet, đây là công việc của giao thức giải mã địa chỉ ARP [RFC 826]. Tất cả máy tính và router trên LAN đều có **module ARP**.

Để hiểu rõ hơn về ARP, xét mạng minh họa trên Hình 5.23. Mỗi nút trong mạng có địa chỉ IP duy nhất và adapter của nút có một địa chỉ vật lý. Địa chỉ IP được viết dưới dạng ký pháp dấu chấm thập phân và địa chỉ LAN được viết dưới dạng ký pháp thập lục phân. Bây giờ, giả sử rằng nút có địa chỉ IP 222.222.222.220 muốn gửi gói dữ liệu IP đến nút có địa chỉ IP 222.222.222.222. Để thực hiện công việc này, nút gửi phải chuyển cho adapter của nó không chỉ gói dữ liệu IP mà cả địa chỉ vật lý của nút nhận

(222.222.222.222). Khi nhận được gói dữ liệu IP và địa chỉ LAN, adapter của nút gửi sẽ tạo ra frame tầng liên kết dữ liệu chứa địa chỉ vật lý của nút nhận và gửi frame đó trên LAN. Nhưng làm thế nào để nút gửi xác định địa chỉ vật lý của nút có địa chỉ IP là 222.222.222.222? Nó sẽ đưa cho module ARP địa chỉ IP 222.222.222.222, sau đó module ARP trả lại địa chỉ vật lý tương ứng với địa chỉ IP được hỏi, là 49-DB-D2-C7-56-2A.



Hình 5.23 Mỗi nút trong mạng LAN có một địa chỉ IP, mỗi adapter của nút có một địa chỉ mạng LAN

Do vậy chúng ta thấy rằng ARP đã xác định địa chỉ vật lý từ địa chỉ IP. Trong khía cạnh nào đó, chức năng này tương tự DNS (đã nghiên cứu trong mục 2.5), DNS xác định địa chỉ IP từ tên máy tính. Tuy nhiên sự khác biệt rất lớn giữa hai dịch vụ này là DNS chuyển đổi tên mọi máy tính trên Internet. Ngược lại ARP chỉ chuyển đổi địa chỉ IP cho những nút trên cùng mạng LAN. Nếu một nút ở Hà Nội cố gắng dùng module ARP để xác định địa chỉ vật lý của một nút ở Huế thì chắc chắn module ARP sẽ trả lại một mã lỗi.

Bây giờ chúng ta xét module ARP làm việc như thế nào. Module ARP trong mỗi nút chứa một bảng ARP trong RAM của mình. Mỗi hàng của bảng là một ánh xạ giữa địa chỉ IP và địa chỉ vật lý.

Hình 5.24 minh họa bảng ARP của nút 222.222.222.220. Với mỗi ánh xạ trong bảng ARP có trường "thời gian sống" (TTL) cho chính ánh xạ đó, xác định thời gian tồn tại của ánh xạ trong bảng. Chú ý rằng bảng này không nhất thiết phải chứa tất cả các ánh xạ cho mọi nút trên LAN, ánh xạ có thể dần dần được thêm vào bảng. Thời gian sống của một ánh xạ thường là 20 phút kể từ khi ánh xạ được đưa vào bảng ARP.

IP address	LAN address	TTL
222.222.222.221	88-B2-2F-54-1A-0F	13:45:00
222.222.222.223	5C-66-AB-90-75-B1	13:52:00

Hình 5. 24 Bảng ARP của nút 222.222.222.220

Bây giờ, giả sử nút 222.222.222.220 muốn gửi gói dữ liệu IP đến một nút trên LAN. Nút gửi cần xác định địa chỉ vật lý từ địa chỉ IP của nút nhận. Công việc này đơn giản nếu bảng ARP của nút gửi chứa ánh xạ của nút nhận. Nhưng nếu bảng ARP đó không chứa ánh xạ tương ứng cho nút nhận thì sao? Giả sử nút có địa chỉ IP 222.222.222.220 muốn gửi gói dữ liệu tới nút 222.222.222.222. Khi đó nút gửi phải sử dụng giao thức ARP để giải mã địa chỉ. Đầu tiên, nút gửi tạo ra một gói đặc biệt gọi là **gói ARP (ARP packet)**. Trong gói ARP có trường chứa địa chỉ IP và địa chỉ vật lý của nút gửi, nút nhận. Cả gói truy vấn và trả lời ARP đều có chung khuôn dạng. Mục đích của gói truy vấn ARP là hỏi tất cả các nút khác trên LAN để xác định địa chỉ vật lý ứng với địa chỉ IP.

Trở lại ví dụ trên, nút 222.222.222.220 gửi gói truy vấn ARP đến adapter và yêu cầu adapter gửi tới tất cả các nút trên mạng LAN, có nghĩa là sử dụng địa chỉ quảng bá FF-FF-FF-FF-FF-FF. Adapter đặt gói ARP trong frame tầng liên kết dữ liệu, với địa chỉ đích của frame là địa chỉ quảng bá và gửi frame vào mạng LAN. Nó cũng giống như việc giáo viên vào lớp hỏi to "Sinh viên nào có tên là Trần Văn X hãy báo số thẻ sinh viên". Câu nói này lan tỏa khắp lớp học (được quảng bá), tất cả các sinh viên đều nghe thấy

nhưng chỉ có sinh viên Trần Văn X mới trả lời. Frame chứa truy vấn ARP được tất cả các adapter trên LAN nhận (do sử dụng địa chỉ quảng bá) và mỗi adapter gửi gói ARP trong frame lên bộ xử lý trung tâm của mình. Sau đó mỗi nút tự kiểm tra xem địa chỉ IP của mình có giống với địa chỉ IP đích trong gói ARP không. Chỉ có duy nhất nút phù hợp mới gửi gói ARP trả lời chứa ánh xạ yêu cầu. Sau đó nút gửi truy vấn (222.222.222.220) có thể cập nhật bảng ARP và gửi đi gói dữ liệu IP.

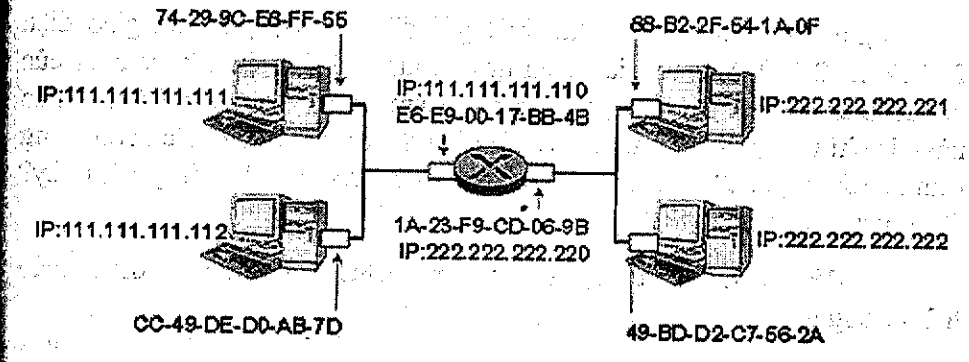
Có hai điểm cần chú ý trong giao thức ARP. Thứ nhất, thông điệp truy vấn ARP được gửi quảng bá trong khi thông điệp trả lời ARP được gửi trong frame bình thường. Thứ hai, ARP hoạt động theo kiểu "cắm vào là chạy" (plug and play) vì bảng ARP của nút được xây dựng tự động, không cần người quản trị thiết lập cấu hình. Và nếu một nút dừng kết nối với LAN thì ánh xạ tương ứng của nó cũng bị xoá khỏi bảng sau một khoảng thời gian nào đấy.

Gửi gói dữ liệu đến nút không nằm trong LAN

Chúng ta đã hình dung rõ ràng hoạt động của ARP khi một nút gửi gói dữ liệu đến nút khác nằm trong cùng mạng LAN. Bây giờ hãy xét tình huống phức tạp hơn khi nút muốn gửi dữ liệu tới nút nằm ngoài mạng LAN. Chúng ta xét ví dụ minh họa trên Hình 5.25 gồm hai mạng LAN kết nối với nhau qua router.

Có một số điểm cần chú ý trong Hình 5.25. Đầu tiên nút chia ra làm hai kiểu: máy tính và router. Mỗi máy tính có duy nhất một địa chỉ IP và một adapter. Như đã nói tới trong phần 4.4, mỗi giao diện ghép nối của router có một địa chỉ IP riêng. Mỗi giao diện của router cũng có module ARP (trong router) và adapter. Router trong Hình 5.25 có hai giao diện nên có hai địa chỉ IP, hai module ARP và hai adapter. Dĩ nhiên, mỗi adapter có một địa chỉ vật lý riêng.

Cũng chú ý rằng tất cả giao diện kết nối vào mạng LAN 1 có địa chỉ IP dạng 111.111.111.xxx và kết nối vào mạng LAN 2 có địa chỉ IP dạng 222.222.222.xxx. Trong ví dụ này, 3 byte đầu tiên của địa chỉ IP xác định địa chỉ mạng trong khi đó byte cuối cùng xác định nút cụ thể nào trong mạng (chính xác hơn là adapter).



Hình 5.25 Hai mạng LAN kết nối với nhau qua router

Bây giờ, giả sử rằng máy tính 111.111.111.111 muốn gửi gói dữ liệu IP đến máy tính 222.222.222.222. Như thường lệ, máy tính gửi sẽ chuyển gói dữ liệu xuống adapter của mình. Nhưng máy tính gửi cũng cần phải chỉ cho adapter biết địa chỉ vật lý đích thích hợp. Adapter sẽ dùng địa chỉ vật lý nào ? Có phải là địa chỉ vật lý của máy tính 222.222.222.222 - 49-BD-D2-C7-56-2A không? Hiển nhiên nếu adapter gửi sử dụng địa chỉ vật lý này thì chắc chắn không một adapter nào của LAN 1 sẽ gửi gói dữ liệu IP lên tầng mạng của mình, vì địa chỉ đích của frame không phù hợp với địa chỉ vật lý của bất kỳ adapter nào trên LAN 1, do đó gói dữ liệu sẽ mất.

Nếu quan sát trên Hình 5.25, chúng ta thấy rằng để gửi gói dữ liệu từ nút 111.111.111.111 đến nút khác trên LAN 2, đầu tiên gói dữ liệu phải được gửi đến giao diện router 111.111.111.110. Như đã thảo luận trong phần 4.4, bảng định tuyến của máy tính 111.111.111.111 sẽ chỉ ra rằng để đi đến máy tính 222.222.222.222, thì đầu tiên gói dữ liệu cần được gửi tới router (chính xác hơn là adapter của router) có địa chỉ 111.111.111.110. Như vậy, địa chỉ vật lý đích của frame là địa chỉ vật lý của giao diện router có địa chỉ 111.111.111.110 tức là E6-E-00-17-BB-4B. Làm thế nào máy tính gửi xác định được địa chỉ vật lý của 111.111.111.110?. Tất nhiên là bằng cách sử dụng ARP. Sau khi xác định được địa chỉ vật lý, nút gửi sẽ tạo ra frame và gửi frame vào LAN 1. Adapter của router trên LAN 1 sẽ nhận frame tầng liên kết dữ liệu gửi cho nó, và sẽ chuyển gói IP lên tầng mạng của router. Gói dữ liệu IP đã được chuyển từ máy tính nguồn đến router. Bây giờ, router phải xác định giao diện để gửi dữ liệu đi tiếp. Như đã nêu trong phần 4.4, công việc này được thực hiện bằng cách tra cứu bảng định tuyến của router.

Bảng định tuyến của router cho biết gói dữ liệu cần gửi qua giao diện 222.222.222.220. Sau đó giao diện này sẽ gửi gói dữ liệu đến adapter của nó, adapter này đặt gói dữ liệu trong một frame mới và gửi vào LAN 2. Lúc này địa chỉ vật lý đích trong frame gửi đi là địa chỉ vật lý của đích cuối cùng (nút 222.222.222.222). Làm thế nào router có được địa chỉ vật lý đích này? Tất nhiên là dựa vào ARP.

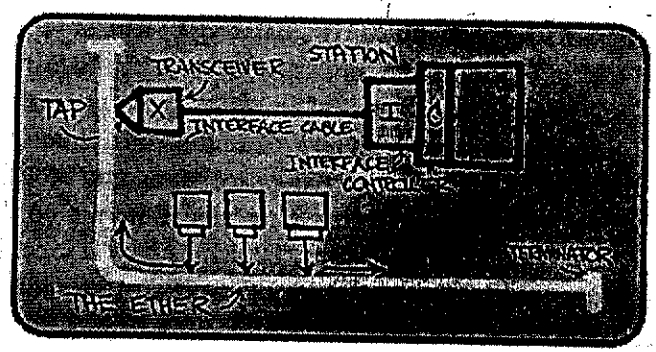
ARP cho Ethernet được đặc tả trong RFC 826 và có thể tham khảo thêm trong RFC 1180.

5.5. ETHERNET

Hiện nay Ethernet thống trị thị trường mạng cục bộ. Mới chỉ đầu những năm 1980 đến đầu những năm 1990, Ethernet còn có nhiều đối thủ khác như FDDI, token-ring, ATM. Một số công nghệ đã thành công trong việc chiếm lĩnh một thị phần nào đó trong vài năm. Nhưng từ khi ra đời vào giữa những năm 70, Ethernet liên tục phát triển để rồi dần dần chiếm lĩnh phần lớn thị phần. Ngày nay, Ethernet là công nghệ vượt xa các công nghệ LAN khác và khả năng này khó có thể bị đảo lộn trong tương lai gần.

Có rất nhiều lý do dẫn đến sự thành công của Ethernet. Thứ nhất, Ethernet là mạng cục bộ tốc độ cao được triển khai rộng rãi đầu tiên. Được triển khai tương đối sớm nên các nhà quản trị mạng lập tức trở nên quen thuộc với Ethernet (ưu điểm, các đặc tính...) - và ngại chuyển sang những công nghệ LAN mới. Thứ hai token-ring, FDDI và ATM phức tạp và đắt hơn Ethernet. Thứ ba, lý do chính đáng nhất để sử dụng các công nghệ LAN khác (FDDI hay ATM) là do công nghệ mới có tốc độ cao hơn, tuy nhiên Ethernet luôn luôn "phản công" lại bằng cách liên tục nâng cấp tốc độ. Ethernet dạng chuyển mạch được phát minh vào đầu những năm 90 với tốc độ rất cao. Cuối cùng vì Ethernet quá phổ biến, nên phần cứng Ethernet (đặc biệt là card mạng) cũng hết sức phổ biến và rẻ. Giá cả thấp này cũng do giao thức đa truy cập của Ethernet - CSMA/CD hoàn toàn phân tán nên có thiết kế đơn giản.

Kiến trúc Ethernet (Hình 5.26) được Bob Metcalf và David Boggs đưa ra vào khoảng giữa những năm 70.



Hình 5.26 Thiết kế đầu tiên của Ethernet

Ngày xưa

Bob Metcalfe và Ethernet

Khi còn là nghiên cứu sinh ở trường đại học Harvard vào đầu những năm 1970, Bob Metcalfe làm việc cho ARRAnet tại MIT và đã thực sự ấn tượng về công trình của Abramson về giao thức truy cập ngẫu nhiên. Sau khi hoàn thành luận án tiến sỹ và ngay trước khi bắt đầu công việc mới ở Xero Palo Alto Research Center (Xerox PARC), Bob đã tới thăm Abrasmon và các đồng nghiệp tại trường đại học Hawaii trong 3 tháng và nhìn tận mắt Alohanet. Tại Xero PARC, Metcalfe đã làm quen với máy tính Alto (về khía cạnh nào đó có thể coi là tiền thân của máy tính cá nhân PC vào đầu thập niên 80). Metcalfe đã nhận thức được nhu cầu kết nối mạng cho các máy tính này với một chi phí không đắt. Nên với những kiến thức về ARPNet, Alohanet và giao thức truy cập ngẫu nhiên, Metcalfe và đồng nghiệp David Bogg đã phát minh ra Ethernet.

Kiến trúc Ethernet đầu tiên của Metcalfe và Bogg hoạt động với tốc độ 2.94 Mbps và có thể kết nối 256 máy trong phạm vi 1 dặm. Metcalfe và Bogg đã thành công trong việc kết nối các máy tính Alto. Sau đó Metcalfe đi đầu trong việc xây dựng liên minh giữa Xerox, Digital và Intel để thiết lập chuẩn Ethernet với tốc độ 10 Mbps và được IEEE thông qua. Năm 1979 Metcalfe thành lập công ty riêng là 3Com, với mục tiêu phát triển và thương mại hóa các công nghệ mạng, kể cả công nghệ Ethernet. Cụ thể 3Com đã bán sản phẩm card mạng Ethernet cho các máy tính IBM PC ngày càng trở nên phổ biến. Metcalfe rời 3Com khi 3Com có 2000 nhân viên, trị giá 400 triệu đô la thu nhập. Tháng 1 năm 2000, 3Com có vốn lên tới 15 tỷ USD và 13.000 công nhân.

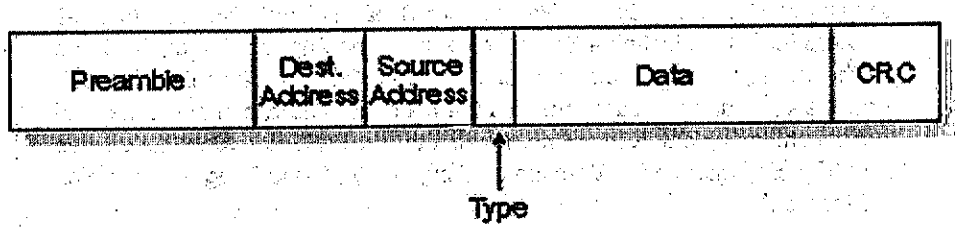
5.5.1 Những khái niệm cơ bản của Ethernet

Ngày nay Ethernet xuất hiện dưới nhiều hình thức. Mạng cục bộ Ethernet có thể có topo dạng bus hay dạng sao. Mạng cục bộ Ethernet có thể sử dụng cáp đồng trục hay cáp quang. Hơn nữa, Ethernet có thể truyền dữ liệu với các tốc độ khác nhau: 10Mbps, 100Mbps hay 1Gbps. Nhưng dù là Ethernet kiểu nào, tất cả các công nghệ Ethernet đều có một số đặc trưng quan trọng sau đây:

Cấu trúc frame Ethernet

Các công nghệ Ethernet khác nhau đều có chung cấu trúc frame. Cho dù công nghệ Ethernet sử dụng cáp đồng trục hay cáp quang, chạy với tốc độ 10Mbps, 100Mbps hay 1Gbps thì cấu trúc frame đều như nhau.

Frame Ethernet được minh hoạ trong Hình 5.27. Kiến thức về khuôn dạng frame Ethernet sẽ giúp chúng ta hiểu kỹ về Ethernet. Chúng ta xét việc gửi gói dữ liệu IP giữa hai máy tính trên cùng một mạng cục bộ Ethernet (Chú ý rằng Ethernet cũng có thể mang các gói dữ liệu tầng mạng khác IP). Giả sử A là adapter gửi có địa chỉ vật lý AA-AA-AA-AA-AA-AA và adapter nhận B có địa chỉ vật lý là BB-BB-BB-BB-BB-BB.



Hình 5.27 Khuôn dạng gói tin Ethernet

Adapter gửi đặt gói dữ liệu (IP datagram) trong frame Ethernet và gửi frame này xuống tầng vật lý. Adapter nhận sẽ nhận frame từ tầng vật lý, lấy ra gói dữ liệu IP và chuyển lên tầng mạng phía trên. Ở đây chúng ta chỉ nghiên cứu 6 trường trong frame Ethernet:

Trường dữ liệu (từ 46 đến 1500 byte): trường này chứa gói dữ liệu IP, MTU (Maximum Transfer Unit) của Ethernet là 1500 byte. Điều này có nghĩa là nếu gói dữ liệu IP vượt quá 1500 byte thì máy tính phải chia nhỏ gói dữ liệu ra (xem 4.4.4). Kích thước tối thiểu của trường này là 46 byte.

Điều này có nghĩa là nếu gói dữ liệu nhỏ hơn 46 byte, trường dữ liệu phải được "chèn" thêm một số dữ liệu giả cho đủ 46 byte. Khi bên gửi chèn thêm dữ liệu vào thì tầng mạng ở bên nhận cũng nhận được cả gói dữ liệu IP lẫn dữ liệu được chèn thêm vào, khi đó nó phải sử dụng trường độ dài trong gói dữ liệu IP để loại bỏ phần thêm vào.

Địa chỉ đích (6 byte): Trường này chứa địa chỉ vật lý của adapter nhận (chẳng hạn BB-BB-BB-BB-BB-BB). Khi adapter B nhận bất kỳ frame nào, nó sẽ kiểm tra địa chỉ đích của frame. Nếu địa chỉ đích là BB-BB-BB-BB-BB-BB (địa chỉ của chính nó), hoặc địa chỉ quảng bá LAN (FF-FF-FF-FF-FF-FF) thì adapter mới chuyển gói tin datagram trong trường dữ liệu lên tầng mạng. Nếu không adapter sẽ loại bỏ frame (trong trường hợp này frame được gửi tới một adapter khác).

Địa chỉ nguồn (6 byte): Trường này chứa địa chỉ vật lý của adapter gửi frame, trong ví dụ này là AA-AA-AA-AA-AA-AA.

Trường kiểu (2 byte): Trường này cho phép Ethernet hỗ trợ nhiều giao thức tầng mạng khác nhau. Cần chú ý rằng máy tính có thể sử dụng nhiều giao thức tầng mạng (không chỉ có IP). Trên thực tế, máy tính nào đó có thể hỗ trợ nhiều giao thức tầng mạng và sử dụng các giao thức khác nhau cho những ứng dụng khác nhau. Vì thế khi nhận được một frame Ethernet, adapter B cần xác định giao thức tầng mạng nào sẽ nhận nội dung của trường dữ liệu. Những giao thức tầng mạng như IP, Novell IPX hoặc AppleTalk đều có một mã định danh (là một số) đã được chuẩn hóa. Hơn nữa, giao thức ARP cũng có một định danh. Trường kiểu tương tự trường protocol trong gói dữ liệu IP hay trường số hiệu cổng trong tầng giao vận; mục đích của tất cả các trường này là kết hợp giao thức ở tầng dưới với giao thức ở tầng trên nó.

Mã kiểm tra dư thừa vòng (Cyclic Redundancy Check -CRC) (4 byte). Như đã nêu trong phần 5.2.3, mục đích của trường CRC là cho phép adapter phát hiện liệu có lỗi nào trong frame nhận được hay không. Có nhiều nguyên nhân lỗi bit, chẳng hạn suy hao năng lượng điện từ của tín hiệu; tỏa nhiệt trong card Ethernet hay cáp mạng. Việc phát hiện lỗi được thực hiện như sau: Khi tạo ra frame Ethernet, máy tính A tính giá trị trường CRC dựa trên trường dữ liệu thực sự. Công việc kiểm tra tại B xem dữ liệu

thực sự và CRC có mâu thuẫn không được gọi là **CRC check**. Nếu việc kiểm tra CRC thất bại (nghĩa là nếu giá trị trường CRC không phù hợp với phần dữ liệu) thì máy tính B xác định trong frame có lỗi.

Lời mở đầu (preamble) (8 byte): Frame Ethernet bắt đầu với trường preamble 8 byte, trong đó bảy byte đầu tiên có giá trị là 10101010; byte thứ tám có giá trị 10101011. Bảy byte đầu tiên của phần mở đầu làm nhiệm vụ "đánh thức" adapter nhận và đồng bộ hoá đồng hồ bên gửi với đồng hồ bên nhận. Tại sao các đồng hồ lại không đồng bộ hoá ? Chú ý rằng adapter A truyền frame với tốc độ 10Mbps, 100Mbps hay 1Gbps phụ thuộc vào kiểu Ethernet. Tuy nhiên, adapter A chưa chắc đã truyền frame với tốc độ xác định mà với tốc độ nào đó. Adapter nhận có thể chốt đồng hồ của adapter A bằng cách chốt tất cả các bit trong bảy byte đầu tiên. Hai bit cuối cùng trong byte thứ 8 (hai bit 1 liên tiếp nhau) báo cho adapter B biết rằng "dữ liệu quan trọng" chuẩn bị đến. Khi máy tính B thấy hai bit 1 liên tiếp nhau, nó biết rằng 6 byte tiếp theo là địa chỉ đích. Adapter có thể phát hiện frame đã được truyền xong khi không thấy dòng điện.

Dịch vụ không hướng nối, không tin cậy

Tất cả công nghệ Ethernet cung cấp cho tầng mạng **dịch vụ không hướng nối**. Nghĩa là khi adapter A muốn gửi gói dữ liệu đến adapter B, adapter A sẽ đặt gói dữ liệu trong frame và gửi frame đó vào LAN mà không cần "bắt tay" trước với adapter B. Dịch vụ không kết nối ở tầng 2 này tương tự với dịch vụ IP ở tầng 3 và dịch vụ UDP ở tầng 4.

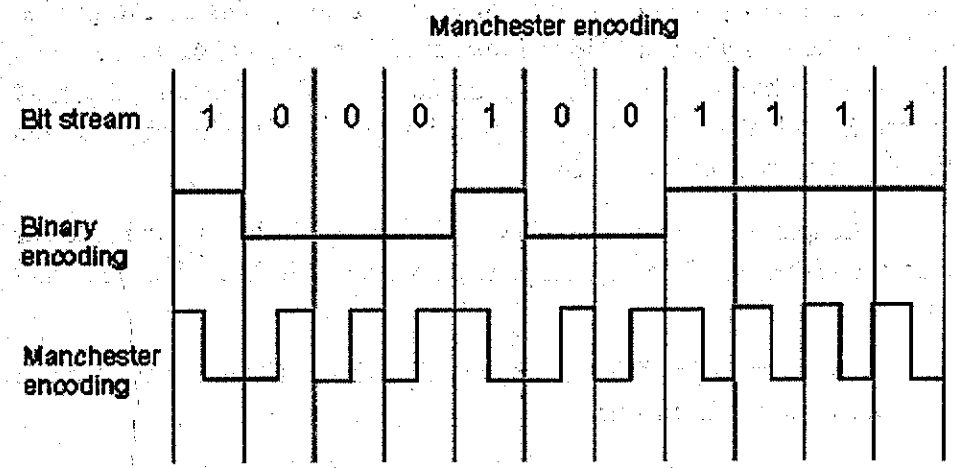
Công nghệ Ethernet cung cấp **dịch vụ không tin cậy** cho tầng mạng. Cụ thể, khi nhận được frame từ adapter A, adapter B sẽ không gửi phản hồi cho A. Adapter A không thể xác định liệu frame nó truyền đi có được nhận đúng hay không. Nếu phát hiện lỗi khi kiểm tra CRC, adapter B sẽ loại bỏ frame. Chính điều này giúp Ethernet đơn giản và rẻ. Nhưng dòng dữ liệu chuyển tới tầng mạng có thể bị gián đoạn.

Nếu có sự gián đoạn do một số frame Ethernet bị loại bỏ, giao thức tầng ứng dụng tại máy B có phát hiện được sự gián đoạn đó không ? Như đã trình bày trong chương 3, điều này phụ thuộc việc ứng dụng sử dụng UDP hay TCP. Nếu ứng dụng dùng UDP thì giao thức tầng ứng dụng trong máy B

sẽ không phát hiện được gián đoạn trong dữ liệu. Mặt khác, nếu ứng dụng dùng TCP thì thực thể TCP trong máy B sẽ không gửi biên nhận cho những dữ liệu đã bị loại bỏ, do vậy thực thể TCP trong máy A sẽ gửi lại. Chú ý rằng khi TCP gửi lại dữ liệu, thì cuối cùng dữ liệu cũng sẽ đi qua các adapter Ethernet. Và như vậy Ethernet truyền lại dữ liệu. Tuy nhiên Ethernet không biết rằng nó đang truyền lại mà coi rằng đó là một gói dữ liệu mới.

Giải tần cơ sở và mã hoá Manchester

Ethernet sử dụng băng tần cơ sở (baseband) nghĩa là adapter gửi tín hiệu số trực tiếp vào kênh truyền dùng chung. Card giao diện không dịch chuyển tín hiệu sang dải tần số khác như trong ASDL và các hệ thống cáp modem. Ethernet sử dụng mã hoá Manchester (Hình 5.28). Trong phương pháp mã hoá Manchester, mỗi bít ứng với một quá trình chuyển trạng thái (transition): bit 1 chuyển từ trên xuống dưới, bit 0 chuyển từ dưới lên trên. Lý do sử dụng mã hoá Manchester là đồng hồ của adapter gửi và nhận không đồng bộ hoàn toàn với nhau. Khi xuất hiện sự chuyển ngay trong phần giữa mỗi bit, máy tính nhận có thể đồng bộ đồng hồ của nó với đồng hồ của máy tính gửi. Sau khi đồng hồ của adapter nhận được đồng bộ hoá, phía nhận có thể thu được tín hiệu của mỗi bit và xác định nó là 0 hay 1. Mã hoá Manchester được sử dụng nhiều trong tầng vật lý chứ không phải trong tầng liên kết dữ liệu.



Hình 5.28 Mã hóa Manchester

5.2.2 CSMA/CD: Giao thức đa truy cập của Ethernet

Các nút trên mạng cục bộ Ethernet được kết nối qua một kênh truyền quảng bá dùng chung, vì vậy khi adapter gửi đi một frame, tất cả các adapter trên LAN đều nhận được frame. Như chúng ta đã đề cập trong phần 5.3, Ethernet dùng thuật toán đa truy cập CSMA/CD. Chú ý rằng CSMA/CD sử dụng các cơ chế sau:

- Adapter có thể bắt đầu truyền tại bất kì thời điểm nào, nghĩa là không chia khoảng thời gian.

- Adapter không bao giờ truyền frame khi thấy có adapter khác đang truyền (cảm nhận sóng mang).

- Adapter đang truyền chấm dứt truyền ngay khi phát hiện ra adapter khác cũng đang truyền (phát hiện xung đột).

- Trước khi cố gắng thử truyền lại, adapter đợi một khoảng thời gian ngẫu nhiên tương đối nhỏ.

Những cơ chế này giúp hiệu suất của CSMA/CD được cải thiện đáng kể so với slotted ALOHA khi vận hành trong môi trường LAN. Trong thực tế, nếu thời gian để tín hiệu lan truyền giữa hai nút là rất nhỏ thì hiệu suất của CSMA/CD có thể đạt tới 100%. Nhưng chú ý rằng cơ chế thứ hai và thứ ba kể trên yêu cầu adapter Ethernet có khả năng (1) cảm nhận được khi nào thì có một adapter khác đang truyền và (2) phát hiện xung đột trong khi truyền. Adapter Ethernet thực hiện hai nhiệm vụ này bằng việc đo mức điện áp trước và trong khi truyền.

Adapter dùng giao thức CSMA/CD không cần kết hợp với adapter khác trên Ethernet. Trên một adapter, giao thức CSMA/CD làm việc như sau:

- Adapter nhận PDU tầng mạng, tạo ra frame Ethernet và đặt frame vào trong bộ đệm của adapter.

- Nếu adapter cảm nhận kênh truyền rỗi (không có năng lượng tín hiệu trên kênh truyền) thì adapter bắt đầu truyền. Nếu adapter thấy kênh

truyền bận, nó sẽ đợi cho đến khi không phát hiện được năng lượng tín hiệu và sau đó bắt đầu truyền.

- Trong khi truyền, adapter kiểm tra xem có năng lượng tín hiệu đến từ adapter khác hay không. Nếu sau khi đã truyền xong frame mà không phát hiện được năng lượng trên đường truyền thì có thể xem frame được truyền thành công.

- Nếu adapter phát hiện năng lượng tín hiệu từ adapter khác trong khi đang truyền thì lập tức nó dừng lại không truyền và gửi đi tín hiệu báo nhiễu 48 bit (jam signal).

Sau khi dừng phát và gửi tín hiệu báo nhiễu, adapter sẽ thực hiện thuật toán **exponential backoff**. Khi truyền frame nào đó, nếu thấy frame đó bị xung đột n lần liên tiếp, adapter chọn một giá trị ngẫu nhiên K trong khoảng $(0,1,2.,..2^m - 1)$ với m = min(n,10). Sau đó adapter sẽ đợi K*512 trước khi quay lại bước 2.

Sau đây chúng ta sẽ giải thích về giao thức CSMA/CD. Mục đích của tín hiệu báo nhiễu là bảo đảm tất cả các adapter đang truyền khác đều phát hiện ra xung đột. Xét ví dụ sau: giả sử adapter A bắt đầu truyền đi một frame và ngay trước khi tín hiệu từ A tới được adapter B, adapter B bắt đầu truyền. Do vậy B chỉ truyền được vài bit trước khi dừng lại không truyền tiếp. Vài bit này sẽ lan tỏa được đến A, nhưng chúng không tạo đủ năng lượng để A có thể phát hiện xung đột. Để đảm bảo A phát hiện được xung đột, B phải truyền thêm tín hiệu báo nhiễu dài khoảng 48 bit.

Tiếp theo ta xét tới thuật toán exponetial backoff. Cần chú ý rằng thời gian để truyền đi một bit rất nhỏ: với tốc độ Ethernet 10Mbps, thời gian này là 0,1 microsecond. Xét ví dụ sau: giả sử adapter lần đầu tiên truyền đi một frame và trong khi truyền phát hiện có xung đột. Sau đó adapter sẽ chọn K=0 với xác suất 0,5 và chọn K=1 với xác suất 0,5. Nếu adapter chọn K=0 thì ngay lập tức nó sẽ nhảy đến bước 2 sau khi truyền đi tín hiệu báo nhiễu. Nếu adapter chọn K=1 thì nó sẽ đợi 51,2 microsecond trước khi quay lại bước 2. Sau xung đột lần thứ hai, K được chọn ngẫu nhiên giữa các giá trị (0,1,2,3) với xác suất bằng nhau, sau ba xung đột K sẽ được chọn ngẫu nhiên giữa các giá trị (0,1,2,…,7) với xác suất bằng nhau, sau nhiều hơn

mười xung đột K được chọn ngẫu nhiên giữa các giá trị (0,1,2,...,1023) với xác suất bằng nhau. Như vậy tổng số giá trị mà K có thể lựa chọn tăng theo luỹ thừa cơ số 2 với số mũ là số lần xung đột cho (cho đến khi N=10).

Chuẩn Ethernet ấn định giới hạn khoảng cách giữa hai nút bất kỳ. Giới hạn này bảo đảm rằng nếu adapter A chọn giá trị K thấp hơn giá trị K của tất cả các adapter khác liên quan đến xung đột trong pha trước thì A có thể truyền đi frame mà không bị xung đột nữa.

Tại sao lại sử dụng thuật toán exponetial backoff ? Tại sao không chọn K trong khoảng {0,1,2,3,4,5,6,7} sau mọi xung đột. Lý do sau khi adapter gặp xung đột lần đầu tiên, nó không hình dung được có bao nhiêu adapter liên quan đến xung đột đó. Nếu chỉ có một số lượng nhỏ adapter thì chắc chắn K sẽ được chọn trong một tập hợp hạn chế. Ngược lại, nếu có nhiều adapter liên quan thì K được chọn trong một tập hợp lớn hơn. Bằng cách tăng kích cỡ của tập hợp sau mỗi xung đột, adapter sẽ thích nghi được với nhiều hoàn cảnh.

Chúng ta cần chú ý thêm rằng mỗi lần adapter chuẩn bị frame mới để gửi đi, nó sử dụng thuật toán CSMA/CD nói trên. Cụ thể adapter không quan tâm tới bất kỳ xung đột nào trước đó. Do vậy, rất có khả năng adapter với frame mới có thể truyền xen vào trong khi một vài adapter khác đang trong trạng thái exponetial backoff.

Hiệu suất Ethernet

Khi chỉ có một nút có frame để truyền, nút đó có thể truyền với tốc độ tối đa (10Mbps, 100Mbps hoặc 1Gbps). Tuy nhiên nếu nhiều nút cùng truyền thì tốc độ truyền thành công (effective rate) của kênh truyền có thể giảm đi đáng kể. Chúng ta định nghĩa **hiệu suất (efficiency)** của Ethernet là tỉ lệ thời gian không có xung đột trên kênh truyền khi có nhiều nút tích cực, mỗi nút cần truyền nhiều frame trong một khoảng thời gian dài. Để xác định hiệu suất gần đúng của Ethernet, giả sử t_{prop} là thời gian lớn nhất năng lượng tín hiệu lan tỏa giữa hai adapter. Giả sử t_{trans} là thời gian để truyền đi một frame Ethernet với độ lớn cực đại (xấp xỉ 1,2 ms với Ethernet 10 Mbps). Có thể xem [Lam 1980] và [Bertsekas 1991] để xác định công thức tính. Ở đây chúng ta sử dụng công thức:

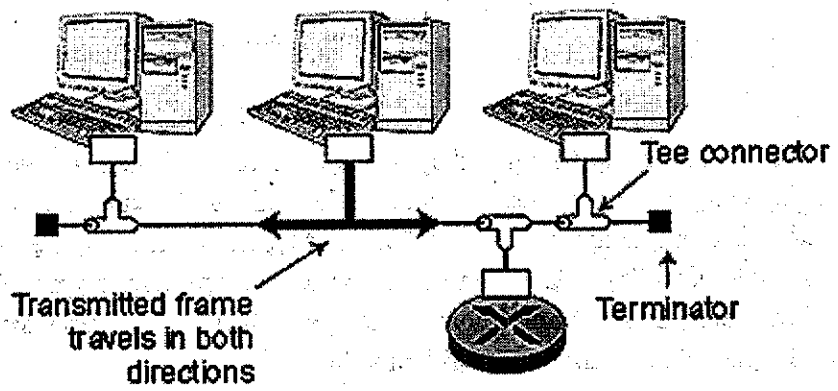
$$\text{Efficiency} = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

Từ công thức này chúng ta thấy nếu t_{prop} đạt tới 0 thì hiệu suất đạt tới 1. Điều này cũng rất hợp lý: nếu thời gian trễ là 0 thì các nút xung đột sẽ lập tức bỏ dở mà không lãng phí kênh truyền. Khi t_{trans} trở lên rất lớn, hiệu suất đạt tới 1. Điều này cũng hiển nhiên vì khi frame có được kênh truyền nó sẽ chiếm dụng kênh truyền trong khoảng thời gian dài, như vậy kênh truyền hầu như lúc nào cũng trong trạng thái làm việc.

5.5.3 Những công nghệ Ethernet

Phần lớn những công nghệ Ethernet phổ biến ngày nay là **10Base2** sử dụng cáp đồng trục gầy (thin coaxial cable) có topo dạng bus, tốc độ truyền là 10Mbps; **10BaseT** sử dụng cáp đồng trục, topo hình sao, tốc độ truyền là 10 Mbps; **100BaseT** sử dụng dây đồng xoắn, topo hình sao, tốc độ truyền là 100Mbps; Gigabyte Ethernet sử dụng cả sợi quang hay dây đồng xoắn, truyền với tốc độ 1Gbps. Những công nghệ Ethernet này được chuẩn hoá bởi IEEE 802.3. Vì thế LAN Ethernet thường được gọi là 802.3 LAN.

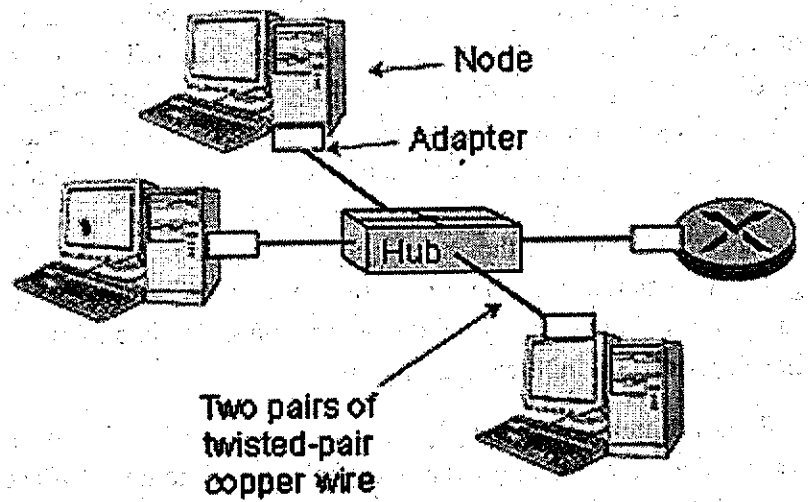
Trước khi tiếp tục, chúng ta cần nói về **bộ tiếp sức (repeater)** - được sử dụng phổ biến trong mạng LAN cũng như các đường truyền trên khoảng cách xa. Repeater là thiết bị tầng vật lý xử lý trên từng bit riêng lẻ chứ không phải trên frame. Khi tín hiệu (biểu diễn bit 0 hoặc 1) đến từ một cổng, repeater thường tái tạo lại tín hiệu này bằng cách gia tăng cường độ năng lượng của tín hiệu và gửi tín hiệu đó qua tất cả các cổng còn lại. Repeater được sử dụng rộng rãi trong LAN để mở rộng phạm vi địa lý. Cần chú ý rằng trong Ethernet, repeater không có khả năng cảm nhận sóng mang hay thực hiện bất kỳ một chức năng nào của CSMA/CD, repeater chỉ tái tạo và gửi tín hiệu đến từ một cổng ra tất cả các cổng khác, kể cả trong trường hợp các cổng kia cũng đang có tín hiệu để truyền.



Hình 5.29 Ethernet 10Base2

10Base2 là một công nghệ Ethernet rất phổ biến. "10" trong 10Base2 có nghĩa là tốc độ truyền trong công nghệ này là 10 Mbps. "2" có ý nghĩa khoảng cách tối đa giữa hai trạm không có repeater ở giữa không vượt quá 200m. Hình 5.29 minh họa mạng Ethernet 10Base2. 10Base2 có topo dạng bus, các nút (chính xác hơn là các adapter) được kết nối trực tiếp vào một môi trường dùng chung – cáp đồng trục gầy (là loại cáp tương tự như cáp truyền hình nhưng mỏng và nhẹ hơn). Khi adapter gửi đi một frame, frame sẽ được truyền qua đầu nối chữ T (T connector). Sau đó frame sẽ lan tỏa theo hai hướng của dây dẫn. Trên đường đi, mỗi adapter sẽ nhận được một bản sao của frame (chính xác hơn là các adapter thu được các tín hiệu của frame). Khi đến điểm cuối cùng của dây dẫn, tất cả các tín hiệu sẽ bị terminator hấp thụ (triệt tiêu). Chú ý rằng, do tất cả adapter đều có khả năng nhận mọi frame nên 10Base2 rõ ràng là môi trường quảng bá.

Nếu không có repeater, độ dài tối đa của bus là 185m. Nếu bus có độ dài lớn hơn, suy hao tín hiệu sẽ làm hệ thống hoạt động không chính xác. Ngoài ra nếu không có repeater, số lượng tối đa các nút là 30. Người ta sử dụng repeater để nối các đoạn 10Base2 liên tiếp nhau, mỗi đoạn có thể có 30 máy và dài 185m. Chỉ có thể sử dụng tối đa 4 repeater – do đó có 5 đoạn 10Base2.



Hình 5.30 Topo dạng sao 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 10Mbps trong khi tốc độ truyền của Ethernet 100BaseT là 100Mbps. 10BaseT và 100BaseT là công nghệ được sử dụng rất phổ biến hiện nay. Chúng có topo dạng sao, như minh họa trên Hình 5.30. Trong topo hình sao có một thiết bị trung tâm được gọi là **hub** (đôi khi gọi là **bộ tập trung - concentrator**). Adapter trên mỗi nút có kết nối trực tiếp đến hub. Kết nối này gồm hai cặp dây đồng xoắn đôi, một để truyền và một để nhận. Tại mỗi đầu của kết nối có một connector (bộ nối) RJ-45 - giống như connector RJ-45 được sử dụng cho điện thoại thông thường. Chữ "T" trong 10BaseT và 100BaseT là viết tắt của "Twisted pair". Đối với 10BaseT và 100BaseT, khoảng cách tối đa giữa adapter và hub là 100m, vì vậy độ dài lớn nhất giữa hai nút là 200m. Chúng ta sẽ trình bày trong phần sau, khoảng cách này có thể được tăng nếu sử dụng các thiết bị như hub, bridge, switch.

Về bản chất, hub là repeater vì khi nhận được tín hiệu từ adapter, hub sẽ gửi tín hiệu đó đến tất cả các adapter khác. Theo cách này, mỗi adapter có thể (1) cảm nhận kênh truyền để xác định liệu kênh truyền có rỗi không và (2) phát hiện xung đột trong khi đang truyền dữ liệu. Nhưng hub được dùng phổ biến do có khả năng trợ giúp việc quản trị mạng. Ví dụ, nếu

adapter trục trặc và tiếp tục gửi frame Ethernet (gọi là jabbering adapter) thì mạng 10Base2 Ethernet sẽ sụp đổ vì không adapter nào có khả năng truyền thông nữa. Nhưng điều này không xảy ra với mạng 10BaseT vì hub sẽ phát hiện vấn đề và ngưng kết nối tới adapter đang trục trặc. Tính năng này có tính chất tự động, có nghĩa là không cần sự can thiệp bằng tay của người quản trị mạng. Hơn thế nữa, hầu hết hub có thể thu thập và báo cáo thông tin đến máy tính có kết nối trực tiếp đến hub. Máy tính kiểm tra này sẽ sử dụng giao diện đồ hoạ hiển thị các thông tin trạng thái của hub như băng thông, tỉ lệ xung đột, kích thước trung bình của frame v.v. Người quản trị mạng có thể sử dụng thông tin này không chỉ để kiểm tra và khắc phục lỗi mà còn để lập kế hoạch phát triển LAN trong tương lai.

Nhiều adapter Ethernet ngày nay là adapter 10/100 Mbps. Tức là chúng ta có thể sử dụng được dùng cả hai kiểu Ethernet: 10BaseT và 100BaseT. 100BaseT, đặc trưng của nó là sử dụng loại cáp xoắn kiểu 5 (loại cáp chất lượng cao với nhiều vỏ). Khác 10Base2 và 10BaseT, 100BaseT không sử dụng phương pháp mã hoá Manchester mà sử dụng phương pháp 4B5B có hiệu suất cao hơn: mỗi nhóm 5 chu kỳ đồng hồ được sử dụng để mã hóa 4 bit và cung cấp đủ thông tin cho phép đồng bộ hoá đồng hồ.

Chú ý rằng cả hai công nghệ 10BaseT và 100BaseT đều có thể sử dụng cáp quang, Cáp quang thường được sử dụng để kết nối đến hub. Giá thành của cáp quang cao do giá thành connector nhưng ưu điểm là khả năng chống nhiễu tuyệt vời. Chuẩn IEEE 802 cho phép LAN có thể trải rộng trên vùng địa lí lớn nếu sử dụng cáp quang để nối các nút nằm trên trục chính (backbone).

Gigabit Ethernet

Gigabit Ethernet là sự mở rộng của chuẩn Ethernet 10BaseT và 100BaseT. Với tốc độ truyền dữ liệu dạng thô là 1000 Mbps, Gigabit Ethernet vẫn duy trì khả năng tương thích hoàn toàn với các thiết bị Ethernet kiểu cũ. Chuẩn Gigabit Ethernet (IEEE802.3x), thực hiện các công việc sau:

- Sử dụng khuôn dạng frame Ethernet chuẩn (Hình 5.27), tương thích với công nghệ 10BaseT và 100BaseT. Điều này cho phép dễ dàng tích hợp Gigabit Ethernet vào các cơ sở đã cài đặt các thiết bị Ethernet

- Cho phép đường truyền point-to-point cũng như kênh truyền quảng bá dùng chung. Đường truyền poit-to-point dùng switch (xem phần 5.6) trong khi kênh truyền quảng bá sử dụng hub giống 10BaseT và 100BaseT. Trong thuật ngữ Gigabit Ethernet, hub được gọi là "buffered distributors".

- Sử dụng CSMA/CD cho kênh truyền quảng bá dùng chung. Để đạt được hiệu suất mong muốn, khoảng cách lớn nhất giữa các nút bị hạn chế chặt chẽ.

- Kênh truyền point-to-point có đặc tính song công, mỗi hướng truyền với tốc độ 1 Gbps.

- Giống 10BaseT và 100BaseT, Ethernet Gigabit có topo dạng sao với hub hoặc switch ở trung tâm (switch Ethernet sẽ được thảo luận trong phần 5.6). Gigabit Ethernet thường được sử dụng trên các backbone (trục chính) kết nối nhiều mạng cục bộ Ethernet 10BaseT và 100BaseT. Gigabit Ethernet có thể sử dụng loại cáp 5UTP hoặc cáp quang.

5.6 HUB, BRIDGE VÀ SWITCH

Cơ quan - các công ty, trường đại học – có đặc điểm gồm nhiều bộ phận con, mỗi bộ phận có mạng cục bộ Ethernet riêng. Tất nhiên, cơ quan muốn kết nối mạng cục bộ của các bộ phận. Trong mục này chúng ta nghiên cứu một số hướng tiếp cận để kết nối các LAN với nhau. Chúng ta sẽ nghiên cứu ba hướng tiếp cận: sử dụng hub, bridge và switch.

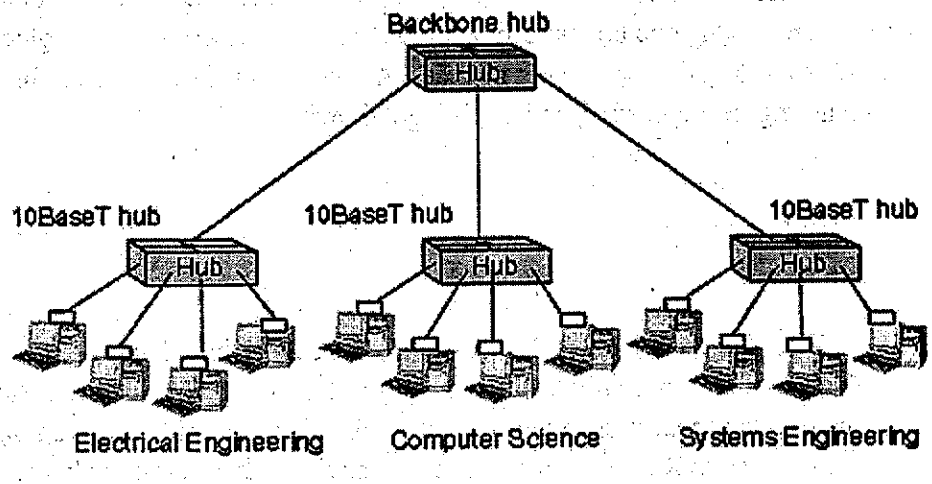
5.6.1 Hub

Cách đơn giản nhất để kết nối LAN là sử dụng hub. Hub là thiết bị đơn giản sao chép tín hiệu đến từ một cổng ra tất cả các cổng còn lại. Bản chất của hub là repeater, 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.

Hình 5.31 minh họa kết nối mạng LAN của ba khoa trong một trường đại học qua hub. Mỗi khoa có một mạng Ethernet 10BaseT để cán bộ và sinh viên của khoa sử dụng. Mỗi máy tính của khoa kết nối điểm nối điểm đến hub. Hub thứ tư, được gọi là backbone hub (hub trục chính) có kết nối điểm nối điểm đến các hub của khoa được sử dụng để liên kết LAN của ba khoa. Thiết kế được chỉ ra trong Hình 5.31là **thiết kế hub nhiều tầng (multi-tier hub design)** vì các hub được tổ chức trong hệ thống phân cấp. Có thể tạo thiết kế nhiều tầng - ví dụ, một tầng dành cho cấp Khoa, một tầng dành cho các trường trong trường đại học lớn (ví dụ: Trường công nghệ, trường kinh tế ...) và một tầng ứng với mức cao nhất của trường.

Trong thiết kế nhiều tầng, chúng ta coi toàn bộ mạng liên kết với nhau là mạng cục bộ LAN và coi mỗi phân mạng LAN ứng với một khoa (nghĩa là hub của khoa và các máy tính nối tới hub đó) là **LAN segment**. Chú ý rằng tất cả LAN segment trong Hình 5.31 thuộc về cùng một vùng xung đột, nghĩa là bất cứ lúc nào nhiều nút trên LAN truyền dữ liệu tại cùng một thời điểm thì sẽ phát sinh xung đột và tất cả những nút liên quan bắt đầu quá trình "exponential backoff".

Mạng cục bộ cấp khoa liên kết tới hub trục chính có nhiều ưu điểm. Đầu tiên và quan trọng nhất, nó cung cấp môi trường truyền thông giữa các khoa với nhau. Thứ hai, nó mở rộng khoảng cách tối đa giữa bất cứ cặp nút nào trên LAN. Ví dụ, với 10BaseT khoảng cách lớn nhất giữa nút và hub là 100m; vì thế trong LAN segment, khoảng cách lớn nhất giữa hai nút lên tới 200m.



Hình 5.31 Kết nối qua hub

Nếu kết nối qua hub, khoảng cách tối đa này có thể được mở rộng vì khoảng cách giữa các hub kết nối trực tiếp với nhau có thể là 100m khi sử dụng cáp xoắn đôi (và khoảng cách này sẽ tăng khi dùng cáp quang). Ưu điểm thứ ba là thiết kế nhiều tầng giảm nguy cơ sụp đổ của toàn hệ thống. Giả sử nếu bất kỳ hub của khoa nào đó bị trục trặc, hub trục chính có thể phát hiện vấn đề và phong tỏa kết nối tới hub khoa đó, như vậy các khoa còn lại vẫn có thể tiếp tục hoạt động và truyền thông trong khi hub bị lỗi không hoạt động.

Tuy vậy hub cũng có nhược điểm. Đầu tiên và có lẽ quan trọng nhất là khi sử dụng hub trung tâm, miền xung đột của mạng cục bộ của từng khoa trở thành miền xung đột chung của toàn bộ hệ thống. Xét ví dụ minh họa trên Hình 5.31. Trước khi kết nối ba khoa, mạng cục bộ mỗi khoa có băng thông cực đại là 10Mbps, vì vậy thông lượng toàn bộ tối đa của 3 LAN là 30Mbps. Nhưng khi mạng LAN của ba khoa được kết nối vào hub trung tâm, tất cả máy tính của ba khoa thuộc về cùng một miền xung đột, và thông lượng bị giảm xuống 10Mbps.

Hạn chế thứ hai là nếu các khoa khác nhau sử dụng các công nghệ Ethernet khác nhau thì không có khả năng để kết nối chúng vào hub trung tâm. Ví dụ, nếu một khoa sử dụng 10BaseT và các khoa còn lại sử dụng 100BaseT, thì không thể kết nối chúng với nhau vì hub về bản chất là repeater.

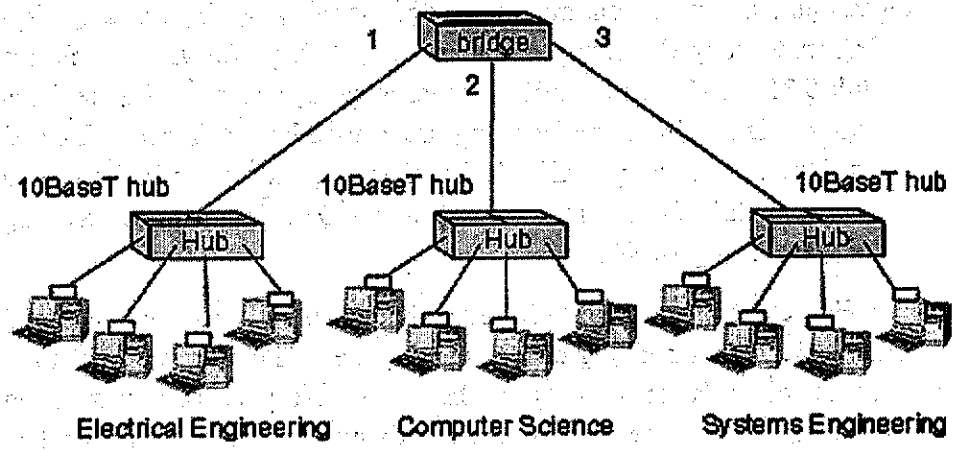
Hạn chế thứ ba là mỗi công nghệ Ethernet (10Base2, 10BaseT, 100BaseT, ...) có giới hạn về số nút, khoảng cách tối đa giữa hai máy tính trong miền xung đột và số tầng tối đa trong thiết kế nhiều tầng. Những hạn chế này hạn chế tổng số máy tính có thể kết nối đến mạng cục bộ cũng như phạm vi địa lí của mạng cục bộ nhiều tầng.

5.6.2 Bridge

Khác với hub (là thiết bị tầng vật lý), bridge có thể xử lý trên frame Ethernet, vì vậy nó là thiết bị tầng 2. Thực tế, bridge chính là thiết bị chuyển mạch gói thực hiện việc chuyển và lọc frame căn cứ trên địa chỉ vật lý. Khi

frame đến từ một cổng nào đó của bridge, bridge không gửi frame đến tất cả các cổng khác. Bridge sẽ xác định địa chỉ đích tầng 2 (địa chỉ vật lý) của frame và chuyển frame đến cổng duy nhất dẫn về đích.

Hình 5.32 minh họa ba khoa trong ví dụ trước kết nối tới bridge. Ba chữ số bên cạnh bridge là số thứ tự các cổng của bridge. Khi các khoa được kết nối qua bridge như trong hình 5.27 chúng ta vẫn coi mạng kết nối toàn bộ là LAN và mạng của mỗi khoa là LAN segment giống như ở phần trước. Nhưng khác với thiết kế hub nhiều tầng trong hình 5.26, mỗi LAN segment bây giờ là một miền xung đột đã được cô lập.



Hình 5.32 Kết nối bằng Bridge

Bridge có thể khắc phục nhiều vấn đề của hub. Thứ nhất, bridge cho phép truyền thông giữa các khoa trong khi cô lập miền xung đột của mỗi khoa. Thứ hai, bridge có thể kết nối các công nghệ LAN khác nhau (Ethernet 10Mbps và 100Mbps chẳng hạn). Thứ ba, không bị giới hạn về khoảng cách tối đa trong mạng cục bộ khi sử dụng bridge để kết nối các LAN segment. Về lý thuyết mà nói sử dụng bridge có thể xây dựng một mạng LAN trải rộng trên toàn thế giới.

Bridge Fowarding và Filtering (chuyển tiếp và lọc)

Lọc (Filtering) là khả năng xác định liệu sẽ chuyển tiếp frame đến cổng nào đó hay loại bỏ luôn frame. **Chuyển tiếp (Fowarding)** là khả năng xác định cổng kế tiếp để chuyển frame đi. Bridge thực hiện hai chức năng

này nhờ **bảng bridge (bridge table)**. Mỗi hàng trong bảng ứng với một nút đích trên mạng LAN. Tuy vậy bảng bridge không nhất thiết phải chứa tất cả các hàng cho mọi nút trong mạng. Mỗi hàng trong bảng bridge gồm có (1) địa chỉ vật lý của nút, (2) cổng bridge có thể dẫn đến nút đó, (3) thời điểm thiết lập hàng đó trong bảng. Ví dụ về bảng bridge cho LAN trong Hình 5.32 được chỉ ra trong Hình 5.33. Mặc dù quá trình chuyển frame có vẻ tương tự quá trình chuyển gói dữ liệu datagram trong chương 4, nhưng chúng ta sẽ thấy ngay chúng hoàn toàn khác nhau. Ở đây, chú ý rằng địa chỉ bridge sử dụng là địa chỉ vật lý chứ không phải là địa chỉ của tầng mạng (IP). Chúng ta cũng thấy ngay bảng bridge được xây dựng khác với bảng định tuyến.

Address	Interface	Time
62-FE-F7-11-89-A3	1	9:32
7C-BA-B2-B4-91-10	3	9:36
....	....	....

Hình 5.33 Một phần bảng bridge cho LAN trong hình 5.26

Để hiểu chức năng lọc và chuyển tiếp làm việc như thế nào, giả sử frame với địa chỉ đích DD-DD-DD-DD-DD-DD đến bridge từ cổng x. Bridge tìm kiếm trên bảng lọc hàng ứng với địa chỉ vật lý DD-DD-DD-DD-DD-DD để tìm ra cổng y tương ứng - là cổng sẽ dẫn đến nút có địa chỉ đích DD-DD-DD-DD-DD-DD. Chúng ta sẽ thấy điều gì sẽ xảy ra nếu không có giao diện y như thế trong bảng:

- Nếu x = y, thì frame đến từ segment chứa adapter DD-DD-DD-DD-DD-DD. Không cần chuyển frame đến bất kỳ cổng nào khác, bridge thực hiện chức năng lọc bằng cách loại bỏ frame.

- Nếu x ≠ y thì frame cần được gửi đến segment nào đó qua cổng y. Bridge thực hiện chức năng chuyển tiếp bằng cách đặt frame vào bộ đệm ra của cổng y.

Những quy tắc đơn giản này cho phép bridge cô lập các miền xung đột của các LAN segment khác nhau. Những quy tắc này cũng cho phép hai cặp thiết bị trên hai segment khác nhau truyền đồng thời mà không bị xung đột.

Xét những qui tắc này cho mạng minh họa trên Hình 5.32 và bảng bridge tương ứng trên Hình 5.33. Giả sử frame với địa chỉ đích 62-EF-F7-11-89-A3 được gửi đến bridge qua cổng 1. Bridge kiểm tra bảng và thấy rằng đích nằm trên LAN segment được kết nối đến cổng 1 (là mạng LAN của khoa Electrical Engineering). Điều này có nghĩa là frame thực sự đã được quảng bá trên LAN segment này. Do vậy bridge sẽ lọc frame (nghĩa là, loại bỏ frame – vì thực sự máy tính đích cũng đã nhận được frame này rồi). Giả sử frame với địa chỉ đích như vậy đến từ cổng 2. Bridge lại kiểm tra bảng và thấy rằng đích nằm ở trên hướng ứng với cổng 1, do đó bridge chuyển frame ra cổng 1. Rõ ràng rằng nếu bảng bridge đầy đủ và chính xác, bridge cho phép truyền thông giữa các khoa nhưng cô lập các miền xung đột.

Khi có frame để gửi chuyển tiếp, hub (hoặc repeater) gửi frame lên trên đường truyền mà không quan tâm xem có thiết bị nào khác đang chiếm dụng đường truyền không. Trái lại, bridge sẽ sử dụng thuật toán CSMA/CD trong phần 5.3 khi cần gửi đi frame. Tức là bridge sẽ không truyền ngay nếu như có nút khác trên LAN segment cũng đang truyền; hơn nữa, bridge cũng sử dụng thuật toán exponential backoff khi có xung đột. Vì vậy, cổng của bridge hoạt động giống như adapter của nút. Nhưng về mặt kỹ thuật mà nói, bridge không phải là adapter vì chúng không có địa chỉ vật lý. Chú ý rằng adapter của nút luôn luôn chèn địa chỉ vật lý của nó vào trường địa chỉ nguồn trong tất cả các frame nó gửi đi. Điều này cũng đúng cho adapter của router. Ngược lại bridge không thay đổi địa chỉ nguồn của frame.

Một tính năng quan trọng của bridge là khả năng kết nối các LAN segment sử dụng những công nghệ Ethernet khác nhau. Ví dụ nếu trong Hình 5.32, khoa Electrical Engineering sử dụng Ethernet 10BaseT, Khoa Computer Science sử dụng Ethernet 100BaseT và khoa System Engineering sử dụng Ethernet 10BaseT thì bridge có thể kết nối cả 3 segment trên. Với bridge Ethernet Gigabit có thể sử dụng đường truyền 1 Gbps nối tới router. Như chúng ta đã đề cập ban đầu, hub không có tính năng có thể kết nối các công nghệ với tốc độ truyền khác nhau.

Khi sử dụng bridge làm thiết bị kết nối, thì về lý thuyết LAN không bị giới hạn bởi phạm vi địa lý. Trên lý thuyết chúng ta có thể xây dựng mạng LAN trải rộng toàn cầu bằng kết nối các hub qua bridge. Theo thiết kế

này, mỗi hub là một miền xung đột và do đó LAN không bị giới hạn. Tuy nhiên sau đây chúng ta sẽ thấy rằng trong mạng lớn người ta sẽ kết nối qua router, chứ không sử dụng bridge.

Tự học (Self-Learning)

Đặc tính tuyệt vời (nhất là đối với những người quản trị mạng) của bridge là khả năng tự học. Bảng lọc của bridge được xây dựng tự động mà không cần bất cứ sự can thiệp nào từ phía người quản trị. Nói cách khác, bridge có khả năng tự học. Khả năng này được thực hiện như sau:

Bảng bridge khởi đầu là rỗng.

Khi frame đến cổng nào đó và địa chỉ đích của frame không có trong bảng, thì bridge sẽ chuyển frame đến bộ đệm ra của tất cả các cổng còn lại (tại mỗi cổng, frame được truyền lên LAN segment nhờ CSMA/CD)

Khi nhận được frame, bridge lưu trữ: (1) địa chỉ vật lý trong trường địa chỉ nguồn của frame, (2) cổng nhận được frame, (3) thời gian hiện tại. Như vậy bridge ghi nhớ được vị trí LAN segment của nút gửi. Nếu nút nào đó trong LAN gửi frame qua bridge thì bridge sẽ xác định được cổng để đi đến nút đó.

Khi địa chỉ đích của frame có trong bảng, thì bridge chuyển frame đến cổng thích hợp.

Bridge sẽ xoá địa chỉ trong bảng nếu adapter có địa chỉ đó không tiếp tục gửi frame trong khoảng thời gian xác định. Theo cách này, nếu PC được thay thế bởi PC khác (với adapter khác) thì địa chỉ vật lý của PC trước sẽ bị bridge xoá.

Xét quá trình tự học của bridge trong Hình 5.32 và bảng bridge tương ứng trong Hình 5.33. Giả sử rằng tại thời điểm 9:39 bridge nhận được một frame có địa chỉ gửi là 01-12-23-45-56 đến cổng 2. Giả sử, địa chỉ này chưa có trong bảng, bridge sẽ bổ sung một hàng mới trong bảng như chỉ ra trên Hình 5.34.

- Tiếp tục với ví dụ này, giả sử rằng "tuổi thọ" của mỗi hàng trong bảng là 60 phút và máy tính với địa chỉ 62-FE-F7-11-89-A3 không gửi đi

bất kỳ frame nào qua bridge trong khoảng thời gian từ 9:32 đến 10:32 thì lúc 10:32, bridge sẽ xóa địa chỉ này khỏi bảng.

Bridge là thiết bị theo kiểu **"cắm vào là chạy" (plug and play)** bởi vì nó không cần sự can thiệp của người quản trị mạng. Người quản trị mạng chỉ cần cắm connector vào các cổng của bridge. Người quản trị mạng không cần thiết lập cấu hình cho bảng bridge trong thời gian cài đặt hay khi máy tính tách khỏi LAN segment. Do đặc tính này, bridge được xem là trong suốt (transparent).

Địa chỉ MAC	Interface	Time
01-12-23-34-45-56	2	9:39
62-FE-F7-11-89-A3	1	9:32
7C-BA-B2-B4-91-10	3	9:36
....	....	....

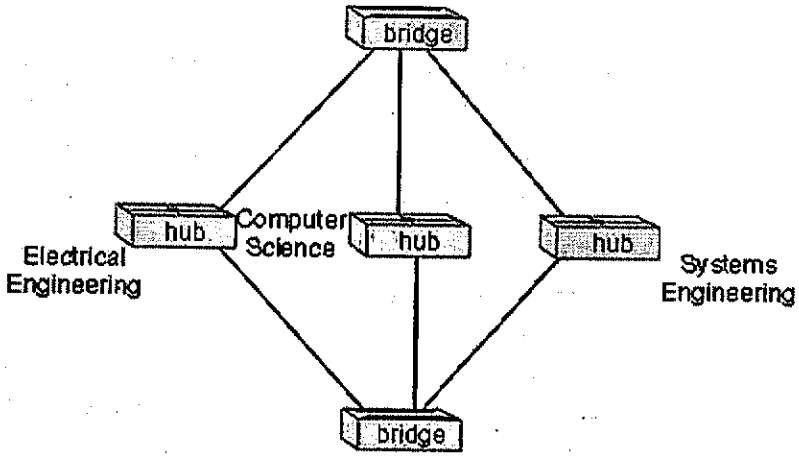
Hình 5.34 Bảng lọc của bridge

Spanning tree

Nếu hoàn toàn kết nối LAN segment theo kiểu phân cấp thì khi hub hoặc bridge gần đỉnh bị hỏng, thì một phần lớn LAN sẽ không được kết nối. Chính vì lý do này người ta thường xây dựng mạng với nhiều đường nối giữa các LAN segment. Một ví dụ về mạng như thế được minh hoạ trong Hình 5.35.

Nhiều đường dư thừa giữa các LAN segment làm giảm khả năng sụp đổ của toàn bộ hệ thống. Nhưng có nhiều đường dẫn giữa các segment cũng sẽ phát sinh ra nhiều vấn đề - một frames có thể di chuyển vòng quanh hay được nhân bản lên nhiều lần trong mạng cục bộ. Để hình dung ra điều này, giả sử bảng bridge trong hình 5.30 rỗng và máy tính trong khoa Electrical Engineering gửi frame đến máy tính trong khoa Computer Science. Khi frame đến hub Electrical Engineering, hub sẽ sinh ra hai bản sao của frame và gửi mỗi bản đến hai bridge. Khi mỗi bridge nhận được frame, nó sẽ tạo ra 2 bản sao của frame, một bản gửi đến hub của khoa System Engineering và

bản kia đến hub của khoa Computer Science. Vì cả hai bridge cùng làm như vậy, sẽ có 4 frame giống hệt nhau trong LAN. Frame có thể được nhân bản liên tục nếu bridge không biết nút nhận nằm ở đâu (chú ý rằng để địa chỉ vật lý của máy tính nhận có trong bảng lọc, thì trước đó máy tính nhận phải gửi đi một frame qua bridge). Trong trường hợp này, số bản sao của frame gốc tăng theo hàm số mũ, làm tràn ngập toàn bộ mạng.



Hình 5.35 Kết nối dư thừa

Để ngăn ngừa những tình huống nêu trên, bridge sử dụng giao thức **spanning tree**. Trong giao thức spanning tree, bridge truyền thông với bridge khác trên LAN để xác định spanning tree, nghĩa là, một tập con của topo 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 topo ban đầu. Ví dụ trong hình 5.30, spanning tree được hình thành bằng cách bridge phía trên phong tỏa kết nối cổng kết nối đến Eletrical Engineering và bridge phía dưới phong tỏa cổng kết nối đến System Engineering. Với các cổng bị phong tỏa và loại bỏ được các vòng lặp, frame sẽ không lặp và nhân bản. Nếu khi nào đó một liên kết trong spanning tree bị lỗi, bridge có thể kết nối lại giao diện đã bị phong tỏa, kích hoạt thuật toán spanning tree lần nữa và xác định spanning tree mới.

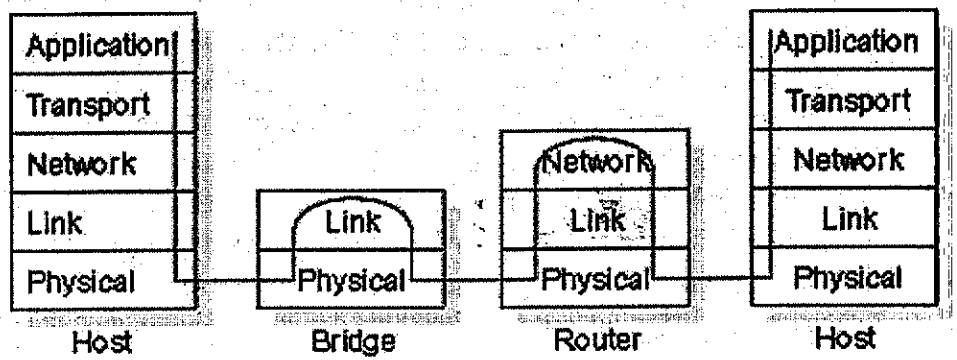
Phân biệt Bridge và Router

Như đã trình bày trong chương 4, router là thiết bị chuyển mạch gói theo kiểu store-and-forward và chuyển gói tin theo địa chỉ tầng mạng. Mặc

dù cũng là thiết bị chuyển mạch kiểu store-and-forward, nhưng điểm khác biệt cơ bản giữa bridge với router là bridge sử dụng địa chỉ vật lý. Như vậy router là thiết bị chuyển mạch gói ở tầng 3 trong khi bridge là thiết bị chuyển mạch gói ở tầng 2.

Người quản trị mạng sẽ phải lựa chọn giữa bridge và router khi cài đặt thiết bị kết nối. Ví dụ, với hệ thống mạng trong Hình 5.32, người quản trị mạng có thể lựa chọn router thay vì lựa chọn bridge. Thật vậy, router cũng sẽ cô lập ba miền xung đột trong khi vẫn cho phép truyền thông giữa các khoa. Như vậy cả bridge và router đều có thể làm thiết bị kết nối. Vậy ưu và nhược điểm giữa chúng là gì ?

Đầu tiên ta xét về bridge. Như đã đề cập trên, 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 lọc và chuyển gói dữ liệu cao – như minh họa trên Hình 5.35, bridge chỉ phải xử lý gói dữ liệu của tầng 2 trong khi router phải xử lý gói dữ liệu của tầng 3. Mặt khác, giao thức spanning tree hạn chế topo của toàn bộ mạng. Điều này có nghĩa là tất cả các frame chỉ được chuyển trên spanning tree, thậm chí khi có nhiều đường dẫn trực tiếp (nhưng bị phong tỏa) giữa nguồn và đích. Sự hạn chế của spanning tree cũng tập trung vào khả năng tải trên đường truyền spanning tree khi nó có thể đã được lan truyền đến tất cả các đường truyền khác của mạng cũ. Hơn nữa bridge không đưa ra bất cứ sự bảo vệ nào để chống lại sự phát ra hàng loạt - nếu máy tính bị lỗi và truyền đi một luồng frame Ethernet liên tục, bridge sẽ chuyển tất cả những frame này khiến toàn bộ mạng có thể bị sụp đổ.



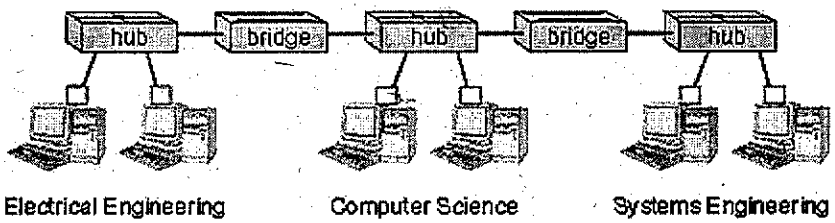
Hình 5.36 Luồng đi của gói dữ liệu trong máy tính, bridge và router

Bây giờ xét đến ưu và nhược điểm của router. Nói chung địa chỉ mạng thường phân cấp (không phẳng như địa chỉ vật lý), gói dữ liệu chắc chắn không quay vòng lại qua router ngay cả khi có nhiều đường đi (Thực sự gói dữ liệu có thể quay vòng nếu cấu hình bảng định tuyến của router bị đặt sai, nhưng như chúng ta đã học trong chương 4, IP sử dụng trường TTL trong tiêu đề gói dữ liệu để hạn chế chuyện này). Vì vậy, gói dữ liệu không bị giới hạn chuyển trong spanning tree, nó có thể sử dụng đường dẫn tốt nhất giữa nguồn và đích. Các router không bị giới hạn trong spanning tree nên Internet có thể có topo cực kỳ phong phú, ví dụ nhiều đường truyền giữa châu Âu và Bắc Mỹ. Một đặc tính quan trọng khác của router là tạo ra "firewall" (bức tường lửa) chống lại sự phát tán liên tục (quảng bá storm) ở tầng 2. Có lẽ yếu điểm duy nhất của router là không có khả năng "cắm vào là chạy"- cần cấu hình địa chỉ IP cho chúng và các máy tính kết nối đến chúng. Hơn nữa, thời gian xử lý gói tin của router thường lâu hơn bridge vì chúng phải xử lí các trường tiêu đề của tầng 3.

Với cả ưu và nhược điểm như đã thảo luận trên, vậy khi nào mạng sử dụng bridge, khi nào sử dụng router? Thông thường một mạng nhỏ gồm vài trăm máy tính nằm trên một số LAN segment chỉ cần sử dụng Bridge. Nhưng với những mạng lớn gồm hàng nghìn máy tính sẽ cần tới nhiều router bên trong mạng (bên cạnh bridge). Những router này cung cấp khả năng cô lập mạnh hơn, kiểm soát việc gửi tràn ngập.

Kết nối LAN segment qua các trục chính (backbone)

Lại xét ví dụ kết nối mạng Ethernet trong 3 khoa trên Hình 5.37 với bridge. Thiết kế này sử dụng hai bridge, mỗi bridge có hai cổng. Bridge thứ nhất kết nối hai khoa Electrical Engineering và Computer Sciense. Bridge kia kết nối khoa Computer Science với Systems Engineering. Mặc dù, bridge hai cổng rất phổ biến do giá rẻ và đơn giản, nhưng mô hình thiết kế trong Hình 5.37 không được ưa chuộng. Có hai lý do, thứ nhất, nếu hub của Computer Science bị hỏng thì máy tính ở hai khoa Electrical Engineering và Systems Engineering không thể trao đổi được với nhau. Thứ hai truyền thông giữa hai khoa Electrical Engineering và System Engineering phải thông qua Computer Science, dễ gây xung đột trong LAN segment của Computer Science.

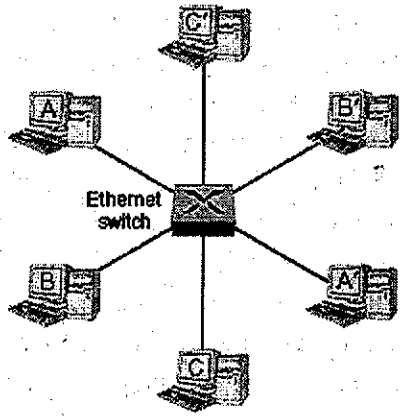


Hình 5.37 Kết nối không có backbone

Một nguyên tắc quan trọng định hướng dẫn việc kết nối các LAN segment khác là sử dụng đường trục chính (backbone) – là một mạng có kết nối trực tiếp đến tất cả các LAN segment. Khi LAN có một trục chính thì mỗi cặp LAN segment có thể truyền thông mà không cần thông qua LAN segment thứ ba. Trong thiết kế ở hình 5.37, bridge ba cổng đóng vai trò một backbone.

5.6.3. Switch

Cho đến giữa những năm 90 ba loại thiết bị kết nối mạng cục bộ được sử dụng chủ yếu là: hub (repeater), bridge, router. Gần đây, một thiết bị trở nên rất thông dụng là switch Ethernet. Switch Ethernet được hỗ trợ bởi ngành công nghiệp sản xuất thiết bị mạng. Về thực chất, switch là là bridge nhiều cổng có hiệu suất cao. Giống bridge, switch chuyển và lọc frame căn cứ vào địa chỉ vật lý đích, tự động xây dựng bảng lọc khi có frame đi qua. Điểm khác biệt quan trọng nhất giữa bridge và switch là bridge có ít cổng (từ 2 đến 4) trong khi switch có thể có nhiều cổng hơn.

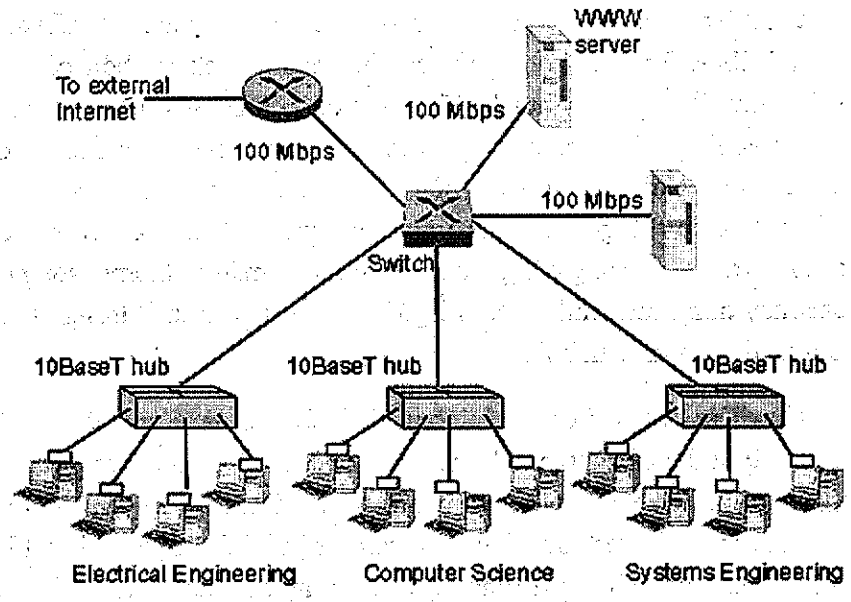


Hình 5.38 Switch Ethernet cung cấp truy cập Ethernet dành riêng đến 6 máy tính

Có thể mua switch có các cổng với tốc độ khác nhau 10 Mbps, 100 Mbps và 1Gbps. Ví dụ, một người có thể mua switch có bốn cổng 100 Mbps, hai mươi cổng 10 Mbps hoặc switch có bốn cổng 100 Mbps và một cổng 1 Gbps. Dĩ nhiên, nhiều cổng và tốc độ truyền của cổng càng cao thì switch càng đắt. Nhiều switch vận hành trong chế độ song công hoàn toàn, nghĩa là chúng có thể gửi và nhận frame tại cùng một thời điểm trên cùng một cổng. Với switch song công (cùng với bộ card mạng Ethernet song công trên các máy tính), máy tính A có thể gửi file đến máy tính B trong khi máy tính B gửi file đến máy tính A.

Ưu điểm của switch nhiều cổng là ở chỗ dễ dàng kết nối trực tiếp giữa các máy tính với switch. Khi có đường kết nối trực tiếp song công với switch, máy tính có thể truyền (và nhận) frame ở tốc độ truyền tối đa của adapter, đặc biệt adapter máy tính luôn cảm nhận kênh truyền rỗi và không bao giờ bị xung đột. Trong trường hợp này, máy tính được xem có **đường dùng riêng (dedicated link)**. Trên Hình 5.38 switch Ethernet cung cấp đường dùng riêng cho 6 máy tính. Các đường truy cập dùng riêng này cho phép đồng thời A gửi file đến A' trong khi B đang gửi file đến B' và C đang gửi đến C'. Nếu mỗi máy tính sử dụng adapter card 10 Mbps thì toàn bộ băng thông của hệ thống là 30 Mbps. Nếu A và A' có adapter 100 Mbps và những máy tính còn lại có adapter 10 Mbps, thì băng thông có thể lên tới 120 Mbps.

Hình 5.39 minh họa cách kết nối một trường đại học với nhiều khoa và một một số server quan trọng qua hub, switch và router. Trên Hình 5.39 mỗi khoa là một LAN segment sử dụng hub 10 Mbps. Vì mỗi hub có kết nối đến switch nên các khoa hoàn toàn có khả năng trao đổi dữ liệu với nhau. Server cho dịch vụ Web và email đều có đường dùng riêng 100 Mbps đến switch. Cuối cùng router sẽ kết nối toàn bộ hệ thống ra Internet, và router cũng có đường dùng riêng 100 Mbps đến switch. Chú ý switch này có ít nhất 3 cổng 10 Mbps và 3 cổng 100 Mbps.



Hình 5.39 Mạng cơ quan sử dụng phối hợp hub, switch và router

Chuyển mạch xuyên suốt (Cut – Through)

Ngoài việc có nhiều cổng, hỗ trợ nhiều môi trường và tốc độ truyền khác nhau, có chức năng quản trị mạng, các nhà sản xuất switch Ethernet thường quảng cáo sản phẩm của mình có khả năng gửi **xuyên suốt** (**cut-through**) chứ không phải kiểu store-and-forward như router và bridge. Khác biệt giữa store-and-forward và cut-through không lớn. Xét một gói dữ liệu chuyển qua thiết bị chuyển mạch (có thể là router, bridge hoặc switch). Gói tin đến switch từ một cổng nào đó và cần chuyển ra một cổng nào đó. Tại bộ đệm ở cổng ra của gói tin có thể có nhiều gói tin khác đang chờ chuyển. Khi đó store-and-forward và cut-through giống nhau. Hai công nghệ chuyển mạch này chỉ khác nhau khi bộ đệm cổng ra rỗng.

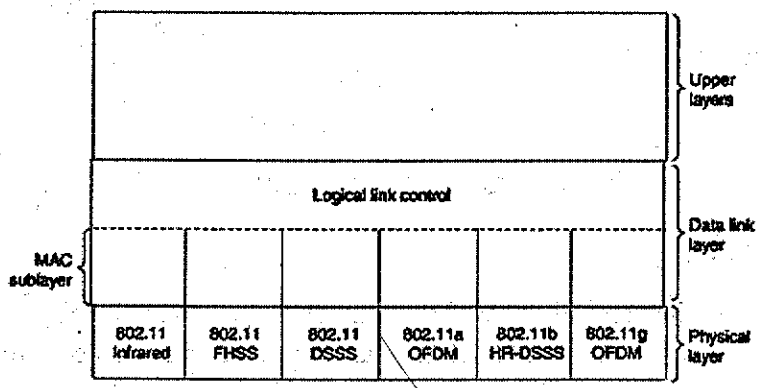
Khi gửi gói tin tới thiết bị chuyển kiểu store-and-forward, thiết bị sẽ thu và lưu trữ **toàn bộ** gói tin trước khi chuyển lên đường ra. Trong trường hợp bộ đệm ở cổng ra rỗng, phải thu thập toàn bộ gói tin rồi mới được chuyển đi (store and forward), việc này sẽ góp phần làm tăng thời gian trễ. Giới hạn độ trễ này là L/R, trong đó L là độ dài của gói tin và R là tốc độ truyền của cổng đến. Chú ý rằng gói tin chỉ chịu độ trễ này nếu bộ đệm cổng ra rỗng trước khi toàn bộ gói tin đến switch.

Với chuyển mạch kiểu cut-through, nếu buffer rỗng trước khi toàn bộ gói tin đến, switch có thể bắt đầu gửi đi phần trước trong khi đang nhận phần sau của gói tin. Tất nhiên trước khi truyền gói tin trên cổng ra, phải xác định được trường địa chỉ đích. (Thời gian trễ này không thể tránh khỏi đối với tất cả các loại chuyển mạch vì switch phải xác định cổng ra thích hợp). Tóm lại, trong chuyển mạch kiểu cut-through, gói tin không cần được "lưu trữ" đầy đủ trước khi chuyển tiếp đi mà gói tin sẽ được chuyển ngay khi cổng ra rỗi. Nếu cổng ra nối với mạng đa truy cập có chung môi trường truyền với những máy tính khác (ví dụ nối đến hub) thì switch phải "lắng nghe" để kiểm tra kênh truyền có rỗi không trước khi chuyển.

5.8. MẠNG LAN KHÔNG DÂY

5.8.1 Giới thiệu chung

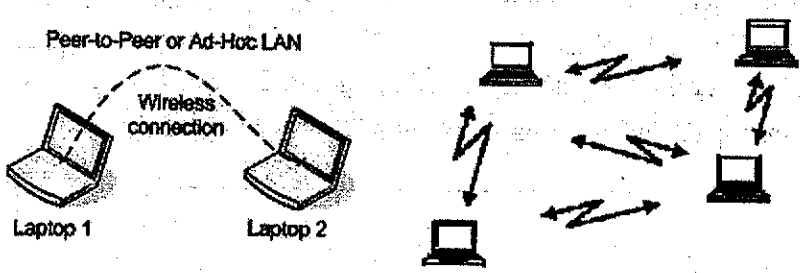
Các đặc tả về mạng LAN không dây ứng với tầng vật lý và tầng liên kết dữ liệu trong mô hình OSI. Thường tầng vật lý ở đây xác định việc sử dụng kênh truyền nào cũng như độ lớn của tín hiệu. Các chuẩn WLAN được IEEE đặc tả trong họ giao thức IEEE 802.11 chủ yếu sử dụng tia hồng ngoại hoặc dải phổ tự do. Dải phổ này còn được gọi là dải ISM (Industry, Science, Medicine) dùng trong các ứng dụng công nghiệp, khoa học và y tế.



Hình 5.40 Vị trí các tầng con trong mạng LAN không dây

Chuẩn 802.11 đưa ra năm 1997 xác định ba kỹ thuật truyền được sử dụng ở tầng vật lý. Kỹ thuật hồng ngoại (giống như trong các thiết bị điều khiển tivi từ xa). Hai kỹ thuật còn lại sử dụng sóng radio có bước sóng ngắn là công nghệ FHSS và DSSS. Băng tần hai công nghệ này là băng tần tự do (2.4GHz ISM). Tốc độ truyền theo những công nghệ này tương đối thấp (1-2 Mbps) và có phạm vi phủ sóng khả bé (để giảm thiểu xung đột). Đến năm 1999, hai kỹ thuật mới có tốc độ cao hơn được đưa ra là OFDM và HR-DSSS với tốc độ 54Mbps, 11Mbps tương ứng. Chú ý rằng các kỹ thuật này ứng với tầng Vật lý (Hình 5.40).

Nhiệm vụ của tầng liên kết dữ liệu ở đây là tổ chức việc truy cập đường truyền, đồng bộ hóa frame, kiểm soát tài nguyên. Cơ chế truy cập thực hiện chức năng điều phối phân tán (DCF – Distributed Coordination Function) được cài đặt ở tất cả các trạm (máy tính) có thiết bị thu phát không dây. Có hai kiểu thiết bị chính: Trạm di động (mobile station, chẳng hạn laptop) và Điểm truy cập (Access Point - AP, có nhiệm vụ chuyển tiếp dữ liệu giữa các trạm di động hay giữa trạm di động với mạng LAN cố định). Có hai kiểu kết nối của mạng LAN không dây: kiểu có cơ sở hạ tầng và kiểu không có cơ sở hạ tầng (adhoc).



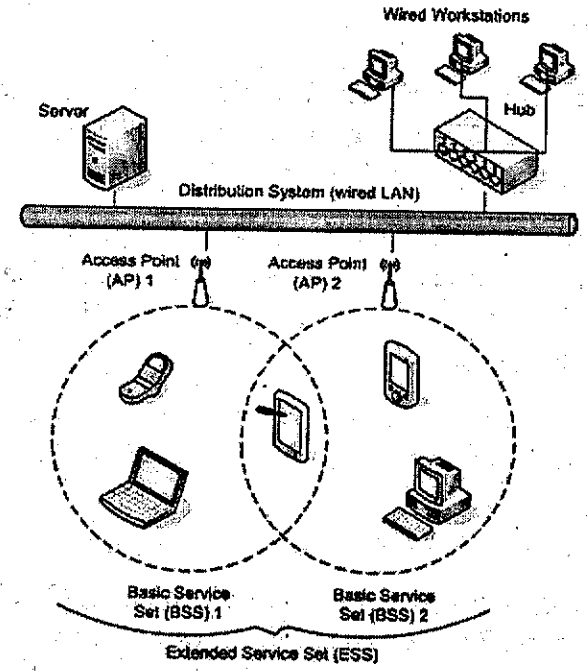
Hình 5.41 Mạng Adhoc LAN

Adhoc WLAN

Trong kiểu kết nối adhoc, hai trạm kết nối trực tiếp với nhau không qua AP (Hình 5.41).Trong kiểu kết nối này, các trạm làm việc có thể đóng vai trò server để chuyển tiếp dữ liệu. Hai trạm làm việc vẫn có thể trao đổi dữ liệu ngay cả khi chúng không nhận tín hiệu trực tiếp từ nhau.

WLAN có cơ sở hạ tầng

Trong kiểu kết nối này (Hình 5.42), hai trạm chỉ có thể trao đổi dữ liệu với nhau qua AP. Thông thường, AP được đính vào tường (trong một văn phòng) và có kết nối với mạng LAN cố định. Bên cạnh chức năng chính là trung chuyển các gói tin giữa các trạm và giữa trạm với mạng LAN có dây, AP có thể thực hiện chức năng điều phối điểm (point coordinating function - PCF) cho phép truyền thông giữa các nút dựa trên việc cấp phát tài nguyên tài nguyên (thời gian, băng thông...).



Hình 5.42 Mạng LAN có cấu trúc

Đa phần tầng MAC trong họ giao thức IEEE WLAN có cơ chế truy cập tương tự mạng Ethernet. Ethernet sử dụng đa truy cập sóng mang có phát hiện xung đột (CSMA/CD). Tuy nhiên đối với mạng không dây, việc phát hiện xung đột không hiệu quả do độ suy hao năng lượng của tín hiệu theo khoảng cách rất lớn. Thay vì sử dụng cơ chế phát hiện, đa phần các mạng không dây sử dụng cơ chế tránh xung đột (collision avoidance). Ý tưởng chính của cơ chế này là đảm bảo giữa hai lần truyền gói tin liên tiếp phải có một khoảng thời gian tối thiểu.

5.8.2 Lớp giao thức IEEE 802.11

Trong họ giao thức 802.11, tầng MAC xác định cách thức phân phối kênh truyền – tức là tại một thời điểm, nút nào được quyền truyền dữ liệu. Tầng phía trên của MAC – tầng LLC (Logical Link Control) có nhiệm vụ "che dấu" các đặc điểm khác nhau của tầng MAC với tầng mạng.

Tầng con Medium Access Control (MAC)

Chức năng chính của mạng LAN là chia sẻ tài nguyên giữa các máy tính trong một khu vực nhỏ, do vậy giống như Ethernet, nhiệm vụ của tầng MAC là xác định cụ thể từng giao diện của các máy tính trên mạng. Bên cạnh chức năng chính tổ chức truy cập kênh truyền và cơ chế điều phối đa truy cập, MAC còn có một số chức năng chính sau đây:

Cấu hình Mạng. Đối với mạng LAN có dây, cấu hình mạng chính là topo mạng và đã được xác lập bằng cách nối dây. Tuy nhiên đối với mạng không dây, đây lại là việc quan trọng vì phải ghép nối các máy tính thành các "nhóm" có thể định danh được.

Truy cập kênh truyền. Đối với mạng LAN, cơ chế truy cập môi trường truyền nhanh, tin cậy và công bằng hết sức cần thiết.

Đặc tả tầng MAC theo IEEE 802.11

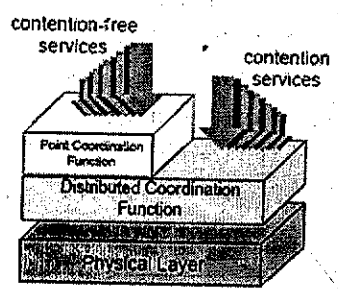
Chức năng truy cập kênh truyền của mạng LAN không dây khá tương đồng với mạng Ethernet. Đặc tả này có một vài chức năng để đảm bảo tính an ninh, và sẽ được bổ sung thêm một vài tính năng để sau này có thể tích hợp được với mạng điện thoại không dây.

Cấu hình mạng

Chuẩn IEEE 802.11 định nghĩa BSA (Basic Servive Area) là một khu vực có thể có nhiều máy tính (trạm) di động. Các máy tính trong BSA được gọi là Service Set (BSS). Các trạm làm việc trong BSS được kết nối với Access Point (AP). Nhiều BSS kết nối với nhau thành Distributed System (SS) để tạo thành Extended Service Set (ESS). Tất cả các trạm trong

cùng BSS sử dụng chung một tốc độ truyền (Basic Rate Set) và một cấu trúc gói dữ liệu PDU (xem minh họa Hình 5.42)

Truy cập kênh truyền trong IEEE 802.11



Hình 5.43 Cơ chế điều phối DCF

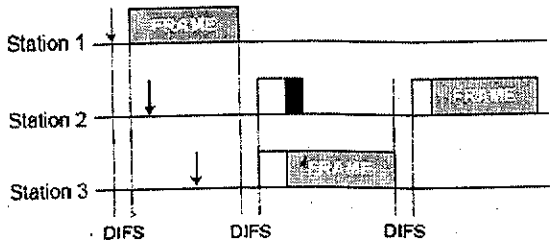
Chuẩn IEEE 802.11 định nghĩa cả tầng MAC lẫn tầng vật lý (Hình 5.40). Tầng MAC cung cấp hai dịch vụ: dịch vụ truy cập có tranh chấp (*Distributed Coordination Function - DCF*) và dịch vụ truy cập không tranh chấp (cài đặt ở *Point Coordination Function* - PCF). Những dịch vụ này được triển khai trên nền tầng vật lý, che dấu hết các đặc tính kỹ thuật của tầng vật lý đối với tầng mạng (Hình 5.43).

DCF là phương pháp chủ yếu trong họ IEEE 802.11 và sử dụng cơ chế *đa truy cập kênh truyền có tránh tắc nghẽn* (CSMA/CA). PCF được cài đặt trên nền DCF và dựa trên cơ chế hỏi vòng. Cơ chế này cho phép một trạm đóng vai trò trung tâm lần lượt "hỏi" các trạm trong mạng có muốn truyền dữ liệu không.

Distributed Coordination Function (DCF)

Trước khi truyền frame, trạm phải "lắng nghe" kênh truyền để xem có trạm khác chiếm dụng kênh truyền hay không. Nếu kênh truyền rỗi trong một khoảng thời gian lớn hơn Distributed InterFrame Space (DIFS) thì trạm được phép truyền (Hình 5.44). Nếu ngược lại (môi trường truyền bị chiếm dụng), việc truyền của trạm bị trì hoãn đến phiên truyền tiếp theo. Khoảng thời gian trì hoãn này (backoff time) được lựa chọn một cách ngẫu nhiên và được sử dụng để khởi tạo cho backoff timer. Giá trị bộ định thời này giảm khi kênh truyền được cảm nhận là rỗi, sẽ giữ nguyên trong trường hợp kênh truyền bận và sẽ được giảm về 0 (reset) khi kênh truyền rỗi trong khoảng

thời gian lớn hơn DIFS. Ví dụ, trên Hình 5.44, chúng ta thấy rằng backoff timer của trạm 2 bị dừng lại trong khi trạm 3 đang truyền. Giá trị timer này sẽ được đặt về 0 sau khi trạm 3 truyền xong một khoảng thời gian DIFS. Trạm 2 sẽ được phép truyền khi backoff timer của mình nhận giá trị 0. Giá trị backoff time cũng được chia khoảng. Cụ thể hơn, giá trị này là một số nguyên ngẫu nhiên nằm trong khoảng (0, CW-1), trong đó CW (*Contention Window*) là độ lớn cửa sổ backoff (Backoff Window). Trong lần thử truyền đầu tiên, *CW* = *CW*min và cứ sau mỗi lần truyền thử, giá trị này tăng gấp đôi cho đến khi đạt đến giá trị *CW*max.



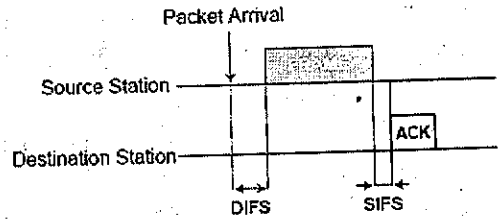
Hình 5.44 Cơ chế tránh tắc nghẽn

Theo chuẩn, giá trị của *CW*min và *CW*max phụ thuộc vào tầng vật lý phía dưới. Ví dụ, với công nghệ FHSS ở tầng Vật lý, giá trị của *CW*min và *CW*max tương ứng là 16 và 1024.

Rõ ràng hoàn toàn có khả năng tại cùng một thời điểm có nhiều trạm có nhu cầu truyền tin, khi đó xung đột chắc chắn xuất hiện. Với CSMA/CA trạm không có khả năng phát hiện xung đột bằng cách lắng nghe môi trường truyền giống như trong cơ chế CSMA/CD của mạng Ethernet. Do đó, để chắc chắn truyền thành công, phía nhận phải gửi phản hồi tích cực. Sau khi nhận đúng một frame, phía nhận đợi một khoảng thời gian Short InterFrame Space (SIFS) và gửi lại frame biên nhận ACK. Để ưu tiên việc gửi frame biên nhận hơn việc gửi frame dữ liệu bình thường khác, giá trị SIFS bé hơn DIFS (xem Hình 5.45). Nếu phía nhận không nhận được ACK, gói tin gửi đi sẽ bị coi là mất và sau một khoảng thời gian sẽ được gửi lại. Để phát hiện lỗi, người ta cũng sử dụng cơ chế CRC trình bày trong phần trước.

Sau khi phát hiện được frame bị lỗi (do xung đột hay lỗi trên môi trường truyền), trạm phải đợi ít nhất một khoảng thời gian bằng *Extended InterFrame Space* (EIFS) trước khi khởi động thuật toán backoff. Cụ thể, DCF phải sử dụng giá trị EIFS mỗi khi tầng vật lý báo cho tầng MAC frame

bắt đầu truyền không thể gửi thành công. Giá trị EIFS cũng được sử dụng để đồng bộ trạm làm việc với trạng thái của môi trường truyền (bận hay rỗi).



Hình 5.45 Gửi biên nhận khi nhận được frame dữ liệu

5.8.3 Một số vấn đề hay gặp đối với mạng không dây

Trong phần này, chúng ta nêu một số vấn đề nảy sinh trong mạng không dây, đây là những đặc điểm cơ bản khác với mạng có dây:

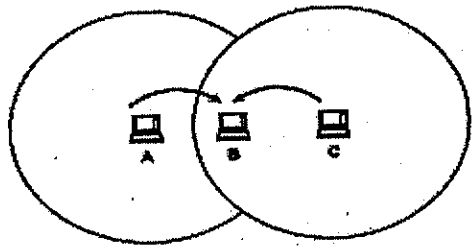
- Không có tiêu chí cụ thể để xác định phạm vi môi trường truyền. Có nghĩa là không thể xác định được phạm vi mà nút nằm trong đó chắc chắn nhận được frame, nằm ngoài phạm vi đó sẽ không nhận được.

- Môi trường truyền có thể bị nhiễu vì tín hiệu bên ngoài.

- Độ tin cậy kém môi trường có dây.

- Độ trễ trên môi trường truyền không cố định.

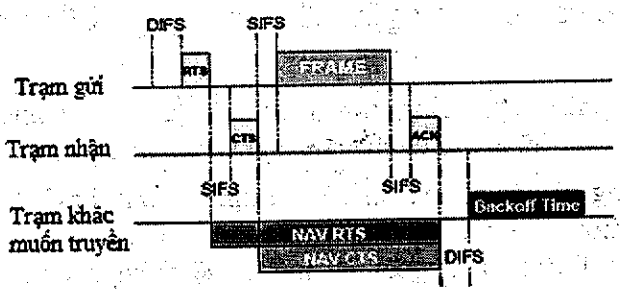
- Trong môi trường không dây, và cơ chế truy cập ngẫu nhiên dựa vào cảm nhận kênh truyền, những đặc tính nêu trên có thể gây nên những hiện tượng phức tạp, chẳng hạn "trạm ẩn" (hidden station) hay "trạm lộ" (exposed-station).



Hình 5.46 Vấn đề trạm ẩn

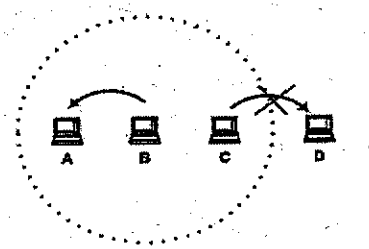
Hình 5.46 minh họa tình huống "trạm ẩn". Trạm B nằm giữa khoảng phủ sóng của A và C. Nhưng A và C không gửi trực tiếp dữ liệu được tới nhau. Giả sử trong lúc A bắt đầu gửi dữ liệu tới B thì C cũng có dữ liệu gửi tới B. Khi C thực hiện cảm nhận môi trường truyền thì tất nhiên không phát hiện dữ liệu do A gửi, vì vậy C coi kênh truyền rỗi và bắt đầu truyền. Tín hiệu từ A và C khi đến B sẽ bị xung đột

Vấn đề trạm ẩn này có thể khắc phục bằng cách bổ sung thêm vào cơ chế DCF cơ bản cơ chế cảm nhận ảo (*virtual carrier sensing*). Khi đó hệ thống có hai frame điều khiển: Request To Send (RTS) và Clear To Send (CTS). Trước khi truyền, trạm gửi sẽ gửi frame RTS tới trạm nhận để thông báo mình sẽ gửi frame dữ liệu (Hình 5.47). Khi nhận được frame RTS, trạm nhận sẽ gửi frame CTS. Chỉ khi nhận được CTS, bên gửi mới truyền đi dữ liệu.



Hình 5.47 Cơ chế báo gửi và báo nhận

Cả hai frame RTS và CTS đều được truyền trong những khoảng thời gian đủ để truyền đi một frame, nên tất cả các trạm trong vùng phủ sóng đều có thể nhận được các frame kiểu này và cập nhật giá trị *Network Allocation Vector* (NAV) tương ứng. Khi NVA dương, trạm không được gửi dữ liệu. Bằng cách sử dụng cơ chế RTS/CTS, trạm có thể nhận biết được sự truyền của các "trạm ẩn" cũng như khoảng thời gian truyền này.



Hình 5.48 Trạm lộ

Hình 5.48 minh họa tình huống "trạm lộ". Giả sử trạm A và C đều có thể nghe trạm B, nhưng trạm A không nghe được trạm C. Giả sử lúc B đang truyền cho A thì C nhận được một frame để truyền tới D. Rõ ràng C cảm nhận kênh truyền và thấy kênh truyền bận (do B truyền tới A), C sẽ trì hoãn việc truyền tới D, mặc dù việc này sẽ không gây xung đột tại A, điều này làm giảm thông lượng hệ thống.

5.9. PPP – GIAO THỨC ĐIỂM NỐI ĐIỂM

Phần trước chúng ta đã nói về kênh truyền quảng bá. Phần này chúng ta sẽ nghiên cứu giao thức liên kết dữ liệu cho kênh truyền điểm nối điểm (point-to-point) - 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, do đó PPP là một trong những giao thức tầng liên kết dữ liệu được sử dụng nhiều nhất nhất ngày nay. Giao thức quan trọng thứ hai là HDLC (High Level Data-Link Control [Spragins 1991]. Giao thức PPP được trình bầy tương đối đơn giản với mục đích khảo sát một số tính năng quan trọng nhất của lớp giao thức điểm nối điểm ở tầng liên kết dữ liệu.

Giao thức PPP [RFC 1661; RFC 2153] 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/SHD, kết nối X.25 hoặc mạch ISDN. Như đã nói trên, PPP chủ yếu được lựa chọn để kết nối máy tính gia đình đến ISP thông qua đường dây điện thoại. Trước khi đi sâu vào chi tiết của PPP, hãy điểm qua một số qui tắc chính mà IETF đã đặt ra cho mọi thiết kế của PPP [RFC 1547]:

Đóng gói gói tin (Framing): phía gửi trong giao thức PPP phải có khả năng lấy gói tin ở tầng mạng, đặt nó trong frame tầng liên kết dữ liệu. Phía nhận xác định được vị trí bắt đầu và kết thúc của frame cũng như vị trí gói tin tầng mạng trong frame.

Tính trong suốt: Giao thức PPP không được đặt ra bất kỳ hạn chế nào trên gói dữ liệu tầng mạng. Tức là nó có khả năng chuyển đi bất kỳ gói dữ liệu tầng mạng 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 lí tại cùng một thời điểm. Điều này cũng giống như giao thức IP có khả năng phân kênh cho nhiều giao thức giao vận khác nhau (ví dụ, TCP và UDP). Như vậy PPP cũng cần có một cơ chế để thực thể PPP phía nhận xác định được cần chuyển gói dữ liệu trong frame cho thực thể tầng mạng nào.

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 liệu có lỗi bit trong frame nhận được hay không.

Thời gian kết nối: PPP phải có khả năng phát hiện đường truyền bị lỗi ở mức link (ví dụ, không có khả năng để truyền dữ liệu từ phía gửi sang phía nhận) và phải thông báo tình trạng lỗi này cho tầng mạng.

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 hay đặt cấu hình địa chỉ tầng mạng cho nhau.

Đơn giản: Người ta đòi hỏi PPP đáp ứng nhiều yêu cầu ngoài những yêu cầu nêu trên. Một trong những yêu cầu quan trọng nhất là "tính đơn giản". Hiện nay hơn 50 RFC định nghĩa những khía cạnh "đơn giản" của giao thức này.

Tuy vậy các đặc tả trong thiết kế PPP không yêu cầu:

Sửa lỗi: PPP cần phát hiện được lỗi bit nhưng không cần phải sửa lỗi.

Kiểm soát lưu lượng: PPP phía nhận được hy vọng có khả năng nhận frame với tốc độ cao nhất của tầng vật lí phía dưới. Nếu tầng mạng không thể nhận với tốc độ này thì việc loại bỏ gói tin hay yêu cầu bên kia truyền chậm lại là trách nhiệm của các tầng cao hơn. Khi đó tầng cao hơn sẽ

yêu cầu thực thể tương đương phía bên kia giảm tốc độ tạo ra dữ liệu gửi cho PPP.

Đánh số thứ tự: PPP không được yêu cầu chuyển frame đến phía nhận theo đúng thứ tự gửi.

Đường truyền đa điểm: PPP vận hành trên những đường truyền với một phía gửi và một phía nhận duy nhất. Một số giao thức tầng liên kết dữ liệu khác (ví dụ, HDLC) cho phép nhiều nút nhận trên cùng một đường truyền (giống Ethernet).

5.9.1 Khuôn dạng gói dữ liệu (Frame PPP)

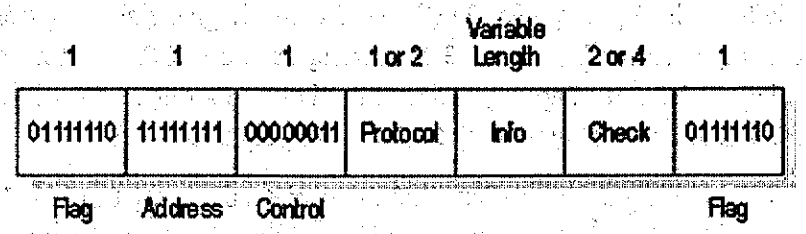
Hình 5.49 minh họa frame PPP giống frame HDLC [RFC 1662]

Frame PPP bao gồm những trường sau:

Trường cờ: Mọi frame PPP bắt đầu và kết thúc bằng một byte cờ có giá trị 01111110.

Trường địa chỉ: giá trị duy nhất của trường này là *11111111*.

Trường điều khiển: giá trị duy nhất của trường này là 00000011. Bởi vì cả hai trường địa chỉ và điều khiển đều mang những giá trị cố định, điều này giải thích vì sao những trường này được định nghĩa đầu tiên. Khuyến nghị PPP [RFC 1662] nói rõ rằng những giá trị này "có thể được định nghĩa sau này". Bởi vì những trường này mang giá trị cố định, PPP cho phép phía gửi không cần gửi byte địa chỉ và byte điều khiển, do đó tiết kiệm được hai byte tiêu đề trong frame PPP.



Hình 5.49 Khuôn dạng frame dữ liệu PPP

Trường giao thức (protocol): Trường giao thức cho PPP xác định giao thức tầng trên sẽ nhận dữ liệu trong frame PPP. Khi nhận được frame PPP, bên nhận sẽ kiểm tra xem frame có lỗi không và sau đó chuyển phần dữ liệu trong gói tin cho giao thức thích hợp. RFC 1700 định nghĩa mã 16 bit cho các giao thức được sử dụng cùng với PPP. Giao thức IP (dữ liệu trong frame PPP là gói dữ liệu IP datagram) 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. Giao thức IP Control được PPP sử dụng khi kích hoạt kênh truyền lần đầu tiên để cấu hình IP giữa các thiết bị trên hai đầu kênh truyền.

Thông tin: Trường này chứa gói tin được giao thức tầng mạng gửi đi trên đường truyền PPP (là IP datagram với giao thức IP). Độ dài lớn nhất của trường thông tin này là 1500 byte, mặc dù giá trị này có thể thay đổi lúc đặt cấu hình cho đường truyền.

Tổng kiểm tra (Checksum): Trường checksum được sử dụng để phát hiện các bit bị lỗi trong frame nhận được. Nó là mã CRC 2 hoặc 4 byte giống như trong giao thức HDLC.

Chèn byte (Byte stuffing)

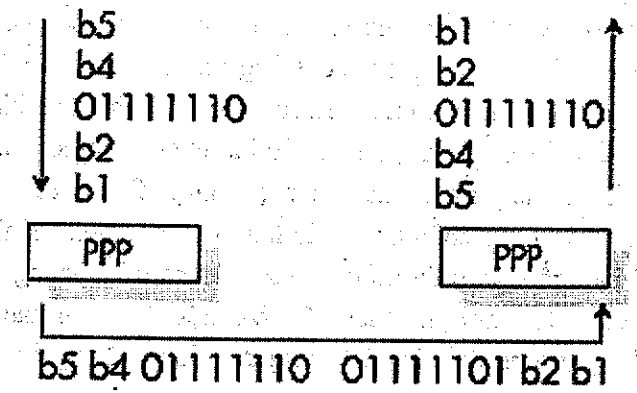
Trước khi kết thúc việc trình bầy về PPP frame, chúng ta hãy xét vấn đề phát sinh khi trong gói dữ liệu của giao thức tầng mạng lại có một byte giống byte cờ đánh dấu điểm bắt đầu và kết thúc của frame. Điều gì xảy ra nếu phát hiện thấy có byte cờ với giá trị 01111110 ở giữa gói tin trong trường thông tin. Bên nhận sẽ cho rằng đây là điểm kết thúc của frame PPP - mặc dù trên thực tế không phải như vậy.

Một cách để giải quyết vấn đề này là cấm các giao thức tầng trên gửi dữ liệu chứa byte cờ. Yêu cầu về tính trong suốt của PPP đã nêu ở trên không chấp nhận giải pháp này. Giải pháp thay thế - được PPP cũng như nhiều giao thức khác áp dụng là kỹ thuật **chèn byte (byte stuff)**.

PPP định nghĩa byte điều khiển đặc biệt có giá trị 01111101 làm nhiệm vụ đánh dấu. Nếu byte cờ - 01111110 - xuất hiện trong frame (trừ vị trí mở đầu và kết thúc của frame), PPP đặt byte đánh dấu trước byte cờ. Như vậy nó đã "chèn" thêm một byte điều khiển để đánh dấu rằng byte 01111110

không phải là cờ mà là dữ liệu thực. Bên nhận thấy 01111110 đứng trước 01111101 nên biết được 01111101 không phải là cờ mà là dữ liệu, nên nó sẽ tự động loại bỏ byte đánh dấu 01111110 được phía nhận chèn vào bên cạnh dòng dữ liệu thực.

Hình 5.50 minh hoạ chèn byte trong PPP. Tương tự như thế, nếu chính byte đánh dấu cũng xuất hiện trong dòng dữ liệu thực sự thì nó cũng cần được đánh dấu.



Hình 5.50 Chèn byte trong byte

5.9.2 Giao thức điều khiển đường truyền PPP (LCP) và kiểm soát mạng

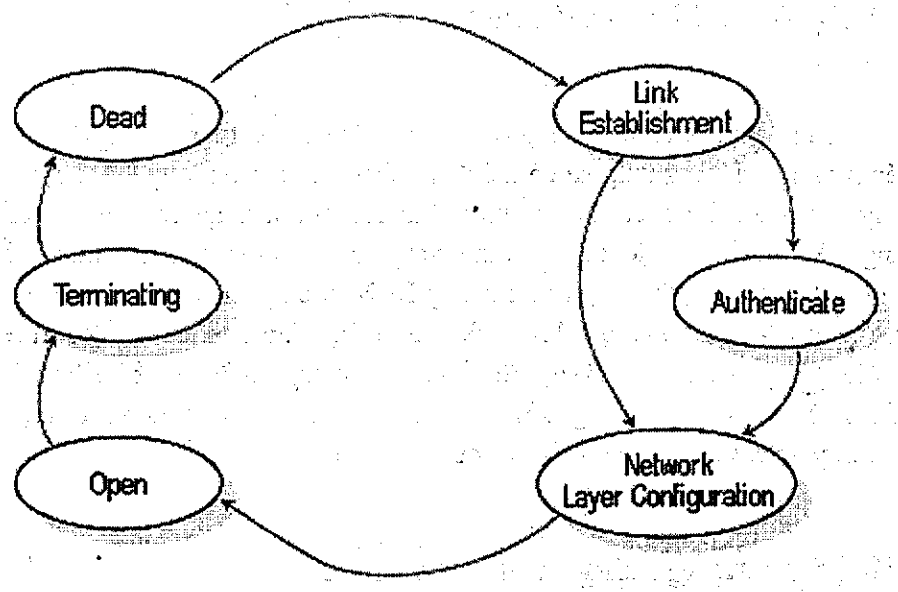
Trong phần trước, chúng ta đã thấy PPP đóng khung dữ liệu được gửi đi trên đường truyền. Nhưng đường truyền được khởi tạo như thế nào khi máy tính hoặc router ở một phía của đường truyền bật trước? Quá trình khởi tạo, duy trì, báo lỗi và đóng đường truyền PPP được thực hiện nhờ giao thức điều khiển đường truyền (LCP –Link Control Protocol) và các giao thức điều khiển mạng của PPP.

Trước khi trao đổi bất kì dữ liệu nào trên đường truyền PPP giao thức (mỗi phía ở một đầu đường truyền) phải thực hiện nhiều công việc nặng tự đặt trọng để thiết lập cấu hình cho đường truyền, điều này cũng ng truyền PPP.

thực thể TCP bên gửi và bên nhận thực hiện bắt tay ba bước (xem phần 3.5) để đặt các tham số của kết nối TCP trước khi trao đổi TCP segment. Hình 5.51 minh hoạ biểu đồ chuyển trạng thái của giao thức LCP để đặt cấu hình, duy trì và kết thúc đường truyền PPP.

Đường truyền PPP bắt đầu và kết thúc trong trạng thái đóng (dead). Khi phát sinh sự kiện như phát hiện sóng mang hay người quản trị mạng tác động để chỉ tầng vật lí sẵn sàng sử dụng, PPP bước sang trạng thái thiết lập đường truyền (link establishment). Trong trạng thái này, một phía của đường truyền gửi tùy chọn cấu hình mong muốn qua frame yêu cầu cấu hình LCP (là frame PPP có giá trị trường protocol ứng với giao thức LCP và trường information chứa nội dung cấu hình yêu cầu). Sau đó phía bên kia trả lời với frame ***configure-ack*** (chấp nhận tất cả các lựa chọn), frame ***configure-nak*** (hiểu nhưng không chấp nhận các lựa chọn) hoặc frame ***configure-reject*** (không thể ghi nhận hoặc chấp nhận các lựa chọn để đàm phán). Tùy chọn cấu hình LCP bao gồm kích thước tối đa của frame trên đường truyền, giao thức kiểm chứng được sử dụng (nếu cần) và một tùy chọn xác định có bỏ qua việc sử dụng trường địa chỉ và trường điều khiển trong frame PPP hay không.

Sau khi đường truyền được thiết lập, thỏa thuận xong các tuỳ chọn của đường truyền và kiểm chứng thành công, hai phía của đường truyền PPP sẽ trao đổi các gói tin kiểm soát của tầng mạng. Nếu IP chạy phía trên PPP, giao thức điều khiển IP [RFC 1332] được sử dụng để thiết lập cấu hình cho module giao thức IP tại mỗi đầu của đường truyền PPP. Gói tin IPCP được đặt trong frame PPP. IPCP cho phép hai module 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]. Sau khi cấu hình xong tầng mạng, PPP có thể bắt đầu gửi gói tin của tầng mạng - đường truyền ở trạng thái mở và dữ liệu bắt đầu chuyển trên đường truyền PPP. Các frame yêu cầu phản hồi và frame trả lời phản hồi LCP có thể được hai phía của đường truyền trao đổi để kiểm tra trạng thái đường truyền.



Hình 5.51 Sơ đồ chuyển trạng thái của LCP

Đường truyền PPP được duy trì cho đến khi gói tin LCP yêu cầu kết thúc được gửi đi. Nếu frame LCP yêu cầu kết thúc (terminate-request) từ một phía kết nối được trả lời bởi frame LCP chấp nhận kết thúc (terminate-ack) từ phía bên kia thì đường truyền bước vào trạng thái đóng.

Tóm lại, PPP là giao thức tầng liên kết dữ liệu cho hai thiết bị ở hai đầu của một đường truyền kiểu point-to-point, trao đổi các frame chứa gói dữ liệu của tầng mạng. Những chức năng chủ yếu của PPP là:

Đóng gói dữ liệu: Phương thức đặt gói dữ liệu trong frame PPP; xác định vị trí bắt đầu và kết thúc của frame; và phát hiện lỗi trong frame.

Giao thức điều khiển đường truyền: khởi tạo, duy trì và kết thúc đường truyền PPP.

Giao thức điều khiển mạng: một nhóm giao thức, mỗi giao thức ứng với một giao thức mạng ở tầng trên, cho phép module tầng mạng tự đặt cấu hình trước khi gói dữ liệu tầng mạng bắt đầu chuyển qua đường truyền PPP.

5.10 MẠNG RIÊNG ẢO (VPN)

Mục đích của mạng riêng ảo (Virtual Private Network - VPN) là cho phép một người dùng ở ngoài có thể đăng nhập được vào mạng LAN cục bộ thông qua mạng Internet như thể máy tính của người dùng đang nằm trong mạng LAN. Khi đó, mặc dù người dùng ở phía ngoài mạng LAN nhưng vẫn có thể sử dụng các dịch vụ trong mạng LAN không khác gì các máy tính đang nằm bên trong mạng LAN. Một trường hợp khác là tổ chức có nhiều địa điểm khác nhau, và mỗi địa điểm có một mạng LAN cục bộ riêng. Tổ chức muốn kết nối hai mạng LAN này lại với nhau để tạo thành một mạng LAN duy nhất. Công nghệ VPN có thể đáp ứng điều này.

5.10.1 Các mạng riêng ảo truyền thống

Phần này trình bày một vài kiến trúc VPN ở tầng Liên kết Dữ liệu và tầng Mạng.

VPN ở tầng Liên kết dữ liệu (tầng 2)

Trong thời kỳ đầu, khách hàng sẽ cài đặt hệ thống VPN bằng cách thuê bao các đường truyền riêng từ nhà cung cấp dịch vụ để kết nối hệ thống mạng cục bộ nằm trên nhiều địa điểm khác nhau.

Từ khi ra đời vào năm 1990, Frame Relay là công nghệ VPN "thống trị". Frame Relay cho phép nhà cung cấp dịch vụ tuy vẫn sử dụng đường truyền chia sẻ bình thường nhưng có thể cung cấp một băng thông thỏa thuận trước với khách hàng. Điều này được thực hiện bằng cách cấp phát "mạch ảo" trên kênh truyền dùng chung cho mỗi khách hàng (chứ không phải bằng cách phân chia kênh truyền). Mạch ảo ở đây được gọi là PVC (Permanent Virtual Circuit). Bằng cách cấu hình PVC, định danh kết nối ở tầng liên kết dữ liệu (Data-Link Connection Identifiers - DLCI) được thiết lập qua nhiều thiết bị khác nhau. Điều này sẽ tạo ra một "đường ống" cho phép dữ liệu của người sử dụng được truyền trên một đường dùng riêng cố định được cấp phát từ trước qua mạng dùng chung của nhà cung cấp dịch vụ.

Dịch vụ của nhà cung cấp dịch vụ chỉ cung cấp các kết nối ở tầng 2 chứ không liên quan đến tầng 3 (do đó gọi là VPN ở tầng 2). Ưu điểm của kiến trúc này là cho phép khách hàng độc lập trong việc thiết kế, định tuyến, đánh địa chỉ ở tầng 3.

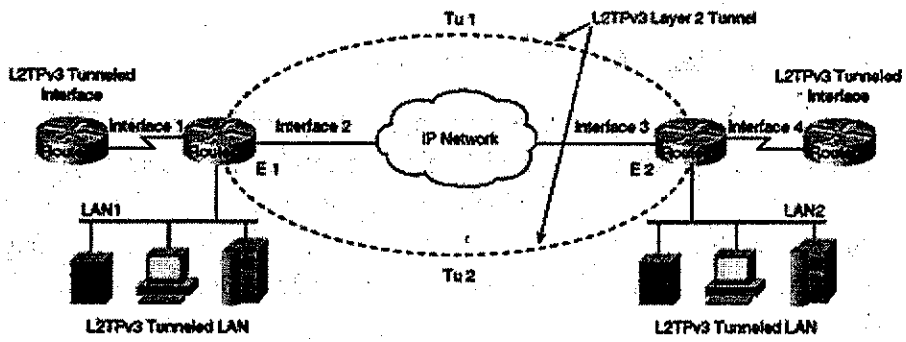
Sự độc lập của Fram Relay với giao thức mạng ở tầng 3 là yếu tố quan trọng để lựa chọn FR là công nghệ kết nối các mạng LAN từ nhiều địa điểm khác nhau. Hiện nay, nhiều nhà cung cấp dịch vụ hỗ trợ VPN trên nền ATM hoặc lai ghép cả ATM lẫn Frame Relay.

VPN ở tầng 3

Hiện nay công nghệ VPN chủ đạo ở tầng 3 là IP Security (IPsec) và MPLS Border Gateway Protocol (BGP). Những công nghệ này cho phép ứng dụng truy cập vào mạng Internet, mạng nội bộ hay mạng externet qua các đường kết nối an toàn.

Trong công nghệ VPN ở tầng 3, nhà cung cấp dịch vụ cho thuê bao các đường dùng riêng hay kết nối PVC giữa khách hàng và điểm truy cập dịch vụ (Point Of Presence – POP) gần nhất vào mạng của nhà cung cấp.

Hình 5.52 minh họa mô hình MPLS VPN cơ bản

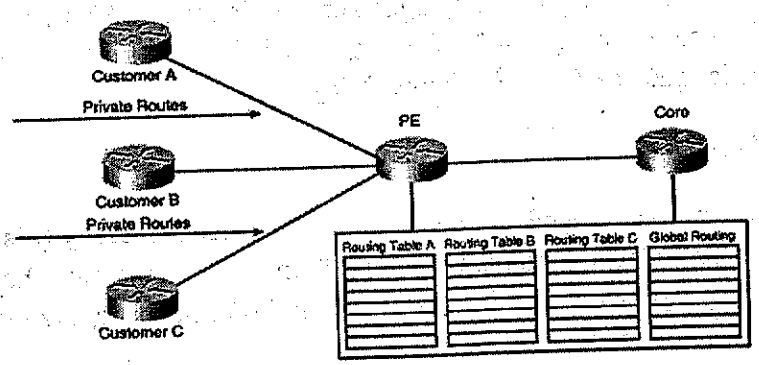


Hình 5.52 Mạng riêng với địa chỉ IP riêng

Trong công nghệ MPLS VPN, router ở phía khách hàng kết nối ngang hàng với router của nhà cung cấp dịch vụ. Khi muốn chuyển thông tin trong mạng riêng, router khách hàng sẽ cung cấp các thông tin và yêu cầu

định tuyến cho router của nhà cung cấp dịch vụ. Bên cạnh bảng định tuyến toàn cục, router của nhà cung cấp dịch vụ có bảng định tuyến riêng cho mỗi khách hàng

Trong Hình 5.53, không phải tất cả mạng riêng của khách hàng được chuyển qua bảng định tuyến toàn cục.



Hình 5.53 Quan hệ giữa router khách hàng và router nhà cung cấp dịch vụ

Thông qua VPN ở tầng 3, khách hàng có thể sử dụng tạo ra các đường kết nối an toàn trực tiếp từ khu vực này sang khu vực mạng khác trên nền mạng Internet toàn cầu với sự hỗ trợ của các ISP.

Các vấn đề của kiến trúc VPN truyền thống

VPN ở tầng 3 cũng có nhiều hạn chế. Ví dụ với giao thức MPLS chỉ hỗ trợ duy nhất IP. Khách hàng phải nhường quyền kiểm soát việc định tuyến của mình cho nhà cung cấp dịch vụ. Thứ hai, router của nhà cung cấp dịch vụ (PE) có thể bị quá tải. Để hỗ trợ khả năng mở rộng của hệ thống, nhà cung cấp dịch vụ phải sử dụng các router có cấu hình mạnh – và do đó đắt tiền.

Như đã nói ở trên, các dịch vụ kết nối ở tầng 2 tạo ra các đường truyền điểm nối điểm để xây dựng VPN. Để hỗ trợ truyền thông ở tầng 3, phải xây dựng thêm một mạng ở tầng 3. Kết quả của điều này là nhà cung cấp dịch vụ phải có hai mạng tách rời cho cả tầng 2 lẫn tầng 3 – một vấn đề khó khăn và tốn kém.

Một vấn đề của dịch vụ ở tầng 2 [illegible] độ tối đa (ATM với đường truyền OC48) không thể theo kịp với tốc độ ngày càng tăng của công nghệ mạng cục bộ (Ethernet). Do vậy nhà cung cấp dịch vụ phải tìm cách nâng cao hiệu suất và giảm chi phí quản lý. Điều này có thể đạt được bằng cách cho phép nhiều kết nối ở tầng 2 có thể được thiết lập qua trục IP/MPLS dùng chung.

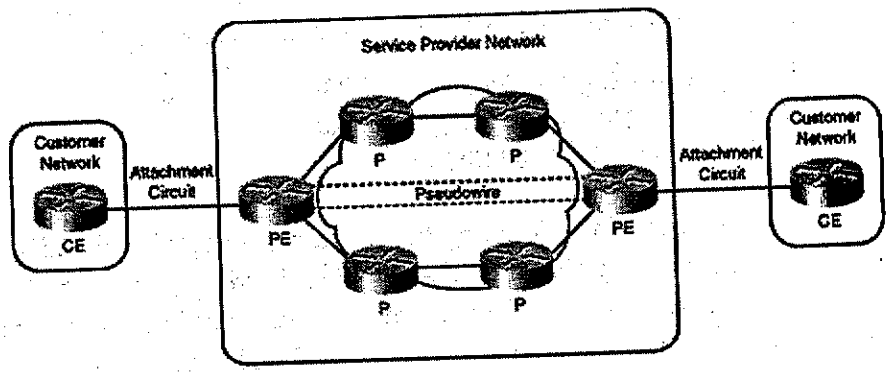
5.10.2 Mô phỏng giả dây dẫn (Pseudowire Emulation Overview)

Mục tiêu của cơ chế này là "tái tạo" các đặc điểm của dịch vụ chuyển mạch ở tầng 1 và tầng 2 (chẳng hạn TDM hay Frame Relay) trên nền tảng mạch chuyển mạch gói (PSN). Cơ chế này gửi các gói PDU giữa máy tính của khách hàng thông qua mạng của nhà cung cấp dịch vụ. Đối với khách hàng, cơ chế giả dây dẫn hoàn toàn trong suốt, nghĩa là khách hàng hoàn toàn không biết "mạch ảo" mà mình sử dụng được triển khai trên nền tảng chuyển mạch gói của nhà cung cấp dịch vụ.

Động lực thúc đẩy sự phát triển của công nghệ này là mong muốn hội tụ các công nghệ mạng cho phép triển khai một số dịch vụ mà trước đây chỉ có trên mạng chuyển mạch ảo trên công nghệ chuyển mạch gói.

Mô hình tham chiếu

Hình 5.54 minh họa mô hình tham chiếu của kiến trúc mô phỏng dây dẫn.



Hình 5.54 Mô hình giả dây dẫn

Các thiết bị ở phía nhà cung cấp dịch vụ (PE) nằm trong miền quản trị. Nhiệm vụ của miền này là cung cấp dịch vụ mô phỏng giả dây dẫn cho các thiết bị ở phía khách hàng thuộc cùng một miền quản trị.

Kết nối giữa PE và CE được thực hiện bằng mạch nối (attachment circuit) – có thể là cổng Etthernet, Ethernet VLAN, PPP session hay kết nối High-Level Data Link Control (HDLC), định danh kết nối Frame Relay (DLCI), định danh mạch ảo ATM.

"Giả dây dẫn" là một mạch ảo giữa hai PE để kết nối hai mạch nối. Mạch ảo này có thể được thiết lập thủ công hay tự động (qua cơ chế báo tín hiệu). Sau khi đã có một "giả dây dẫn" giữa hai PE, các frame dữ liệu nguyên thủy từ các mạch nối được đặt trong các gói dữ liệu của mạch ảo truyền từ đầu này sang đầu kia. Khi nhận được PDU mạch ảo, PE sẽ lấy ra frame ban đầu và gửi cho mạch nối.

Các thiết bị của nhà cung cấp (P) tạo nên hệ thống chuyển mạch gói ở giữa và hoàn toàn trong suốt với khách hàng. P cũng không "nhận biết" được luồng truyền thông "giả dây dẫn" và điều này giúp việc thiết kế mạng đơn giản hơn. Mạng trung tâm sẽ không phức tạp vì chỉ tập trung vào khía cạnh định tuyến và chuyển tiếp dữ liệu sao cho đạt hiệu suất cao nhất. Khi đó mạng cũng dễ mở rộng hơn và có thể cung cấp được nhiều mạch "giả dây dẫn hơn". Ở đây chúng ta cũng thấy lại quan điểm tổ chức mạng ở "lõi" hệ thống càng đơn giản càng tốt và độ phức tạp nên chuyển ra phía ngoài.

Giao thức và kiến trúc hệ thống

Giao thức của kiến trúc mô phỏng giả dây dẫn có ba tầng:

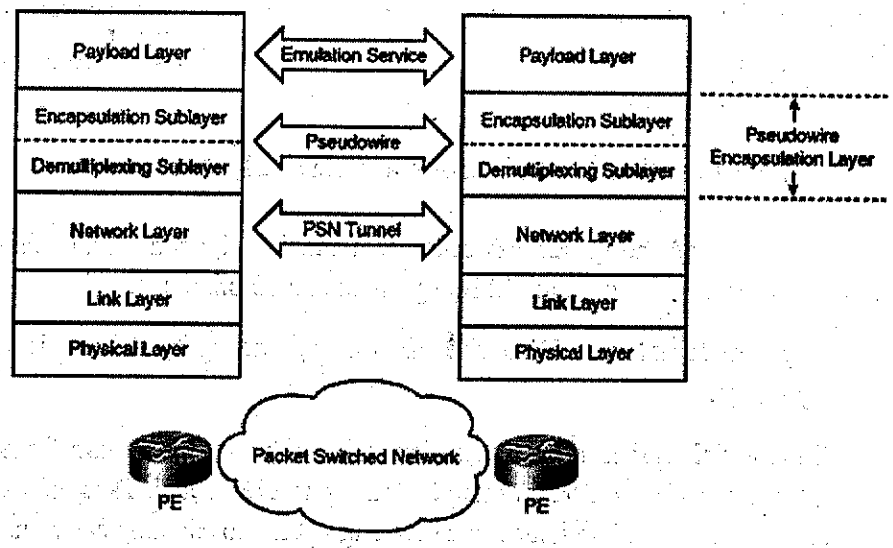
- Giao thức của Tầng chuyển mạch gói PSN
- Giao thức bao bọc dữ liệu của giả dây dẫn
- Giao thức của tầng dữ liệu (payload)

Tầng PSN quy định các thông tin về địa chỉ mạng của các thiết bị PE, ở đây có thể sử dụng giao thức địa chỉ IPv4, IPv6, hoặc nhãn MPLS. Các thiết bị mạng sử dụng tầng PSN để xác định tuyến đường chuyển tiếp của các gói tin trên mạch "giả dây dẫn". Bạn có thể hình dung mạch giả dây dẫn này như một đường ống tạo thành từ công nghệ chuyển mạch gói.

Tầng bao bọc dữ liệu bao gồm hai tầng con: tầng phân kênh và tầng bao bọc dữ liệu. Nhiệm vụ của tầng phân kênh là phân biệt các đường "giả dây dẫn" khác nhau cùng được thiết lập trên cùng một "đường ống" từ mạng chuyển mạch gói. Trong mỗi đường ống, mỗi giả dây dẫn có một định danh phân kênh duy nhất. Tầng bao bọc dữ liệu có nhiệm vụ đặt thêm một số thông tin về dữ liệu, những thông tin này sẽ được PE đầu kia sử dụng để loại bỏ để tạo ra frame ban đầu (từ payload) trước khi chuyển tiếp cho mạch nối đến CE.

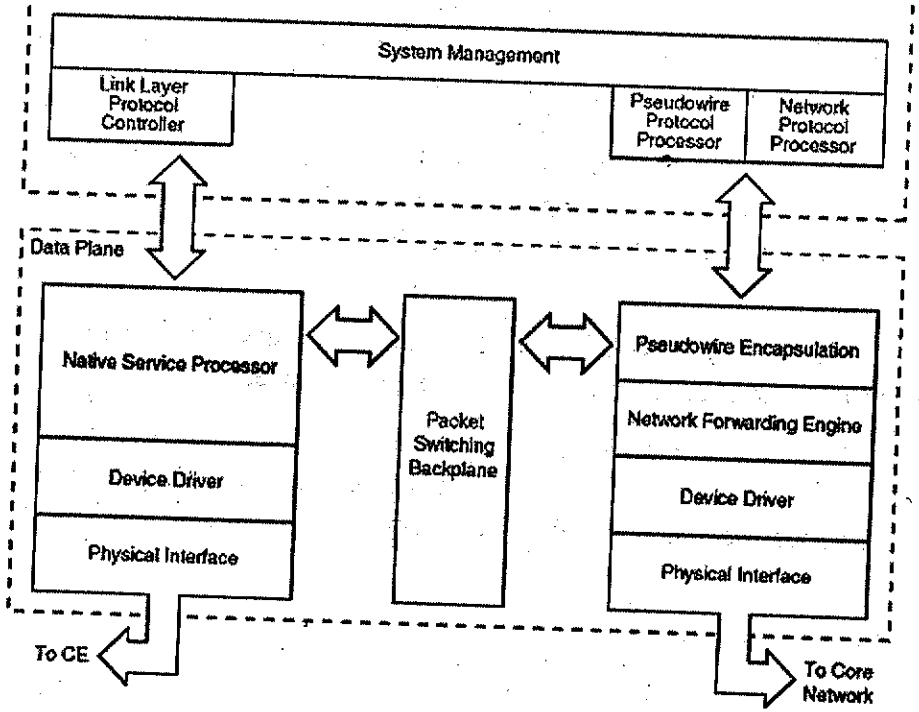
Tầng dữ liệu payload tải các kiểu dữ liệu giả dây dẫn khác nhau, có thể là frame Ethernet, ATM cell hay Frame Relay packet.

Hình 5.55 minh họa quan hệ của các tầng giả dây dẫn trên hai thiết bị PE. Mỗi tầng trên mỗi PE tương tác với tầng ngang cấp của phía PE bên kia thông qua các tầng bên dưới, và đến lượt mình lại cung cấp dịch vụ cho tầng cao hơn.



Hình 5.55 Mô hình các tầng trong "Giả dây dẫn"

PE đóng vai trò cực kỳ quan trọng trong kiến trúc. Trên thực tế, truyền thông giữa hai mạch nối diễn ra chủ yếu qua PE. Hình 5.56 minh họa kiến trúc của PE.



Hình 5.56 Kiến trúc PE (Thiết bị của nhà cung cấp dịch vụ)

Kiến trúc hệ thống của PE được chia thành hai phần chính: phần điều khiển và phần dữ liệu. Phần dữ liệu gồm có các bộ phận sau:

- **Physical interfaces**: Điều chế các tín hiệu tương tự thành bit và ngược lại.
- **Device drivers**: là tầng trung gian, với nhiệm vụ tạo ra các frame (phụ thuộc vào môi trường truyền) cho giao diện vật lí. Tầng nên sẽ che dấu các đặc điểm khác nhau của tầng vật lí cho các tầng cao hơn.
- **Native service processor** và **Pseudowire encapsulation**: Thực hiện biến đổi các gói dữ liệu (sẽ được trình bày kỹ hơn ở phần sau).
- **Network forwarding engine**: Khi nhận được dữ liệu từ bộ phận pseudowire encapsulation cùng với địa chỉ mạng đích, bộ phận

network forwarding engine tìm kiếm địa chỉ này trong các bảng chuyển tiếp. Nếu tìm thấy một giao diện ra, nó sẽ gói gói dữ liệu này trong một frame có cấu trúc phù hợp để gửi trên giao diện vừa tìm thấy. Nếu không tìm thấy, gói tin bị loại bỏ.

Phần điều khiển gồm các thành phần sau:

- **Link layer protocol controller**: Thực hiện giao thức báo hiệu trên đường truyền, chẳng hạn Frame Relay Local Management Interface (LMI) hay ATM Integrated Local Management Interface (ILMI), với mục đích thiết lập các mạch nối.
- **Pseudowire protocol processor** và **network protocol processor:** Tương ứng thực hiện các thủ tục báo hiệu trong giao thức "giả dây dẫn" và giao thức định tuyến. Các thiết bị PE sử dụng các thủ tục này để thiết lập các "giả dây dẫn" và tuyến đường để chuyển dữ liệu (xem Hình 5.56). Các thể lấy được các thông tin dùng để chuyển tiếp gói tin thông qua các thủ tục báo hiệu

Xử lý các gói tin từ dịch vụ ban đầu

Trong một số trường hợp, các gói dữ liệu của mạch nối có thể được chuyển tiếp ngay cho tầng giả dây dẫn. Tuy nhiên trong một số trường hợp, gói tin nguyên thủy này cần phải được xử lý trước khi áp dụng cơ chế bao bọc dữ liệu của giao thức "giả dây dẫn".

Thông thường, chúng ta phải áp dụng các thủ tục xử lý gói tin khác nhau cho các kiểu mạch nối khác nhau. Thậm chí cùng một kiểu nhưng các cấu hình khác nhau cũng có thể dẫn đến các thủ tục xử lý khác nhau.

The native service processor (NSP) xử lý bất kỳ gói tin nào truyền qua nó theo một cách thức phù hợp. Ví dụ khi nhận được một gói tin của giao thức PPP dưới dạng một frame HDLC, NSP loại bỏ các tiêu đề của HDLC sao cho phần dữ liệu của giao thức PPP có thể đặt trong khuôn dạng không phụ thuộc vào môi trường truyền. Khi gói dữ liệu của tầng "giả dây dẫn" chứa dữ liệu PPP đến thiết bị PE ở phía đầu kia, phần dữ liệu này sẽ được chuyển cho bộ phận NSP trước khi loại bỏ các tiêu đề của gói tin "giả

dây dẫn". Khi đó NSP biết rằng trường dữ liệu này cần phải đặt trong trường dữ liệu của giao thức PPP để gửi đi tiếp.

Xử lý việc bao bọc dữ liệu ở tầng giả mã dây

Sau khi qua giai đoạn xử lý gói tin ban đầu, trường dữ liệu có thể được bao bọc theo khuôn dạng của giao thức giả mã dây. NSP có thể sử dụng thêm một số thông tin đặc thù của trường dữ liệu để chuyển qua cho pseudowire encapsulation processor (PEP)

Các thông tin điều khiển này được sử dụng để báo hiệu cho từng gói tin (cần thiết đối với một số kiểu dịch vụ). Ngoài các thông tin này, quá trình bao bọc có thể đặt thêm các thông tin về thời gian cho truyền thông theo thời gian thực hoặc các thông tin đánh số thứ tự giúp phía nhận phát hiện các gói tin không đúng thứ tự.

Truyền thông qua mạng PSN

Phụ thuộc vào kiến trúc của PSN, chúng ta có thể sử dụng IP hoặc MPLS để gửi gói tin của tầng giả dây dẫn. Kiến trúc PSN không chỉ xác định cách thức bao bọc gói tin ở tầng mạng mà còn xác định khuôn dạng của bộ phân kênh của giả dây dẫn. Ví dụ nếu IP là giao thức tầng mạng của PSN, thì trong trường Upper layer của gói tin IP sẽ xác định cần gửi gói tin này cho bộ phân kênh nào.

Tài liệu tham khảo

[1]. J. Kurose and K. Ross, ***Computer Networking: A Top-Down Approach Featuring the Internet***", Addison-Wesley, 3rd edition, May 2004.

[2]. Andrew S. Tanenbaum, **Computer Networks**, *Prentice Hall PTR, 2004*

[3]. S.Keshav, *An Engineering Approach to Computer Networking*, Addison-Wesley, 2003

[4]. Jean Walrand, ***Communication Networks: A first cost***, second edition, McGraw Hill, 2003

[5]. Douglas E. Comer, *Computer Networks and Internet with Internet Application*, 3rd edi, Prentice Hall, 2003.

[6]. Michael A. Gallo and William M. Hancock, ***Computer Communications and Network Technologies***, Brooks/Cole Thomson Learning, 2003

[7]. William Stalling, ***Data and Computers Communications***, 6th edition, Prentice Hall, 2003.

[8]. Behrouz A. Forouzan, ***Data Communications and Networking***, McGraw Hill, 2003

[9]. Ramesh Subramanian and Brian D. Goodman, ***Peer To Peer Computing The Evolution Of A Disruptive Technology***, IDEA Group Publishing, 2006

[10]. Wei Luo, *Layer 2 VPN Architectures*, Cisco Press, 2006

TRƯỜNG ĐẠI HỌC KINH TẾ QUỐC DÂN
KHOA TIN HỌC KINH TẾ

MẠNG MÁY TÍNH

(TÀI LIỆU THAM KHẢO)

HÀ NỘI 2004

MỤC LỤC

(TÀI LIỆU THAM KHẢO) 1

PHẦN I NHỮNG VẤN ĐỀ CƠ BẢN VỀ MẠNG MÁY TÍNH 5

§ 1. Các khái niệm về mạng máy tính 5
1.1 Khái niệm về mạng (Network concepts). 5
1.2 Ứng dụng của mạng máy tính trong các tổ chức. 5
1.3 Một số yếu tố đặc trưng của mạng 5
1.4 Phân loại mạng máy tính. 7

§ 2. Chuẩn mạng 7
2.1 Chuẩn 7
2.2 Mô hình tham chiếu 9
2.2.1 Mô hình tham chiếu OSI (Open System Interconnection) 9
(OSI Reference Model) 9
2.3 Mô hình tham chiếu TCP/IP 16
* So sánh với mô hình tham chiếu OSI: 18
2.4 Một số chuẩn mạng 19

§ 3.hình trạng mạng (Network Topology) 21

§ 4. Các thiết bị liên kết mạng 23
4.1. Dây cáp mạng 23
4.2. Vỉ mạch mạng (Network Interface Card - NIC) 25
4.3. Bộ tập trung (Hub) 27
4.4. Chuyển mạch (Switch) 27
4.5. Bộ phát lặp (Repeater) 26
4.6. Cầu nối (Bridge) 28
4.7. Bộ dẫn đường (Router) 29
4.8. Cổng kết nối (Getway) 29

§ 5. Các giao thức truyền trên mạng (Protocol) 30
5.1 Khái niệm 30
5.2. Các bộ giao thức phổ biến 32

§ 6. Thiết kế mạng cục bộ 32
6.1. Qui tắc thiết kế mạng: 33
6.2. Phương pháp thiết kế : 33
6.3. Một số thiết kế mạng 34

PHẦN II HỆ ĐIỀU HÀNH MẠNG NOVEL NETWARE 39

§ 1. Giới thiệu mạng NOVELL NETWARE 39
1.1. Sự phát triển của Novell Netware 39
1.2. Novell Directory Services 39

§ 2. Quản trị FILE SERVER 41

§ 3. Cài đặt mạng NOVELL NETWARE 42
3.1. Yêu cầu về phần cứng - phần mềm 43

3.2. Các bước cài đặt: 43
3.3. Cài đặt WORK STATIONS (DOS) 49
3.3. Cài đặt WORK STATIONS (Windows 9.X) 49

§ 4. Quản trị hệ thống thư mục và files 52
4.1. Cấu trúc thư mục và files của Novell NetWare 52
4.2. Thiết kế hệ thống thư mục và FILES 53
4.3. Quyền hạn 54
4.4. Thuộc tính 55

§ 5. Quản trị USER Và GROUP 56
5.1. Quản trị USER 57
5.2. Quản trị GROUP 61
5.3. Quản trị USER ACCOUNT 62
5.4. Thao tác với USER ACCOUNT 63

§ 6. LOGIN SCRIPTS 64
6.1. Khái niệm: 64
6.2. Khởi tạo LOGIN SCRIPT 65
6.3. Các lệnh hệ thống trong LOGIN SCRIPT 66

§ 7. Quản trị dịch vụ in trên mạng 67
7.1.Tổng quan về công việc in trên mạng (DOS client) 67
7.2. Cài đặt PRINT SERVER , cài đặt máy in, cài đặt PRINT QUEUE, ... 68
7.3. Ở work station : nối máy in với mạng 69
7.4. Tổ chức máy in PRINT SERVER 69

§ 8. Một số lệnh cơ bản của Novell Netware 70

PHẦN III HỆ ĐIỀU HÀNH MẠNG WINDOWS 2000 76

§ 1. Tổng quan về Microsoft windows 2000 76
1.1. Windows 2000 Server 76
1.2. Activate Directory (AD) 77
1. 3. Địa chỉ trong giao thức TCP/IP 79

§ 2. Cài đặt WINDOWS 2000 SERVER 80
2.1. Một số vấn đề chung 80
2.2. Cài đặt windows 2000 server 81
2.3. Cài đặt Activate Directory 85
2.4. Hiệu chỉnh các tham số 89
2.5. Cơ sở dữ liệu Registry 91

§ 3. Quản trị USER và GROUP 94
3.1. Khái niệm user và group 94
3.2. Khái niệm quyền hạn 98
3.3. Thiết kế hệ thống user và group 100
3.4. Tạo user và group 101
3.5. Login Script 105

§ 4. Quản trị hệ thống thư mục và files 106
4.1 Cấu trúc thư mục của Windows 2000 106

4.2 Thiết kế hệ thống thư mục cho người dùng 107
4.3 Quản lý quyền truy cập 107
4.4 Cấp phát hạn ngạch đĩa (quota) 111

§ 5. Cài đặt client, Truy nhập vào mạng 113
5.1. Hệ điều hành DOS 113
5.2. Hệ điều hành WINDOWS 9X 116
5.3. Truy nhập vào Workstation khác trong mạng (mạng ngang hàng) 118

§ 6. Quản trị phương tiện lưu trữ 118
6.1. Một số khái niệm 118
6.2. Quản lý đĩa cứng 120

§ 7. Quản trị dịch vụ in ấn trên mạng 121
7.1. Nguyên lý in ấn trên windows 2000 : 121
7.2. Một kiểu cấu hình in trên mạng: 122

§ 8. Quản trị sao lưu dữ liệu 125
8.1. Khái niệm 125
8.2. Sao lưu hệ thống 125
8.3. Sao lưu dữ liệu 126

§ 9. Một số dịch vụ mạng của Windows 2000 127
9.1. Giao thức cấu hình máy động (Dynamic Host Configuration Protocol -DHCP) . 127
9.2. Domain Name System (DNS) 130

§ 10. Tích hợp Novell Netware với Windows 2000 133
X.1. Khả năng liên kết với Novell Netware 134
10.2. Gắn kết với NWLink 134
10.3. Cấu hình dịch vụ cổng nối cho Novell 134

§ 11. Một số tiện ích 136

TỪ ĐIỂN THUẬT NGỮ 139

TÀI LIỆU THAM KHẢO 144

PHẦN I

NHỮNG VẤN ĐỀ CƠ BẢN VỀ MẠNG MÁY TÍNH

§ 1. CÁC KHÁI NIỆM VỀ MẠNG MÁY TÍNH

1.1 Khái niệm về mạng (Network concepts).

Mạng máy tính là tập hợp các máy tính và các thiết bị được liên kết với nhau thông qua đường truyền vật lý hoặc lôgíc (môi trường truyền thông – communication medium) và tuân theo những qui tắc truyền thông nào đó gọi là mạng máy tính.

Mạng máy tính được thiết lập để có thể chia sẻ các tài nguyên mạng.

Đường truyền (môi trường truyền thông) dùng để truyền dẫn tín hiệu điện tử (electronic signal). Đường truyền có thể là hữu tuyến hoặc vô tuyến.

- Các qui tắc truyền thông là cơ sở để các máy tính có thể liên lạc được với nhau.
- Tài nguyên mạng bao gồm: thiết bị phần cứng, dữ liệu và ứng dụng.
- Chủ thể làm việc trên hệ thống mạng: con người

1.2 Ứng dụng của mạng máy tính trong các tổ chức.

Mạng được sử dụng chủ yếu để chia sẻ, dùng chung tài nguyên, và giao tiếp trực tuyến.

Chia sẻ thiết bị ngoại vi (devices): ổ đĩa mạng, máy in, modem, và một số thiết bị khác có thể được cài đặt trên mạng để mọi người có thể dùng chung các thiết bị này.

Chia sẻ dữ liệu (Data): các máy trên mạng đều có thể cập nhật và khai thác thông tin từ máy chủ dùng chung trên hệ thống mạng.

Chia sẻ file (file sharing): các máy tính trên mạng có thể chia sẻ việc sử dụng các file với nhau.

Chia sẻ các ứng dụng: có thể cài đặt các ứng chạy trên mạng để có thể khai thác từ nhiều máy trên mạng hoặc thiết lập các ứng dụng làm việc theo nhóm trên mạng.

Hỗ hợ giao tiếp (liên lạc) trên mạng: chẳng hạn như gửi nhận thông điệp (message), tán gẫu (chat - instant message), thư điện tử (e-mail); reminder, schedule,…

Quản lý tập trung và bảo mật tài nguyên.

…

1.3 Một số yếu tố đặc trưng của mạng

Đường truyền:

Là phương tiện dùng để truyền tín hiệu điện tử (electric signal) giữa các nút mạng, các tín hiệu này là thông tin được biểu thị dưới dạng các xung nhịp. Tùy

theo điều kiện kỹ thuật người ta có thể sử dụng các loại đường truyền khác nhau, có hai kiểu đường truyền:

- Đường truyền hữu tuyến: Các nút mạng được nối với nhau trực tiếp bằng dây cáp kim loại hoặc cáp quang (fibre optic).
- Đường truyền vô tuyến: Các nút mạng được nối với nhau thông qua các thiết bị điều chế/giải điều chế (modulation and demodulation) và truyền tín hiệu qua sóng vô tuyến.

Kiến trúc mạng (network architecture)

Kiến trúc mạng máy tính (network architecture) thể hiện cách thức nối các máy tính, thiết bị mạng với nhau và tập hợp các quy tắc, quy ước mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để đảm bảo cho mạng hoạt động tốt.

Kiến trúc mạng bao gồm toàn bộ những thiết kế và nền tảng của mạng.

Khi nói đến kiến trúc của mạng người ta muốn nói tới hai vấn đề là hình trạng mạng (Network topology) và giao thức mạng (Network protocol).

- Network Topology: Cách thức kết nối các máy tính với nhau về mặt hình học mà ta gọi là topo của mạng. Các hình trạng mạng cơ bản đó là: hình sao, hình bus, hình vòng.
- Network Protocol: Tập hợp các quy ước truyền thông giữa các thực thể truyền thông mà ta gọi là giao thức (hay nghi thức) của mạng. Các bộ giao thức thường gặp nhất là: TCP/IP, NETBIOS, IPX/SPX, . . .

Hệ điều hành mạng

Hệ điều hành mạng là một phần mềm hệ thống mạng (Network Operating System) làm nhiệm vụ quản lý các luồng thông tin vào ra, kiểm tra và phát hiện sai sót của thiết bị, quản lý và phân chia tài nguyên, quản lý hệ thống tệp tin, thi hành chế độ bảo mật.

Các chức năng chính của hệ điều hành mạng:

+ Quản lý người dùng và các công việc trên hệ thống: Hệ điều hành đảm bảo giao tiếp giữa người sử dụng với các chương trình ứng dụng, giữa các ứng dụng với thiết bị của hệ thống.

+ Cung cấp các dịch vụ mạng (network services) và các tiện ích (utilities) cho việc khai thác hệ thống.

+ Bảo mật hệ thống mạng.

Các hệ điều hành mạng thông dụng hiện nay là: Windows NT, Windows 2000, Unix (Sco Unix, Digial Unix, Sun Solaris), Linux (Redhat linux, Debian Linux,..), Novell netware.

Máy chủ mạng(Server) và máy trạm (Workstation)

- Server: Máy chủ mạng thường là một máy tính có cấu hình đủ mạnh, được cài đạt hệ điều hành mạng cùng với các dịch vụ mạng (network services) và

các tiện ích để làm nhiệm vụ cung cấp tài nguyên, dịch vụ mạng tới người sử dụng và quản lý mạng (network supervising).

- Workstation: máy thành viên trên mạng thường được người sử dụng dùng để khai thác các tài nguyên và dịch vụ mạng.

1.4 Phân loại mạng máy tính.

* Phân loại mạng theo phạm vi nối máy:

- Mạng LAN (Local Area Network): Là một mạng máy tính ở quy mô nhỏ kết nối các máy tính trong phạm vi một văn phòng, cơ quan hoặc trường học (có bán kính cỡ vài trăm mét đến vài km), mạng này thường có tốc độ truyền thông lớn. Nó còn được gọi là mạng cục bộ.
- Mạng MAN (Metropolitan Area Network): Là một mạng được cài đạt trong phạm vi một đô thị hay trumg tâm kinh tế xã hội (bán kính khoảng 100km).
- Mạng WAN (Wide Area Network): Là một mạng máy tính diện rộng có phạm vi bao trùm một quốc gia hay lục địa.
- Mạng GAL (Global Area Network): Một mạng toàn cầu, kết nối các hệ thống mạng trên toàn cầu.

* Phân loại mạng theo cách thức khai thác/chia sẻ dữ liệu

+ Mạng ngang hàng (Peer-to-Peer): Tất cả các máy tính trong mạng đều có vai trò như nhau và tất cả đều có thể chia sẻ tài nguyên với nhau.

Các hệ điều hành được sử dụng trong mạng ngang hàng như:

- Windows for workgroup (Windows 3.11).
- Lantastic Antisoft
- Windows 98/ME/2K/XP

+ Mạng phân quyền (Client/Server): Tất cả các máy tính trong mạng (máy khách – Client) đều khai thác và chia sẻ tài nguyên thông qua một máy chủ phục vụ (Server).

Các hệ điều hành mạng phân quyền (client/server) phổ biến như:

- Novell NetWare (2.x / 3.x / 4.x /5.x/6.x).
- Win dows NT , Windows 2000

* Phân loại theo cách thức nối máy

+ Điểm điểm (point to point)

+ Quảng bá (broadcast)

* Phân loại theo phương thức chuyển mạch

+ Chuyển mạch kênh (Circuit Switching Network)

+ Chuyển mạch thông điệp (Message Switching Network)

+ Chuyển mạch gói (Packet Switching Network)

§ 2. CHUẨN MẠNG

2.1 Chuẩn

2.1.1 Khái niệm

Trong công nghiệp truyền thông, từ lâu các chuẩn đó được chấp nhận như là yêu cầu cần thiết để quản lý các đặc tính về thủ tục, điện tử và vật lý của thiết bị truyền thụng. qua đó cho phép các thiết bị tương tác với nhau.

- Các ưu điểm của việc xây dựng và tuõn theo chuẩn:
  - Đảm bảo cho một thị trường rộng lớn cho các thiết bị và phần mềm.
  - Cho phép các sản phẩm của các nhà cung cấp khác nhau truyên thông được với nhau
- Các nhược điểm:
  - Không thay đổi công nghệ (Freeze technology)
  - Cú thể cú nhiều chuẩn cho cựng một loại thiết bị.

Các tổ chức quốc tế chính, quan trọng trong việc xây dựng và đưa ra các chuẩn :

- Internet Society
- ISO
- ITU-T (formally CCITT)

…

ISO (International Standards organization): là tổ chức tiêu chuẩn quốc tế làm việc dưới sự bảo trợ của Liên hợp quốc với các thành viên là các cơ quan tiêu chuẩn quốc gia, trong ISO được chia thành các ủy ban kỹ thuật phụ trách các lĩnh vực khác nhau. Các công trình chuẩn hóa sau khi được ISO biểu quyết thông qua sẽ được công bố như tiêu chuẩn quốc tế chính thức (International Standard - IS).

CCITT (Commite Consultatif International pour le Telegraphe et la Telephone): Tổ chức tư vấn quốc tế về điện tín và điện thoại, làm việc dưới sự bảo trợ của Liên hợp quốc với các thành viên là cơ quan bưu chính viễn thông của các quốc gia hoặc các hãng xuyên quốc gia. Các tiêu chuẩn của CCITT được coi là các khuyến nghị. CCITT đã ban hành khuyến nghị loại V: liên quan đến vấn đề truyền dữ liệu, loại X: liên quan đến các mạng truyền dữ liệu công cộng ...

Sự tồn tại của cả hai tổ chức giải thích cho sự giao thoa giữa các chức năng xử lý và truyền tin trong các ứng dụng viễn thông-tin học.

IEEE (Institute of Electrical and Electronics Engineers) Viện kỹ thuật điện tử, một tổ chức đưa ra các chuẩn cụ thể của mạng.

ANSI (American National Standards Institute): là tổ chức của các nhóm doanh nghiệp và công nghiệp Mỹ, chuyên phát triển các tiêu chuẩn thương mại và truyền thông: mã, bảng chữ cái, lược đồ tín hiệu.

Common Open Software Environment (COSE): Môi trường chung cho phần mềm mở

Open Software Foundation (OSF): cơ sở phần mềm mở

SQL Access Group (SAG): Nhóm truy cập ngôn ngữ hỏi có cấu trúc

Object Management Group (OMG): nhóm quản lý đối tượng

Corporation for Open Systems (COS): tập đoàn các hệ mở

Electronics Industries Association (EIA): Hiệp hội các ngành công nghiệp điện tử

2.1.2 Qui luật và tiến trình liên lạc

Các mạng đều dựa vào nhiều qui luật để trao đổi thông tin. Một số quy trình được các mạng chuẩn điều hành như sau:

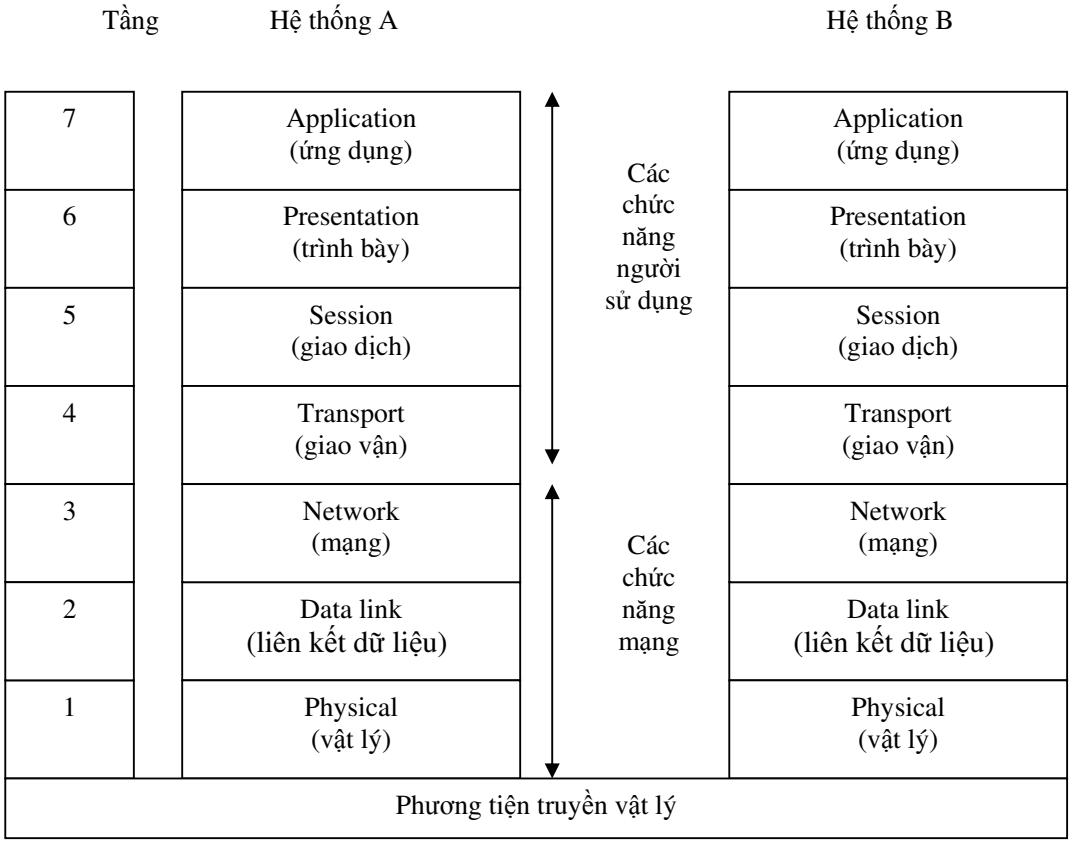
- Qui trình để nối và cắt liên lạc
- Tín hiệu để trình bày dữ liệu trên phương tiện truyền thông.
- Loại tín hiệu được dùng.
- Phương pháp truy cập để chuyển tiếp một tín hiệu liên lạc.
- Phương pháp truyền trực tiếp một thông điệp gửi tới nơi đến.
- Thủ tục dùng để kiểm soát tốc độ truyền dữ liệu.
- Phương pháp dùng cho máy điện toán khác loại liên lạc được với nhau.
- Các cách bảo đảm cho thông điệp được nhận đúng cách

2.2 Mô hình tham chiếu

Năm 1978 tổ chức tiêu chuẩn quốc tế – ISO đã ban hành tập hợp đặc điểm kỹ thuật mô tả kiến trúc mạng dành cho việc kết nối những thiết bị không cùng chủng loại. Ban đầu tài liệu này áp dụng cho những hệ thống mở với nhau do chúng có thể dùng chung giao thức và tiêu chuẩn để trao đổi thông tin. Vào năm 1984, ISO đã phát hành bản sửa đổi mô hình này và gọi là mô hình tham chiếu mạng hệ mở OSI (Open Systems Interconnection)

2.2.1 Mô hình tham chiếu OSI (Open System Interconnection) (OSI Reference Model)

Mô hình tham chiếu OSI: Mô hình này là một khung mà các tiêu chuẩn lập mạng khác nhau có thể khớp vào. Nó định rõ các mặt nào của hoạt động của mạng có thể nhằm đến bởi các tiêu chuẩn mạng khác nhau. Mô hình OSI có 7 tầng, hai tầng đáy là các tầng có tác động thực tế nhất trên các mạng nhỏ. Các tiêu chuẩn mạng như Ethernet và Tonken Ring là các tiêu chuẩn tầng 1 và tầng 2. Các tầng cao hơn trong mô hình OSI không tạo ra các tiêu chuẩn phổ biến.



Mô hình OSI

Liên lạc giữa các tầng ngang hàng

Tầng thứ i của một máy tính sẽ hội thoại với tầng thứ i của một máy khác.

Các qui tắc và qui ước được dùng trong việc hội thoại đó được gọi là **giao thức** (protocol) của mức i.

Giữa mỗi cặp tầng kề nhau tồn tại một **giao diện** (interface) xác định các thao tác và các dịch vụ mà tầng dưới cung cấp cho tầng trên.

Mỗi tầng sẽ truy cập với tầng trên và dưới nó thông qua các điểm truy cập dịch vụ (Service Access Point -SAP).

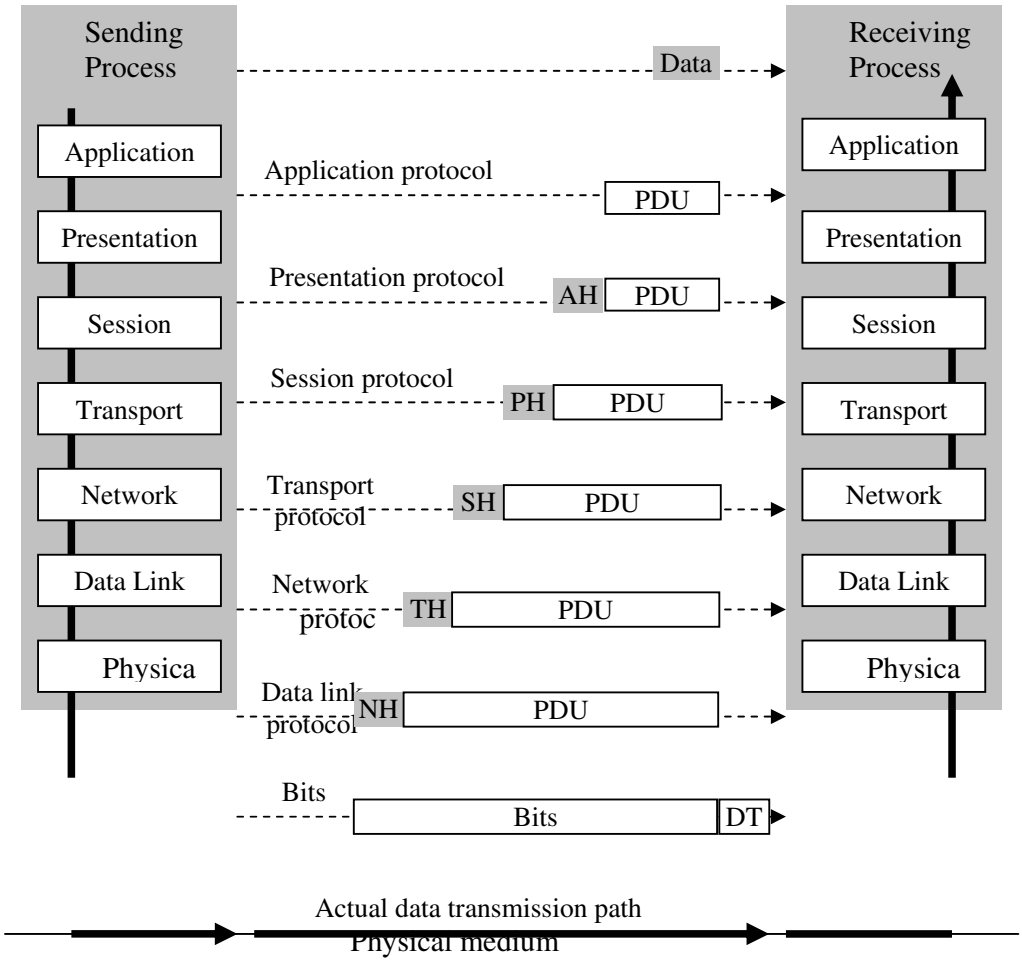
Trong thực tế dữ liệu không được truyền trực tiếp từ tầng thứ i của của máy này sang tầng thứ i của máy khác (riêng tầng 1 dữ liệu truyền trực tiếp). Mỗi tầng chuyển dữ liệu và thông tin điều khiển xuống tầng ngay dưới nó và tiếp tục cho đến tầng 1. Ở tầng 1 sẽ có đường truyền thông vật lý tới tầng thấp nhất của máy tương ứng và từ đó dữ liệu và thông tin điều khiển lại được truyền ngược lên các tầng trên. Như vậy ở các tầng trên tầng 1 sẽ chỉ xác định các đường

truyền LOGIC tới tầng tương ứng của các máy khác. (đường đứt nét biểu thị đường truyền LOGIC, đường liền nét biểu thị đường truyền vật lý).

Ví dụ: Giả sử có 2 máy A và B cần truyền dữ liệu cho nhau. (Giả thiết dữ liệu được tạo ra tại tầng ứng dụng - tầng 7 - của máy A).

Thông tin xuất phát từ mức 7 của máy A chuyển tới mức 6 bao gồm thông tin của mức 7 cộng với phần Header của mức sáu. Cứ như vậy cho đến mức 1 và được chuyển ra đường truyền. Thông tin nhận được ở mức 1 của máy B được kiểm tra phần Header của chính mức 1 nếu đúng thì phần Header sẽ được cắt bỏ và chuyển phần còn lại lên mức 2 của máy B. Nếu kiểm tra phần Header không đúng thì sẽ báo lại cho máy A truyền lại. Quá trình cứ tiếp tục như vậy cho tới mức 7 của máy B.

Đường truyền dữ liệu trong OSI:



Tầng 1 (physical - vật lý):

Tầng này không xác định rõ phương tiên vật lý nhưng nó vẫn nêu rõ những yêu cầu về những phương tiên kỹ thuật cần thiết trên phương diện vật lý (chẳng hạn các cable mạng phù hợp chuẩn IEEE 802.2). Cung cấp các phương tiện điện, cơ hàm và thủ tục để khởi động, duy trì và hủy bỏ các liên kết vật lý cho phép truyền các dòng dữ liệu ở dạng bit.

Tầng physical định nghĩa cách kết nối dây cáp với card mạng, ví dụ nó định rõ bộ nối có bao nhiêu chân, kỹ thuật truyền nào được dùng để gửi dữ liệu lên cáp mạng. Tầng này chịu trách nhiệm truyền dữ liệu theo mã nhị phân, nó định rõ mã hóa dữ liệu và sự đồng bộ hóa bit, bảo đảm rằng khi máy nguồn gửi bít 1, máy đích sẽ nhận được bít 1.

Các thành phần hoạt động ở tầng vật lý:

Bộ chuyển tiếp (repeater), là một thiết bị mạng nó lặp lại tín hiệu từ một cổng rồi chuyển tiếp đến những cổng khác. Nó không lọc hoặc diễn giải điều gì mà chỉ lặp lại để khuyếch đại tín hiệu truyền trên mạng (Amplifiy signal – khuyếch đại tín hiệu). Do vâyh các thiết bị loại này thường khuếch đại cả tín hiệu nhiễu (noise signal) .

Tầng 2 (data link - liên kết dữ liệu):

Tầng này làm nhiệm vụ nhận nhận dữ liệu từ tầng mạng (network layer) rồi chia thành các khung dữ liệu (data frame) và gửi đến tầng vật lý (physical layer). Tại đầu nhận, tầng này đóng gói dữ liệu thô (chưa xử lý) từ tầng physical thành từng khung dữ liệu. Tại tầng này có thực hiện việc điều khiển luồng, cắt/hợp dữ liệu và phát hiện sai sót.

Tóm lại tầng data link chịu trách nhiệm chuyển khung dữ liệu không lỗi từ nút mạng này sang nút mạng khác thông qua tầng physical. Thông thường khi gửi một khung dữ liệu nó chờ tín hiệu báo nhận từ nút nhận.

Tầng data link của nút nhận sẽ dò tìm bất cứ vấn đề nào có thể xảy ra trong quá trình truyền. Khung dữ liệu nào không được báo nhận hoặc bị hư tổn trong quá trình truyền sẽ bị gửi trả lại.

Tầng data link được chia làm 2 lớp phụ là:

- Điều khiển truy nhập phương tiện (Media Access Control – MAC): lớp phụ MAC điều khiển việc truy nhập thiết bị, phương tiện truyền thông để truyền dữ liệu từ tầng MAC của thiết bị này sang tầng MAC của thiết thị kia (chẳng hạn card mạng).

Cụ thể tầng MAC sẽ quản lý các thiết bị mạng truy cập vào môi trường truyền thông, cung cấp địa chỉ (MAC Address) để liên lạc giữa các thiết bị trên mạng.

- Điều khiển liên kết logic (Logical Link Control – LLC): tầng này làm nhiệm vụ giao tiếp với tâng trên, kiểm soát luồng và lỗi, thiết lập và duy trì liên kết giữa các thực thể tham gia truyền thông.

Thành phần hoạt động:

Cầu nối (bridge - thiết bị mạng tại tầng liên kết dữ liệu): Bridge làm nhiệm vụ chuyển các khung dữ liệu (data frame) từ mạng này sang mạng khác (cầu nối giữ các mạng). Nhờ việc kiểm tra địa chỉ MAC tại tầng liên kết dữ liệu nên nó có thể phát hiện ra đích đến của khung dữ liệu là nằm **trong** mạng hay **ngoài** mạng dựa vào một bảng địa chỉ MAC (MAC Address table) của các nút (Node) trong mạng. Khi một khung dữ liệu đến, Bridge kiểm tra địa chỉ đích của khung và đưa khung đến mạng chứa thiết bị đích hoặc nút đích. Nếu nút đích ở cùng mạng với nút nguồn bridge sẽ không đưa đến các mạng khác. Nếu cầu không tìm được địa chỉ đích, nó đưa khung đến tất cả các mạng trừ mạng nguồn.

Tầng 3 (network - mạng):

Tầng mạng được hình thành nhằm đảm bảo trao đổi thông tin giữa các mạng con trong một hệ thống mạng lớn. Tầng này quyết định đường đi cho các gói số liệu, khả năng định tuyến dựa trên cơ chế đánh địa chỉ thống nhất mà lớp mạng qui định cho các nút mạng. Ngoài ra nó còn kiểm soát dòng dữ liệu trên các tuyến để tránh tắc nghẽn hoặc mất mát thông tin. Tóm lại tầng này phụ trách những việc sau:

- Chọn đường đi, cung cấp địa chỉ.
- Kiểm soát lỗi và thông lượng.
- Chuyển mạch và dọn đường.
- Chia thông tin lại thành từng bó và lựa chọn đường đi đến đích cho từng bó khác nhau sao cho thời gian truyền tin là ngắn nhất và tin cậy nhất.
- Hợp nhất các bó thông tin tại nút nhận.

Tầng này phụ trách việc thiết lập các tuyến đường giữa các nút mạng khác nhau. Chính nhờ tầng này mà các mạng lắp theo các chuẩn khác nhau có thể liên kết với nhau.

* Định địa chỉ trên tầng mạng: Tầng mạng làm nhiệm vụ đánh địa chỉ các máy tham gia truyền thông trên mạng. Địa chỉ này chỉ là địa chỉ Logic để thay thế cho các địa chỉ vật lý, vì trong các mạng lớn việc sử dụng địa chỉ chỉ vật lý là không thực tế, không đáp ứng được yêu cầu truyền tin. Tầng mạng sử dụng địa chỉ logic để định tuyến (Router) và lọc bỏ gói tin (packet filter).
* Phân phối gói tin: Một gói tin được gửi từ một phần trên mạng của máy nguồn và đi theo nhiều con đường khác nhau của mạng để đến máy đích. Tầng mạng sẽ giám sát quá trình chọn đường truyền và phân phối gói tin qua mạng.
* Các kỹ thuật chuyển mạch cơ bản: chuyển mạch kênh (Circuit switching), chuyển mạch thông điệp (Message switching), chuyển mạch gói (Packet switching).

+ Chuyển mạch kênh (Circuit switching): Khi có hai nút cần truyền thông với nhau thì giữa chúng sẽ thiết lập một kênh cố định và duy trì kết nối đó cho tới khi hai bên ngắt liên lạc. Các dữ liệu chỉ truyền đi theo con đường cố định đó. Kỹ thuật này cung cấp cho các thiết bị một băng tần xác định. Phương pháp này

có ưu điểm là đường truyền thông suốt, tốc độ ổn định. Nhưng cũng có các nhược điểm như quá trình thiết lập sự kết nối thiết bị chậm, đường truyền bị chiếm giữ ngay cả khi không có dữ liệu truyền qua dẫn đến lãng phí đường truyền.

+ Kỹ thuật chuyển thông điệp (Message switching): thông điệp là một đơn vị dữ liệu của người sử dụng có khuôn dạng được quy định trước. Mỗi thông điệp tin có chứa các thông tin điều khiển trong đó chỉ rõ đích cần truyền tới của thông điệp. Căn cứ vào thông tin điều khiển này mà mỗi nút trung gian có thể chuyển bản tin tới nút kế tiếp trên con đường dẫn tới đích của bản tin. Kỹ thuật này có thể dùng chung kênh dữ liệu để nâng cao hiệu suất sử dụng giải tần, có khả năng lưu trữ bản tin đến khi có kênh truyền vì vậy giảm mật độ ùn tắc trên mạng. Tuy nhiên nhược điểm chính của nó là không phù hợp với các ứng dụng thực tế như truyền dữ liệu,truyền thanh.

+ Kỹ thuật chuyển mạch gói (Packet switching): ở đây bản tin hoặc thông điệp được chia ra thành nhiều gói nhỏ hơn được gọi là các gói tin (packet/diagram) có khuôn dạng qui định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và địa chỉ đích (người nhận) của gói tin. Các gói tin của cùng một tin báo có thể được gởi đi qua mạng tới đích theo nhiều con đường khác nhau. Chuyển mạch gói không lưu trữ gói tin lâu (ví kích thước packet < mesage) nên tuyến đường của gói tin qua mạng sẽ nhanh và hiệu quả hơn so với kỹ thuật chuyển mạch tin.

* Loại giao thức: Tầng mạng trong mô hình OSI xác định định tuyến để các gói tin từ máy nguồn đến máy đích qua nhiều mạng khác nhau. Trong tầng này sử dụng hai loại giao thức cơ bản để truyền tin là: Giao thức hướng kết nối (Connection Oriented) và Giao thức không hướng kết nối (Connectionless).

+ Giao thức hướng kết nối (Connection Oriented): Trong giao thức bao giờ cũng thiết lập kết nối (Connection) trước khi truyền số liệu, tức là giữa nút đích và nút nguồn sẽ thiết lập một liên kết logic để truyền số liệu (phân phối các một cách có thức tự các gói tin qua kênh logic này và có báo nhận để đảm bảo dữ liệu đến đích một cách tin cậy).

+ Giao thức không hướng kết nối (Connectionless): Trong giao thức này việc phân phối các gói tin dựa vào địa chỉ đích mà không thiết lập kết nối (connectionless) và báo nhận. (tức là nút phân phối gói tin không biết được gói tin phát đi có đến đích hay không).

* Thành phần hoạt động: Bộ định tuyến (router) thiết bị mạng hoạt động tại tầng mạng. Bộ định tuyến tại tầng mạng dựa vào bảng định tuyến (routing table) để lựa chọn đường đi phù hợp cho gói tin nhận được. Các bảng định tuyến có thể tĩnh (static) hoặc động (dynamic).

Tầng 4 (transport - giao vận, vận chuyển):

Trong mô hình tham chiếu thường chia ra 3 tầng cao (Application, Presentation,Session) và 4 tầng thấp (Transport, Network, Data link, Physical). Các tầng thấp quan tâm đến việc đáp ứng việc truyền số liệu giữa các hệ thống cuối (end systems) qua ơhương tiện truyền thông, còn tầng cao tập trung đáp ứng các yêu cầu của ứng dụng và người sử dụng.

Tầng giao vận là tầng cao nhất của nhóm các tần thấp, nó phụ trách việc lưu chuyển thông tin trên mạng một cách "trong suốt" đối với các tầng cao. Do đó các tính năng mức này phải đảm bảo là:

+ Kiểm soát việc truyền số liệu từ nút tới nút (end to end):
  . Đồng nhất một nút bằng một địa chỉ duy nhất.
  . Nhận biết mạng và hoạt động được với cả mạng hướng kết nối (Connection oriented) và cả không hướng kết nối (Connectionless).
  . Phải nhận biết được chất lượng dịch vụ (Quality of Service – QoS) của người sử dụng và khả năng cung cấp dịch vụ của tầng bên dưới.

+ Khắc phục sai sót: Trong quá trình truyền cố liệu có thể xảy ra sai sót thì tầng giao vận phải có khả năng khắc phục các sai sót đó bằng các cơ chế báo nhận và truyền lại các gói tin (packet/diagram).

+ Ghép kênh, cắt/hợp dữ liệu nếu cần: Khối lượng dữ liệu truyền có thể lớn hoặc nhỏ, đường truyền có thể nhiều hoặc ít, Vì vậy tầng giao vận phải có khả năng cắt/ hợp dữ liệu tại đầu gửi/nhận và phân kênh/hợp kênh khi cần thiết (TDM – Time Division Multiplexing, FDM – Frequency Division Multiplexing).

Tầng 5 (Session – Tầng phiên):

Tầng phiên là tầng thấp nhất trong các tầng cao, nó đáp ứng các yêu cầu của người dùng phục vụ cho các ứng dụng trên mạng thông qua các phương tiện truyền thông cung cấp bởi nhóm các tầng thấp.

Tầng phiên cho phép người sử dụng tiếp xúc với nhau qua mạng. Do đó nhiệm vụ chủ yếu là quản lý các cuộc tiếp xúc (giao dịch) giữa người sử dụng trên mạng tạo, duy trì hay hủy bỏ các cuộc tiếp xúc đó. (tầng này chủ yếu là thiết lập giao dịch giữa các ứng dụng trong mạng).

Tầng phiên thiết lập các giao dịch giữa các nút mạng. Một giao dịch phải được thiết lập trước khi truyền dữ liệu trên mạng. Tầng phiên phải đảm bảo cho các giao dịch được thiết lập và duy trì đúng qui định. Khởi tạo cho các giao dịch đầu cuối. Đồng bộ và đồng bộ lại việc truyền thông tin. Điều khiển hội thoại.

Một số dịch vụ của tầng phiên:

- Thiết lập một liên kết với một người sử dụng ở tầng phiên khác, trao đổi dữ liệu với người sử dụng đó một cách đông bộ, và huỷ bỏ liển kết một cách có trật tự khi không dùng đến nữa.
- Thương lượng về việc dùng các thẻ bài (token) để trao đổi dữ liệu, đồng bộ hoá và hủy bỏ liên kết, sắp xếp phương thức trao đổi dữ liệu
  + Half-duplex
  + Full-duplex

- Thiết lập các điểm đồng bộ hoá trong các hội thoại và khi xảy ra sự cố có thể khôi phục lại việc hội thoại bắt đầu từ một điểm đồng bộ hoá đã thoả thuận.
- Ngắt một hội thoại và khôi phục lại hội thoại sau đó từ một điểm xác định trước.

Tầng 6 (presentation - trình bày):

Tầng trình bày chịu trách nhiệm biểu diễn thông tin, chuyển đổi dữ liệu gửi đi trên mạng từ một loại biểu diễn này sang một loại biểu diễn khác. Chẳng hạn như các kỹ thuật nén dữ liệu trước khi truyền đi, sau đó nhận được và khôi phục trở lại.

Tầng trình bày cũng là tầng có thể dùng kỹ thuật mã hoá để xáo trộn dữ liệu trước khi truyền đi và giải mã sau khi nhận lại nhằm đảm bảo việc bảo mật và an toàn số liệu trên đường truyền.

Tầng này cũng đảm bảo cho các hệ thống cuối thuyền thông được với nhau ngay cả khi chúng sử dụng các biểu diễn dữ liệu khác nhau. Để làm được điều đó nó cung cấp một biểu diễn chung để dùng trong truyền thông và cho phép chuyển đổi từ biểu diễn cục bộ sang biểu diễn chung đó.

- Tồn tại 3 dạng cú pháp thông tin được trao đổi giữa các thực thể ứng dụng, đó là:
  + Cú pháp dùng bởi thực thể ứng dụng Nguồn
  + Cú pháp dùng bởi thực thể ứng dụng Đích
  + Cú pháp được dùng giữa các thực thể tầng trình diễn.

* Dịch vụ của tầng Phiên:

Tầng phiên có hai loại dịch vụ cơ bản

- Dịch vụ loại thứ nhất: dùng để biểu diễn dữ liệu của người dùng (dữ liệu cục bộ) thành dạng dữ liệu chung.

  Các nhiệm vụ của loại 1:

  + Thương lượng về cú pháp truyền: với mỗi kiểu dữ liệu người dùng cho trước, một cú pháp truyền được thương lượng.

  + Chuyển đổi: dữ liệu cung cấp bởi người sử dụng được chuyển đổi thành biểu diễn theo cú pháp truyền để truyền đi; ngược lại, dữ liệu nhận được sẽ được chuyển đổi từ biểu diễn theo cú pháp truyền sang biểu diễn của người sử dụng.
- Dịch vụ loại thứ hai: cho phép các thực thể ứng dụng có thể sử dụng các dịch vụ tầng phiên để quản lý hội thoại.

Tầng 7 (application - ứng dụng):

Là tầng cao nhất của mô hình OSI, tầng ứng dụng trên giải quyết các kỹ thuật mà các chương trình ứng dụng có thể trao đổi với mạng.

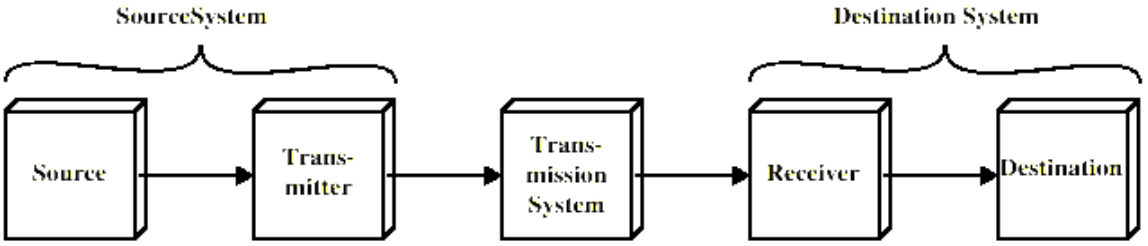
Tầng này phục vụ trực tiếp cho người sử dụng trên mạng với các dịch vụ mạng và phương tiện cần thiết để truy cập môi trường OSI.

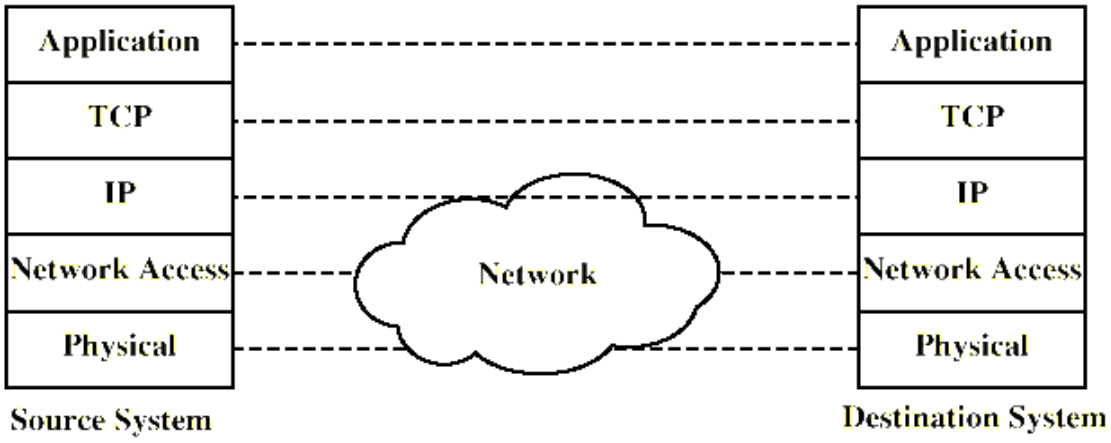
2.3 Mô hình tham chiếu TCP/IP

Bộ giao thức TCP/IP phỏt triển bởi trụ sở nghiờn cứu cỏc dự ỏn cao cấp bộ quốc phũng mỹ (DARPA) cho hệ thống mạng chuyển mạch gúi. Nú được sử dụng bởi mạng Internet toàn cầu (global Internet). Bộ giao thức TCP/IP có thể được xem gồm các tầng sau:

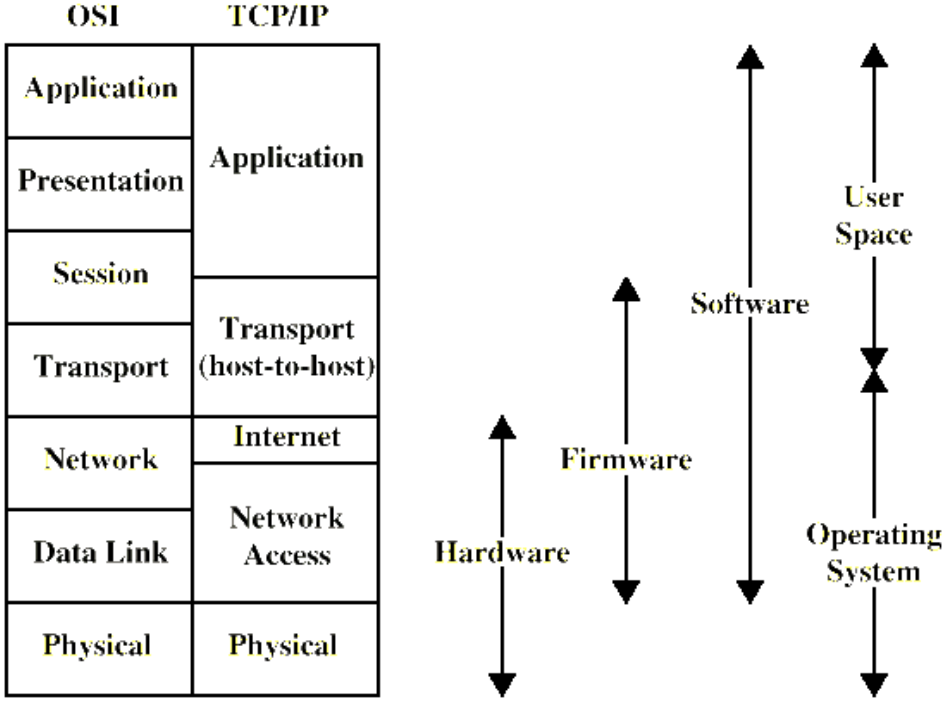
- Tầng ứng dụng (Application layer)
  - Hỗ trợ cho các ứng dụng người sử dụng
  - chẳng hạn như : http, SMTP,…
- Tầng Host to Network hoặc Transport (giao vận)
  - Đảm bảo việc truyền dữ liệu tin cậy
  - Nhận dữ liệu theo đúng thứ tự truyền.
- Tầng Internet (Internet layer)
  - Các hệ thống có thể kết nối được với các mạng khác nhau
  - Các chức năng truyền thông xuyên suốt các mạng đa kênh
  - Được cài đặt ở các end system và router
- Tầng truy cập mạng (Network Access Layer)
  - Trao đổi dữ liệu giữa các hệ thống đầu cuối (end system) và mạng.
  - Kiểm soát địa chỉ đích.
  - Yờu cầu dịch vụ.
- Tầng vật lý (Physical layer)
  - Giao tiếp vật lý giữa các thiết bị truyền dữ liệu và môi trường truyền thông hoặc mạng.
  - Đặc tính của môi trường truyền thông
  - Cỏc mức tớn hiệu
  - Tốc độ truyền dữ liệu
  - ...

Sau đây là mô hỡnh kiến trỳc giao thức TCP/IP.





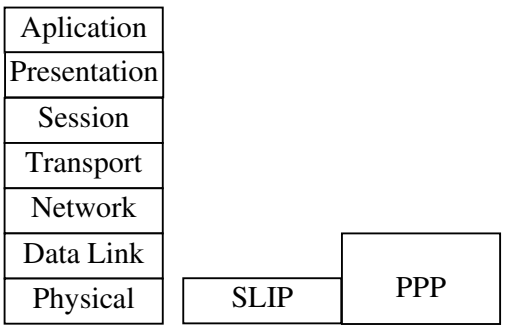
*** So sánh với mô hình tham chiếu OSI:**



2.4 Một số chuẩn mạng

2.4.1 SLIP và PPP

SLIP là một giao thức đầu tiên (RFC 1055) dùng để chuyển tiếp các gói tin (IP Packets) qua đường truyền quay số (Dial-up lines). Đến nay thì SLIP hoàng toàn bị thay thế bởi PPP (Point-to-Point Protocol). PPP là một giao thức phân tầng (a layered protocol) được thiết kế để hỗ trợ việc thiết lập liên kết để truy cập mạng bằng quay số cho nhiều bộ giao thức truyền tải khác nhau. SLIP là giao thức đơn giản hoạt động như ở tầng vật lý, PPP là một giao thức tăng cường hoạt động giống như tầng vật lý và tầng liên kết dữ liệu. Các hệ điều hành của Microsoft hỗ trợ cả SLIP và PPP ở máy trạm nhưng server chỉ hỗ trợ PPP.



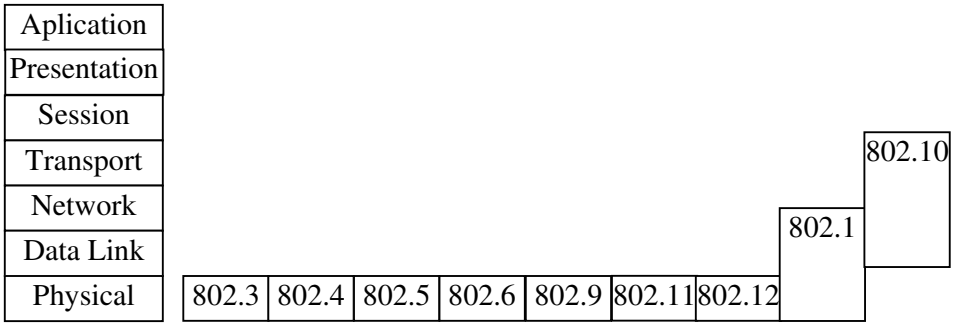
Mối quan hệ giữa SLIP, PPP và mô hình OSI

- SLIP được sử dụng trên các hệ thống cũ, chỉ hỗ trợ giao thức TCP/IP và chỉ hoạt động trên hệ thống sử dụng địa chỉ IP tĩnh.

- PPP hỗ trợ giao thức TCP/IP, NetBeui, IPX, AppleTalk và cả DECNet, hỗ trợ các giao thức hỗn hợp. PPP hoạt động cả trên hệ thống dùng địa chỉ tĩnh (static) và động (DHCP).

2.4.2 Bộ tiêu chuẩn IEEE 802

Họ tiêu chuẩn IEEE do Viện kỹ thuật điện tử đưa ra, những tiêu chuẩn này hướng về việc kết nối giữa card giao diện mạng (NIC) và các phương tiện truyền. Những tiêu chuẩn này đã được thông qua hệ thống ISO, họ tiêu chuẩn này bao gồm:



Mối quan hệ giữa các tiêu chuẩn IEEE 802 và mô hình OSI

IEEE 802.1 (High Level Interface): Tiêu chuẩn này đi với tầng Data Link của mô hình OSI, đây là một chuẩn tổng quát cho quản trị mạng và cung cấp các chuẩn quản trị mạng cho các tiêu chuẩn 802 khác.

IEEE 802.2 (Logical Link Control): Tiêu chuẩn này định nghĩa một lớp phụ LLC được sử dụng bởi các giao thức tầng thấp hơn. Các giao thức tầng mạng có thể được thiết kế độc lập trong cả tầng vật lý và sự thực hiện ở lớp phụ MAC, tiêu chuẩn này ít được sử dụng..

IEEE 802.3 (CSMA/CD): Tiêu chuẩn này được phát triển bởi Digital, Intel, Xerox, tiêu chuẩn này định nghĩa các tính chất có liên quan tới tầng con MAC. Lớp con MAC sử dụng kiểu truy cập tranh chấp Carrier Sense Multiple Access with Collision Dection (CSMA/CD). Kỹ thuật này làm giảm sự tác động của xung đột bằng cách giúp cho mỗi thiết bị trong mạng xác định được nó có đang ở trạng thái tĩnh hay không, một thiết bị sẽ thử truyền đi khi nào mạng đang thụ động. Khi các thiết bị truyền đi, chúng tiếp tục xem xét rằng nó có để xảy ra xung đột hay không. Khi có xung đột xảy ra, tất cả các thiết bị đều ngừng truyền và gửi một tín hiệu "jamming" giúp thông báo tất cả các trạm khác về sự xung đột này. Sau đó mỗi thiết bị chờ một khoảng thời gian ngẫu nhiên trước khi truyền thử lại.

IEEE 802.4 (Token Bus): Tiêu chuẩn này định ra một mạng với các giao thức BUS vật lý mà điều khiển sự truy cập các phương tiện với một cơ cấu dấu hiệu. Tiêu chuẩn này được thiết kế để đáp ứng nhu cầu của hệ thống tự động công nghiệp, tiêu chuẩn này ít được sử dụng.

IEEE 802.5 (Token ring): Tiêu chuẩn này bắt nguồn từ IBM, mạng này dùng cấu trúc ring topology và điều khiển phương tiện dựa theo vòng

IEEE 802.6 (Metropolitan Area Network - MAN): Tiêu chuẩn này định nghĩa một tiêu chuẩn MAN gọi là Distributed Queue Dual-Bus (DQDB), DQDB thích hợp với việc truyền dữ liệu, giọng nói và video. Mạng này dựa vào dây cáp quang trong bus topology hai hàng, khả năng tải trên mỗi bus đều được điều khiển nhiều chiều, tiêu chuẩn này ít được sử dụng.

IEEE 802.7 (Broadband Technical Advisory Group): Tiêu chuẩn này giúp giải quyết những giải pháp về băng tần rộng tích hợp trong môi trường mạng, tiêu chuẩn này hiện vẫn đang được phát triển.

IEEE 802.8 (Fiber Optic Technical Advisory Group): Tiêu chuẩn này giải quyết các phương pháp bổ xung kỹ thuật cáp quang vào trong môi trường mạng.

IEEE 802.9 (Integrated Data and Voice Networks): Tiêu chuẩn này hỗ trợ kênh truyền bất đồng bộ 10 Mbps cùng với kênh 9664 Kbps dành cho dòng dữ liệu riêng biệt. Tiêu chuẩn này được gọi là Isochromous Ethernet (IsoEnet).

IEEE 802.10 (Standards for Interoperable LAN Security): Tiêu chuẩn này giải quyết những vấn đề về tính bảo mật và mã hóa, hiện đang được phát triển.

IEEE 802.11 (Wireless LAN): Là tiêu chuẩn cho mạng cục bộ LAN không dây, Phương thức CSMA/CD đã được phê chuẩn, hiên đang được phát triển.

IEEE 802.14: Tiêu chuẩn này dùng cho việc truyền dữ liệu qua đường dây cáp TV, tiêu chuẩn này vẫn đang được phát triển nhằm tiếp cận Internet qua đường truyền TV.

IEEE 802.3 và IEEE 802.5 media: Tiêu chuẩn này mô tả tính năng và mục đích của các phương tiện sử dụng trong IEEE 802.3 và IEEE 802.5. IEEE 802.2 độc lập về topology, IEEE 802.3 dựa trên ethernet, IEEE 802.5 dựa trên ring topology là những tiêu chuẩn thông dụng nhất của IEEE 802. Tiêu chuẩn IEEE 802.3 mô tả các phương pháp tín hiệu (cả trên băng tần cơ sở và băng tần rộng), tốc độ dữ liệu, các phương tiện và cấu trúc liên kết. Tiêu chuẩn này quy định cụ thể các phương tiện truyền dẫn vật lý như cáp xoắn, cáp đồng trục, cáp quang.

2.4.3 NDIS và ODI

Tiêu chuẩn NDIS (network driver interface specification), được phát triển bởi Microsoft và 3COM, tiêu chuẩn này mô tả giao diện giữa giao thức vận chuyển mạng và trình điều khiển mạng của tầng datalink. NDIS có chức năng cung cấp một ranh giới trung gian giữa **người cung cấp giao thức vận chuyển** và **bộ điều khiển card mạng**, để các nhóm giao thức tương thích NDIS có thể hoạt động với card mạng. NDIS định nghĩa một phương pháp kết hợp các giao thức hỗn hợp thành một trình điều khiển đơn để bộ điều khiển có thể hỗ trợ đồng thời các giao tiếp.

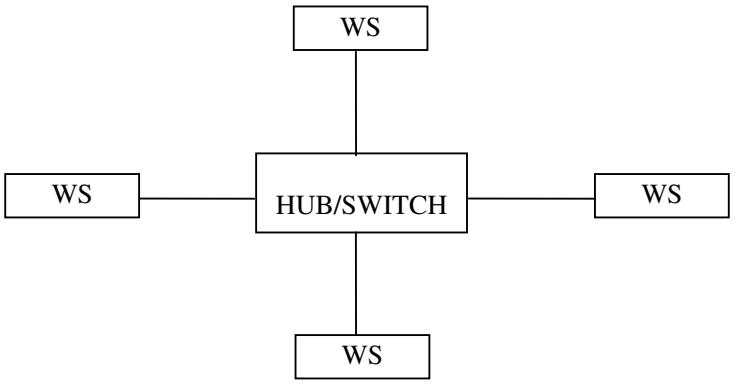
ODI (open datalink interface), phát triển bởi Novell và Apple, hoạt động cùng chức năng như NDIS. ODI cung cấp những qui tắc giúp thiết lập một **giao diện trung gian với người cung cấp, giữa nhóm giao thức và trình điều khiển card mạng**. Giao diện này cũng giúp cho một hay nhiều trình điều khiển mạng có thể hỗ trợ một hay nhiều giao thức.

NDIS và ODI là những tiêu chuẩn giúp cho card mạng hoạt động tốt trong môi trường của hệ điều hành.

§ 3.HÌNH TRẠNG MẠNG (NETWORK TOPOLOGY)

Máy tính và thiết bị xử lý thông tin khác có thể nối với nhau tạo thành một mạng gọi chung là mạng máy tính. Cách nối các máy tính và tập hợp các qui tắc mà tất cả các thực thể tham gia truyền thông trên mạng phải tuân theo để mạng có thể hoạt động được được gọi là kiến trúc mạng. Cách nối các máy tính được gọi là hình trạng (topology) của mạng. Topology của mạng được áp dụng cho mạng cục bộ (LAN) hoặc là mạng diện rộng, trong thực tế chỉ có những mạng máy tính với qui mô nhỏ là áp dụng một topology nào đó, những mạng máy tính lớn thường được tổ chức theo những topology hỗn hợp. Các kiểu topology dưới đây là các kiểu cơ bản.

3.1. Star topology



ở dạng hình sao, tất cả các máy trạm đều nối trực tiếp vào bộ điều khiển trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển tín hiệu đến trạm đích.

Tùy theo yêu cầu truyền thông trong mạng, thiết bị truyền thông có thể là một bộ chuyển mạch (SWITCH), một bộ chọn đường (Router) hoặc chỉ đơn giản là một bộ phân kênh (HUB).

Vai trò của bộ trung tâm này là liên kết điểm-điểm (point-to-point) giữa các máy trong mạng.

- Ưu điểm :
  + Lắp đặt đơn giản.
  + Dễ dàng cấu hình lại (thêm, bớt máy trạm).
  + Dễ dàng kiểm soát và khắc phục sự cố.
  + Tận dụng được tối đã tốc độ của đường truyền vật lý.
- Nhược điểm:
  + Độ dài của đường truyền nối từ một máy trạm tới trung tâm bị hạn chế.
  + Khi bộ điều khiển trung tâm có sự cố sẽ ảnh hưởng đến toàn mạng.
  + Cần nhiều cáp hơn dạng bus

3.2. Bus topology

ở dạng BUS tất cả các máy trạm phân chia chung một đường truyền chính (bus). Đường truyền chính được giới hạn hai đầu bởi một loại đầu nối đặc biệt gọi là **Terminator**. Mỗi trạm được nối vào vào bus qua một đầu nối chư T (T-connector) hoặc một bộ thu phát (Transceiver).

Khi một trạm truyền dữ liệu, tín hiệu được quảng bá (broadcast) trên hai chiều của bus, có nghĩa là mọi trạm còn lại đều có thể nhận tín hiệu trực tiếp.

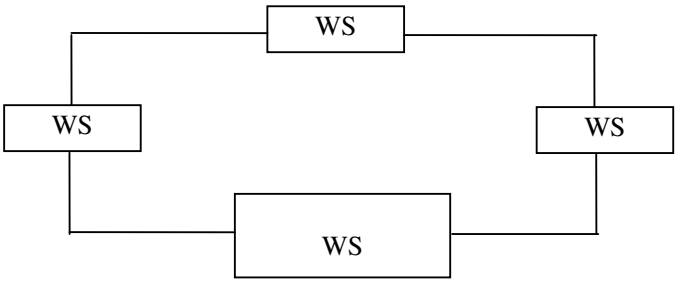
Đối với các bus một chiều thì tín hiệu chỉ đI về một phía, lúc đó terminator phải thiết kế sao cho các tín hiệu phải được "dội lại" trên bus để có thể đến được các trạm còn lại ở phía bên kia.

Như vậy, trong bus topology thì dữ liệu được truyền dựa trên các liên kết điểm-nhiều điểm(point-to-multipoint) hay quảng bá (broadcast).

- Ưu điểm: Lắp đặt đơn giản, sử dụng ít cáp.

- Nhược điểm: Khi có sự cố trên đường truyền thì toàn mạng ngừng hoạt động.

3.3. Ring topology (vòng)



ở dạng vòng mỗi máy trạm được nối với vòng qua một bộ chuyển tiếp (repeater) có nhiệm vụ nhận tín hiệu rồi chuyển đến trạm kế tiếp trên vòng. Như vậy ở dạng này tín hiệu được lưu chuyển trên vòng theo một chiều duy nhất, một chuỗi liên tiếp các liên kết điểm-điểm giữa các repeater.

Giao thức sử dụng cho dạng mạng này là "chuyển thẻ bài " (token passing) để cấp phép truy cập đường truyền.

- Nhược điểm:

+ Đoạn nối giữa hai trạm hỏng sẽ ảnh hưởng đến toàn mạng.

+ Kiến trúc lại mạng khó (thêm, bớt các máy trạm).

+ Giao thức truy cập đường truyền phức tạp.

§ 4. CÁC THIẾT BỊ LIÊN KẾT MẠNG

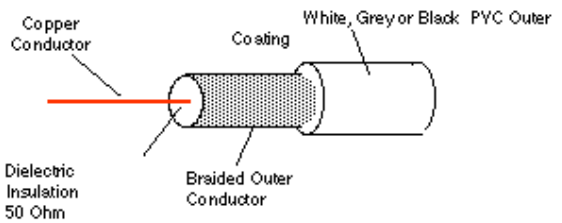
4.1. Dây cáp mạng

Dây cáp đóng vai trò là phương tiện truyền tín hiệu giữa các nút mạng, có nhiều loại cáp nhằm đáp ứng qui mô của nhiều loại mạng khác nhau. Có một số loại cáp thường được như sau.

- Cáp đồng trục (coxial cable):

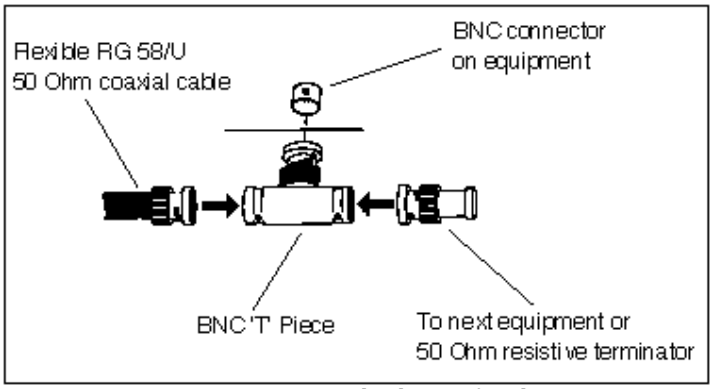
Có độ ảnh hưởng nhiễu thấp, có thể truyền tín hiệu với tốc độ cao trên khoảng cách lớn. Cáp đồng trục có thể dùng cho giải tần cơ sở (Baseband) và giải tần rộng (Broadband).

+ Cáp gầy (thin coxial cable - 10B2 / IEEE 802.3a): trở kháng 50 Ω, có thể đưa tín hiệu đi xa đến 185 mét.

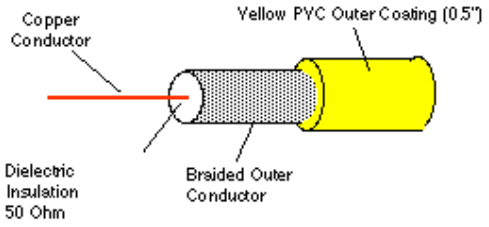


10B2 Cable or "Thin Ethernet"

+ Cáp béo (thick coxial cable - 10B5): có thể đưa tín hiệu đi xa đến 500 mét. Cáp đồng trục sử dụng các bộ nối cáp BNC để tạo kết nối giữa cáp và máy tính, giữa cáp và đoạn cáp khác. Bộ nối gồm có: bộ nối hình chữ T (T-



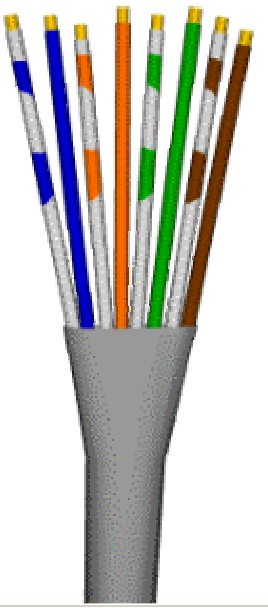
connector) để nối cáp và card mạng; bộ nối ống để nối giữa hai đoạn cáp (BNC-connector) ; bộ nối cuối (Terminator).



10B5 Cable or "Thick Ethernet"

- Cáp không vỏ bọc chống nhiễu (UTP):

Đôi dây cáp điện thoại có thể sử dụng để truyền dữ liệu khi tín hiệu được lọc nhiễu và khoảng cách không lớn lắm. Với loại cáp này mức độ chống nhiễu, khoảng cách truyền, giải tần cũng như số thiết bị gắn vào được xếp ở mức trung bình. Khi truyền ở mức độ cao (1Mbps) nó tạo ra sóng RF, do đó phải sử dụng thêm các bộ lọc cần thiết. Cáp xoắn đôi trần 10BASET có thể đưa tín hiệu đến 100 mét. Cáp xoắn đôi dùng giắc cắm RJ45.

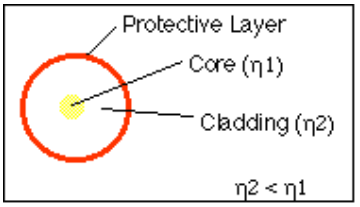
Wire pair #1:	White/Blue Blue	
Wire pair #2:	White/Orange Orange	
Wire pair #3:	White/Green Green	
Wire pair #4:	White/Brown Brown	

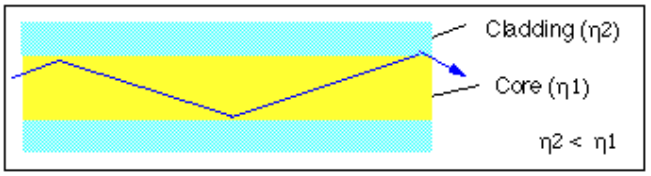
- Cáp có vỏ bọc chống nhiễu (STP)

Là loại cáp có 1 hoặc hai đôi dây nằm trong vỏ bọc kim loại, vỏ bọc giảm nhiễu và giảm phát sinh sóng RF do đó nó cho phép truyền dữ liệu ở tốc độ cao hơn trên khoảng cách lớn hơn loại UTP.

- Cáp quang 10BASEFL, 10BASEFB (công nghệ cao hơn, cho phép truyền tín hiệu đồng bộ)

Trong cáp sợi quang, sợi quang truyền tín hiệu dữ liệu dưới dạng số ở hình thái xung ánh sáng. Cáp này không bị ảnh hưởng nhiễu điện, lý tưởng cho cáp chạy ngoài trời hoặc gần những nguồn điện cao thế . Có khả năng truyền dữ liệu với tốc độ rất lớn (hàng trăm đến hàng nghìn Mbps), là giải pháp tốt cho đường truyền tốc độ cao, làm đường trục (backbone) cho mạng. Cáp quang thường được sử dụng cho giải tần cơ sở.

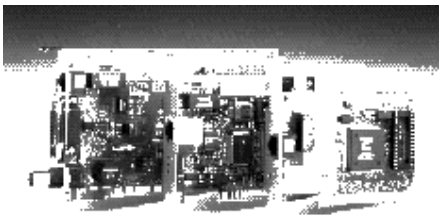




4.2. Vỉ mạch mạng (Network Interface Card - NIC)

Là thiết bị được lắp đặt vào khe mở rộng (expansion slot) của máy tính (có thể được tích hợp trên MainBoard), nó đảm nhiệm truyền dữ liệu từ bus dữ liệu

của một nút (node) (pc, server, printer,...) tới một nút khác trong mạng. Vai trò của NIC là chuẩn bị dữ liệu, gửi dữ liệu đến nút mạng khác, kiểm soát luồng dữ liệu giữa máy tính và hệ thống cáp.



Chuẩn bị dữ liệu: NIC phải chuyển đổi dữ liệu từ dạng thức mà máy tính có thể hiểu được sang dạng thức có thể truyền qua dây cáp mạng. Dữ liệu di chuyển qua một máy tính theo các tuyến gọi là BUS, có thể có nhiều tuyến (8 bits, 16 bits, 32 bits) cùng được truyền dữ liệu dọc theo các tuyến này, gọi là truyền song song. Có một số kiến trúc bus thường dùng như : ISA (16 bít dữ liệu, 32 bit địa chỉ); EISA (32 bít dữ liệu, 16 hoặc 32 bít địa chỉ); PCI (32 hoặc 64 bít dữ liệu, địa chỉ).

Trên cáp mạng, dữ liệu phải đi theo một luồng bít đơn lẻ, dữ liệu được truyền theo một hướng bít này nối đuôi bít kia, gọi là truyền nối tiếp. NIC tiếp nhận tín hiệu chạy song song, sắp xếp lại để có thể truyền nối tiếp theo tuyến rộng một bít của cáp mạng. Việc thực hiện chuyển dịch tín hiệu số của máy tính sang tín hiệu điện và tín hiệu quang do một thiết bị chịu trách nhiệm thi hành gọi là máy thu – phát (Transceiver, transmitter/receiver).

Gửi và kiểm soát dữ liệu: Trước khi NIC ở đầu gửi gửi dữ liệu, nó tiến hành kiểm tra với NIC ở đầu nhận để cả hai cùng thống nhất các tham số:

+ Kích thước tối đa của cụm dữ liệu được gửi

+ Lượng dữ liệu được gửi đi trước khi được xác nhận

+ Thời gian cách quãng giữa những lần gửi dữ liệu

+ Thời gian chờ trước khi tín hiệu báo nhận được gửi đi

+ Mỗi NIC chứa được bao nhiêu dữ liệu

+ Vận tốc truyền dữ liệu

Các tùy chọn và xác lập cấu hình

+ Ngắt (IRQ)

+ Địa chỉ cổng xuất/ nhập (I/O) cơ sở

+ Địa chỉ bộ nhớ

+ Máy thu – phát (qua giắc cắm RJ45, BUS, AUI)

4.3. Bộ phát lặp (Repeater)

Thiết bị trung gian thực hiện chức năng chuyển tiếp ở mức vật lý, nó có tác dụng khuyếch đại tín hiệu trên đường truyền do đó được sử dụng để kéo dài cáp mạng. Nó không thể sử dụng để nối các mạng có công nghệ khác nhau.

Bộ phát lặp hoạt động tại tầng vật lý, nó tiếp nhận tín hiệu từ một đoạn mạng tái tạo và truyền đến đoạn mạng kế tiếp. Muốn chuyển gói dữ liệu qua bộ phát lặp từ đoạn mạng này sang đoạn mạng kế tiếp, gói dữ liệu và giao thức Logical Link Control (LLC) phải giống nhau trên mỗi đoạn mạng. Bộ phát lặp không dịch hoặc lọc bất kỳ tín hiệu nào, để thiết bị này có thể hoạt động, cả hai đoạn mạng nối bộ chuyển tiếp phải có cùng phương pháp truy cập.

4.4. Bộ tập trung (Hub)

Là trung tâm của mạng hình sao (điểm tập trung các đầu dây trong mạng). Hub nhận tín hiệu tại một cổng (port) và lặp lại tín hiệu đó (relay and/or amplify/generate signal) trên tất cả các cổng (port) còn lại. Tại mỗi thời điểm chỉ có một trạm được chuyển dữ liệu. Vì vậy một hub 10Mbps thì có tốc độ tổng cộng là 10Mbps.

8-port 3Com Office Connect Hub

- Hub thụ động (passive hub): chúng đóng vai trò như điểm kết nối và không khuyếch đại (amplify) hay tái tạo (generate) tín hiệu.
- Hub chủ động (Active hub): tái tạo và truyền lại tín hiệu theo cách tương tự cách thức vận hành của bộ chuyển tiếp (Repeater).
- Hub lai (hybrid hub): công nghệ cải tiến chấp nhận nhiều loại cáp khác nhau, có thể mở rộng mạng bằng cách liên kết nhiều hub.

Hub là thiết bị đa năng có nhiều tính năng có thể dùng trong Topology: Star, Star-Bus, Star-Ring. Khi thay đổi hoặc mở rộng hệ thống, đường dây sẽ ít ảnh hưởng đến hoạt động của mạng. Thiết bị Hub có thể để tập trung, tiện cho quá trình bảo dưỡng quản lý mạng.

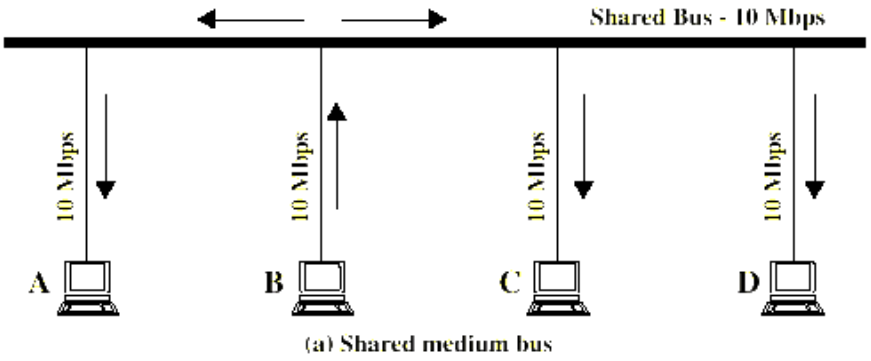
4.5. Chuyển mạch (Switch)

Chuyển mạch được phát triển để thiết kế cho mạng nhiều đoạn, hoặc các mạng máy tính. Chuyển mạch cho phép nối mạng với nhau ở tốc độ cao.

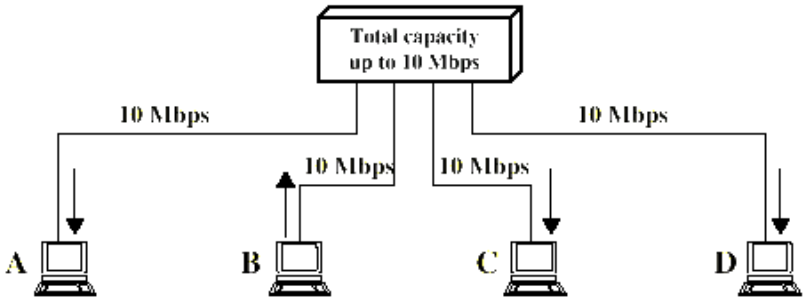
Switch nhận tín hiệu từ một cổng và chuyển tiếp tín hiệu đến cổng kết nối với thiết bị đích. Tại mỗi thời điểm có thể có nhiều hơn một trạm truyền dữ liệu. Vì vậy một switch 10Mbps thì có tốc độ tổng cộng có thể là 10Mbps, 20Mbps, … qua đó ta thấy tốc độ truyền dữ liệu của một switch trên mạng hiệu quả hơn hub (tham khảo hình vẽ mô tả việc truyền dữ liệu sau).

Chuyển mạch Ethernet: được thiết kế để phân chia mạng thành các segment để tăng giải thông cho từng segment. Các chuyển mạch hiện nay có thể sử dụng ở mạng tốc độ cao như Fast Ethernet, FDDI/CDDI hoặc ATM.

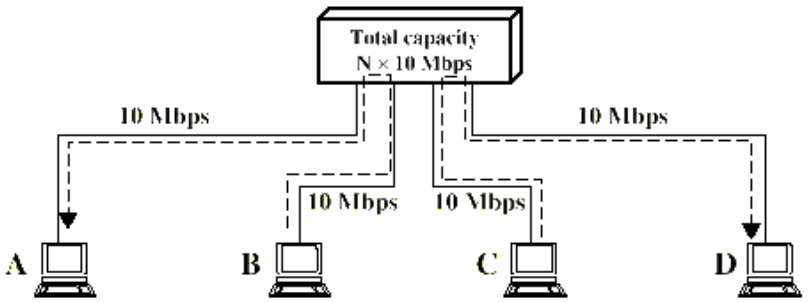
Chuyển mạch cho mạng diện rộng (WAN): được thiết kế để nối các mạng cục bộ thành mạng diện rộng.



(a) Shared medium bus



(b) Shared medium hub



(c) Switching hub

4.6. Cầu nối (Bridge)

Nối các đoạn mạng phân biệt lại với nhau và truyền dữ liệu qua lại giữa chúng. Nó cho phép mở rộng kích thước tối đa của mạng khi gặp giới hạn như chiều dài cáp, giới hạn về số lượng trạm.

Cầu nối hoạt động tại tầng Data Link (thuộc tầng con Media Access Control) nên nó không hiểu được các thông tin ở tầng cao hơn. Do vậy nó không phân biệt giao thức này với giao thức khác, cầu nối có nhiệm vụ chuyển tất cả các

giao thức dọc theo mạng. Khi chuyển dữ liệu qua mạng, tùy thuộc vào từng máy tính quyết định chúng có thể nhận giao thức nào. Cầu nối có nhiệm vụ:

+ Lắng nghe các lưu thông trên mạng

+ Kiểm tra địa chỉ nguồn và địa chỉ đích của mỗi gói dữ liệu

+ Xây dựng bảng định tuyến

+ Gói dữ liệu được chuyển theo cách thức sau: Nếu đích đến không được liệt kê trong bảng định tuyến cầu nối sẽ chuyển dữ liệu đến mọi đoạn mạng. Nếu đích đến được liệt kê, cầu nối sẽ chuyển gói dữ liệu đến đoạn mạng đó. Cầu nối hoạt động trên nguyên tắc mỗi nút mạng có địa chỉ riêng. Một cầu nối chuyển đi các gói dữ liệu dựa trên địa chỉ nút đến.

4.7. Bộ định tuyến (Router)

Bộ định tuyến làm việc tại tầng 3 trong mô hình OSI hay tầng Internet trong mô hình TCP/IP, nó thường được dùng để nối hai hay nhiều đoạn mạng với giao thức và kiến trúc mạng khác nhau. Thiết bị này có chức năng quyết định tuyến đường tốt nhất để truyền dữ liệu và sàng lọc gói tin (Packet filter).

Bộ định tuyến sử dụng bảng định tuyến (routing table) để chứa địa chỉ của các nút mạng, nó sử dụng bảng này để xác định địa chỉ cho dữ liệu đến, bảng này liệt kê các thông tin sau:

+ Toàn bộ số địa chỉ mạng

+ Cách kết nối vào các mạng khác

+ Các lộ trình có thể có giữa các bộ định tuyến

+ Phí tổn truyền dữ liệu qua các lộ trình đó

Các giao thức định tuyến: DECnet, IP, IPX, OSI, XNS, DDP (AppleTalk)

Các giao thức không hỗ trợ định tuyến: LAT (giao thức của hãng Digital Equipment), NetBEUI.

Bộ dẫn đường chia làm hai loại

+ Tĩnh (static): đòi hỏi người quản trị mạng phải cài đặt và lập cấu hình bảng định tuyến đồng thời tự mình định rõ mỗi lộ trình.

+ Động (dynamic): Tự động phát hiện lộ trình và do đó được lập cấu hình ít hơn.

4.8. Cổng kết nối (Getway)

Cổng kết nối cho phép truyền thông giữa các kiến trúc mạng và môi trường khác nhau. Chúng đóng gói lại và biến đổi gói dữ liệu được truyền từ môi trường này đến môi trường khác, sao cho các môi trường có thể hiểu dữ liệu của nhau. Một kết nối liên kết hai hệ thống không sử dụng cùng giao thức truyền thông, cấu trúc định dạng dữ liệu, ngôn ngữ, kiến trúc mạng.

Cổng kết nối chuyên dùng cho tác vụ cụ thể, chúng được dành riêng cho một truyền tải cụ thể nào đó. Ví dụ cổng kết nối giữa Windows NT, Windows 2000 và SNA. Cổng kết nối tiếp nhận dữ liệu từ môi trường, tước bỏ chồng giao thức cũ và đóng gói chồng giao thức của mạng đích.

Một số cổng kết nối sử dụng toàn bộ 7 tầng của mô hình OSI, nhưng cổng kết nối thường thực hiện việc chuyển đổi giao thức tại tầng Application. Trong thực tế mức độ tính năng thay đổi đáng kể giữa các loại cổng giao tiếp.

§ 5. CÁC GIAO THỨC TRUYỀN TRÊN MẠNG (PROTOCOL)

5.1 Khái niệm

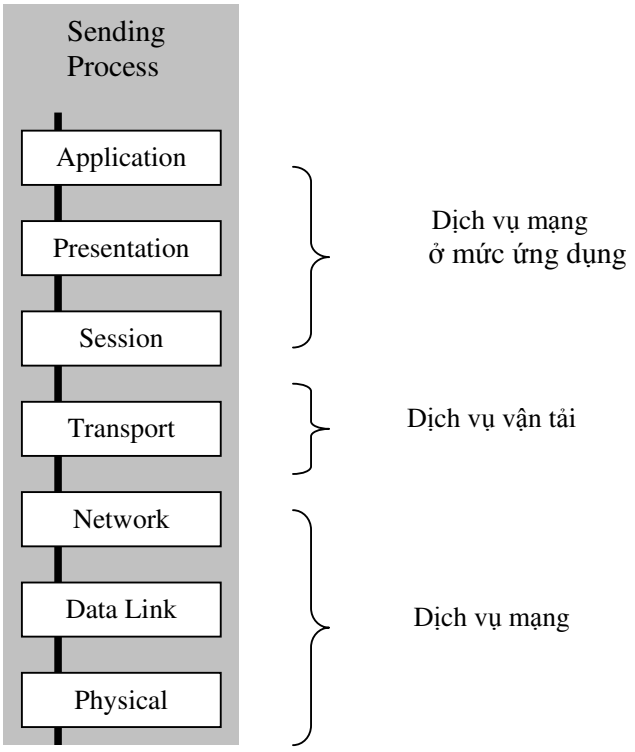
Giao thức là những nguyên tắc và thủ tục điều khiển sự giao tiếp và tương tác khi các máy tính muốn liên hệ với nhau trong môi trường mạng.

Có nhiều giao thức, mỗi giao thức cho phép thực hiện các cuộc giao tiếp cơ bản và thi hành những tác vụ khác nhau.

Một số giao thức hoạt động ở nhiều tầng OSI. Tầng nơi giao thức hoạt động sẽ mô tả chức năng của giao thức đó.

Nhiều giao thức có thể hoạt động phối hợp gọi là chồng giao thức. Cấp độ trong chồng giao thức tương ứng với tầng của mô hình OSI.

Chồng giao thức chuẩn: Giao thức tồn tại ở mỗi tầng của chồng giao thức, làm công việc do tầng đó qui định. Những tác vụ truyền thông cần thi hành qua mạng được gán cho những giao thức đang hoạt động như một trong ba loại giao thức. Ba loại giao thức này ánh xạ đến mô hình OSI



Các tác vụ truyền thông trong mô hình OSI

Giao thức ứng dụng:

- FTAM (File Transfer Access and Management): giao thức truy nhập tập tin của mô hình OSI
- X.400: giao thức CCITT cho việc truyền e-mail quốc tế
- X.5000 giao thức CCITT cho dịch vụ tệp tin và thư mục ngang qua nhiều hệ thống
- SMTP (Simple Mail Transfer Protocol): Giao thức Internet cho việc chuyển mail
- FTP (File Transfer Protocol): Giao thức chuyển tệp tin trên Internet
- SNMP (Simple Network Management Protocol): Giao thức Internet cho việc theo dõi mạng và các thành phần mạng
- Telnet : giao thức Internet cho việc đăng nhập máy chủ ở xa và xử lý dữ liệu trên máy cục bộ
- Microsoft SMP (Server Messaga Block) và shell hoặc bộ đổi hướng (redirector) trên máy khách
- NCP (Novell Netware Core Protocol): và shell hoặc bộ đổi hướng (redirector) trên máy khách của Novell Netware
- Appletalk and Appleshare: Dãy giao thức của Apple
- AFP (Appletalk Filing Protocol): giao thức cho việc truy cập tập tin từ xa của Apple
- DAP (Data Access Protocol): Giao thức truy cập tệp tin Decnet.

Giao thức vận tải:

- TCP (Transmission Control Protocol): giao thức TCP/IP bảo đảm dữ liệu tuần tự
- SPX : một phần của dãy giao thức IPX/SPX (Internetwork Packet Exchange) của Novell Netware
- Nwlink một cài đặt trên IPX/SPX của Microsoft
- NetBEUI (NetBIOS - Network Basic Input/output System) Extended User Interface: thiết lập việc truyền thông giữa các máy tính (Netbios) và cung cấp dịch vụ vận tải dữ liệu (NetBEUI)
- ATP (Appletalk Transaction Protocol), NBP (Name Binding Protocol): Giao thức phiên truyền thông và giao tức vận tải dữ liệu của Apple.

Giao thức mạng

- IP (Internet Protocol)
- IPX: Giao thức của Novell Netware

- Nwlink: một cài đặt trên IPX/SPX của Microsoft
- NetBEUI: Giao thức vận tải cung cấp dịch vụ vận tải dữ liệu cho phiên làm việc và chương trình ứng dụng NetBIOS
- DDP (Datagram Delivery Protocol): Giao thức vận tải dữ liệu của Appletalk

5.2. Các bộ giao thức phổ biến

Trong thực tế các giao thức được dùng phổ biến và được tập hợp thành các giao thức như sau:

1. TCP/IP (transmission Control Protocol / Internet protocol)

- Là giao thức chuẩn cho các hệ trên cơ sở UNIX, đặc biệt sử dụng cho mạng Internet. TCP/IP hỗ trợ việc định tuyến (routing), là giao thức chuẩn cho khả năng liên kết hoạt động của nhiều loại máy tính.
- Mỗi card mạng được gán địa chỉ Internet hay địa chỉ IP
- Địa chỉ IP được phân thành 4 lớp (A B C D)

- TCP/IP là một tập hợp nhiều giao thức nó là tiêu chuẩn thực tế cho liên mạng:

+ FTP (File Transfer Protocol) một ứng dụng để truyền file giữa các máy
+ SMTP (Simple Mail Transfer Protocol) hệ thống thư điện tử
+ SNMP: quản lý mạng

2. IPX/SPX

Giao thức trao đổi thông tin trên mạng của hãng Novell. IPX sử dụng phương thức truyền gói dữ liệu và nguyên tắc tìm đường theo chuẩn IEEE 802.3 , hỗ trợ định tuyến

3. NetBIOS (Network basic input/output system)

Giao thức này được sử dụng trên mạng với các ứng dụng của IBM, Microsoft, và Lotus.

- Chuẩn của IBM về giao diện chương trình cho mạng máy tính
- Cơ sở là ngang hàng
- Nó là giao thức định hướng cho việc nối mạng, không có mức mạng
- Không có giao thức tìm đường
- Kiểm tra được luồng thông tin

Quy tắc của NetBIOS không quy định chặt chẽ về phần cứng, phần mềm, giao thức hay đường truyền vật lý mà mạng sử dụng.

NetBEUI (NetBIOS Extended User Interface): Giao thức kèm theo sản phẩm mạng của Microsoft.

§ 6. THIẾT KẾ MẠNG CỤC BỘ

6.1. Qui tắc thiết kế mạng:

Khoảng cách tối đa cho từng đoạn mạng tùy theo kiểu đường truyền:

- 10Base5: 500 mét cho đoạn cáp không có bộ phát lặp
- 10Base2: 185 mét cho đoạn cáp không có bộ phát lặp
- 10BaseT: 100 mét
- 10BaseFL, 10BaseFB: 2000 mét

Trong thực tế, khoảng cách tối đa còn phụ thuộc vào môi trường lắp đặt, ví dụ như điều kiện thời tiết... có thể làm giảm khoảng cách thực tế. Hoặc việc sử dụng các thiết bị khác như HUB thông minh có thể làm tăng khoảng cách thực tế.

 Xác định số các bộ phát lặp tối đa giữa hai trạm trên mạng:

- Theo Ethernet: một cặp bộ phát lặp được đếm là một bộ lặp, chỉ có 2 bộ lặp trên đường truyền giữa hai trạm bất kỳ.
- Theo IEEE: không lớn hơn 5 đoạn phát lặp, không quá 4 bộ phát lặp giữa hai trạm bất kỳ, trong đó 3 đoạn có thể nối máy trạm (5 segments, 4 repeaters, 3 populated segments)

 Số trạm tối đa: Tùy theo kiểu đấu nối mà có được số workstations tối đa

- 10Base5: 100 workstations
- 10Base2: 30 workstations
- 10BaseT: 2 workstations
- 10BaseFL, 10BaseFB: 2 workstations

Mạng nhiều đoạn (Multisegment):

- Đoạn mạng là phần dây cáp được chặn bởi các thiết bị như cầu, bộ dọn đường, bộ phát lặp, terminator. Tác dụng của 1 đoạn và nhiều đoạn

- 1 đoạn mạng:

+ Cấu trúc đơn giản

+ Tốc độ bảo đảm vì không bị trễ (do các thiết bị nối đoạn mạng)

+ Giá thành cho một trạm thấp

+ Lưu lượng thông tin trên mạng lớn, sẽ dẫn đến tình trạng quá tải, ảnh hưởng đến toàn bộ mạng.

+ Khó khăn khi mở rộng mạng.

- Nhiều đoạn mạng:

+ Tăng kích thước mạng và số các thiết bị gắn vào mạng.

+ Tính sẵn sàng của mạng cao: các mạng con được tách biệt cả về logic và vật lý, do vậy lỗi từ đoạn này không làm ảnh hưởng đến đoạn khác.

+ Cung cấp các khả năng kết nối giữa các trạm nằm trên các đoạn mạng khác nhau.

+ Dễ dàng mở rộng mạng

Có thể sử dụng cầu (Bridges) hoặc chuyển mạch (Swiches) để nối các đoạn mạng.

6.2. Phương pháp thiết kế :

Thiết kế mạng là chọn ra một cấu hình đúng cho một phương án cụ thể, phải thỏa mãn được mục đích và nhu cầu của người sử dụng, đồng thời thỏa mãn các ràng buộc của hệ thống.

Thu thập các nhu cầu, các thông tin về mạng:

Số lượng người dùng của từng ứng dụng, kiểu ứng dụng. Các thông tin này sẽ được xem xét khi lựa chọn thiết kế phân đoạn mạng, cấu hình của các thiết bị.

Xác định nhu cầu về truyền tin của từng trạm, mối quan hệ giữa các trạm. Khi đó xác định những nhóm người sử dụng.

Xem xét khoảng cách vật lý giữa các trạm trong mạng, 1 tòa nhà hay nhiều. Tùy thuộc vào khoảng cách thực tế mà quyết định phân đoạn mạng bằng cầu, chuyển mạch.

Các thông tin phục vụ cho bản vẽ chi tiết:

+ Số lượng các trạm, vị trí ổ cắm.
+ Vị trí, kích thước của tủ đấu dây
+ Độ dài các đoạn cáp giữa các thiết bị
+ Trạng thái nhà cửa, văn phòng

Sự phát triển của mạng trong tương lai.

 Thiết kế chi tiết:

- Bản vẽ chi tiết về cách mắc dây của mạng: kiến trúc mạng, cấu hình các thiết bị trên mạng.
    + Thiết kế tổng thể.
    + Thiết kế cụ thể từng đoạn mạng, khu vực thiết bị để tập trung.
- Các phần mềm sử dụng trên mạng: hệ điều hành, cơ sở dữ liệu, các ứng dụng...

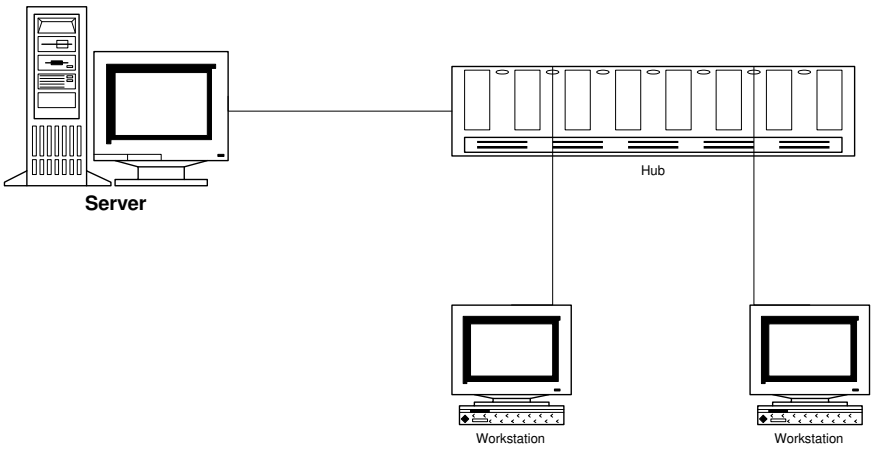
 Một số yêu cầu khác

- Qui định về quản lý mạng:
    + Người quản lý mạng: số lượng, vị trí.
    + Các thủ tục phải thực hiện hàng ngày
    + Thủ tục phát hiện và khắc phục sự cố trên mạng
    + Thủ tục lưu trữ dự phòng và khôi phục
    + Theo dõi sử dụng mạng
    + Hướng dẫn người sử dụng tại chỗ
    + Phân phối phần mềm

6.3. Một số thiết kế mạng

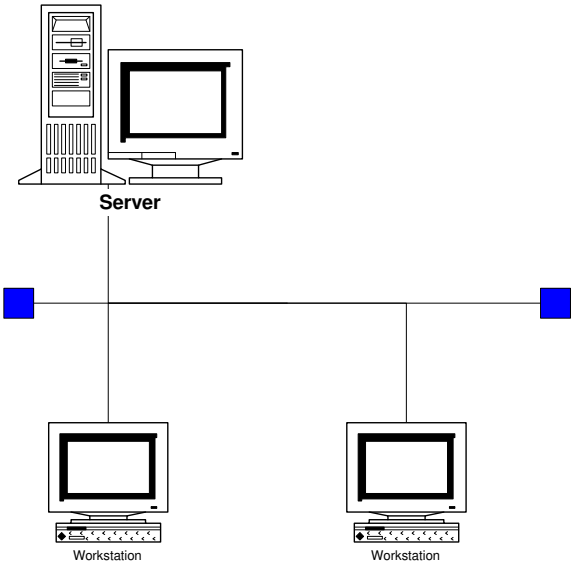
Star Topology : SERVER được nối trực tiếp vào HUB bằng cáp UTP:

Dùng trong mạng nhỏ, bố trí máy tập trung, số lượng trạm phụ thuộc vào số cổng của HUB, có loại Hub 4, 8 ,12, 24 port, ... tùy số lượng máy mà lắp đặt hub phù hợp.



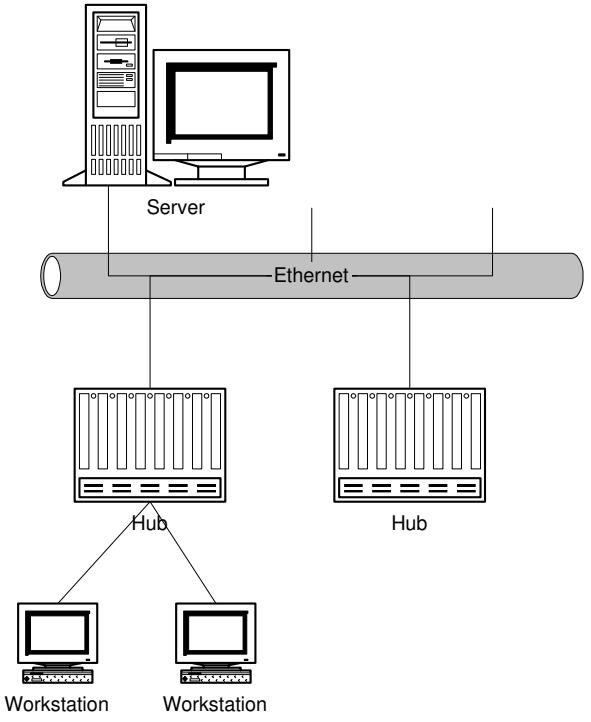
Topology Bus: SERVER nối trực tiếp với các máy bằng cáp BNC

Dùng cho mạng nhỏ, số lượng máy không lớn, thường được dùng cho một văn phòng hoặc một mạng máy tính nhỏ nhưng các máy tính cách nhau khá xa, trong phạm vi cho phép của cáp đồng trục.



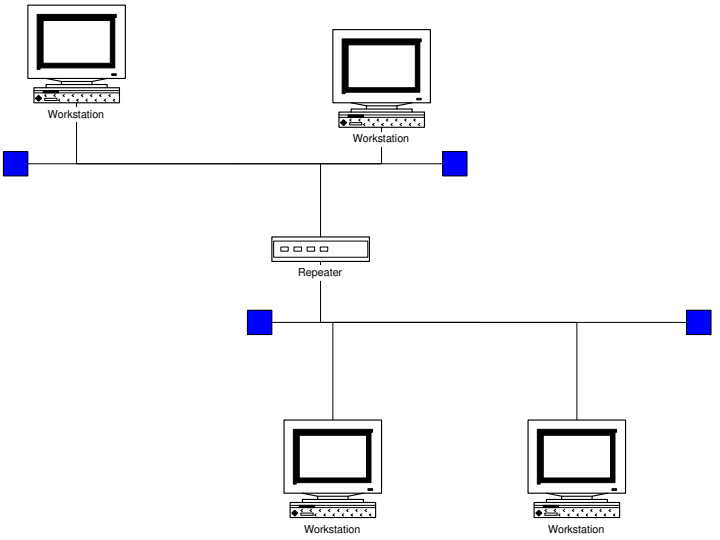
Topology Bus – Star: SERVER nối với HUB - HUB bằng cáp BNC

Dùng cho mạng với qui mô khá lớn, có thể có nhiều phòng máy, các phòng máy lại cách nhau xa (trong phạm vi cho phép của cáp đồng trục), các HUB có thể để ở các phòng khác nhau, mỗi hub có thể cung cấp đường truyền cho 1 hoặc 2 phòng tùy theo số lượng máy.



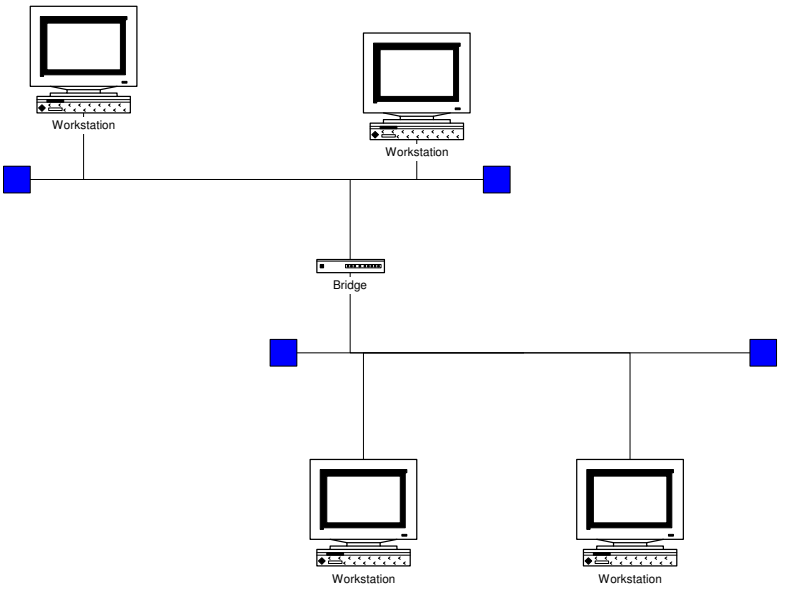
Liên kết hai đoạn mạng bằng Repeater:

Đoạn mạng có thể là 1 tầng nhà hoặc tòa nhà, số lượng máy của mỗi đoạn mạng không lớn lắm, máy trạm để tập trung.



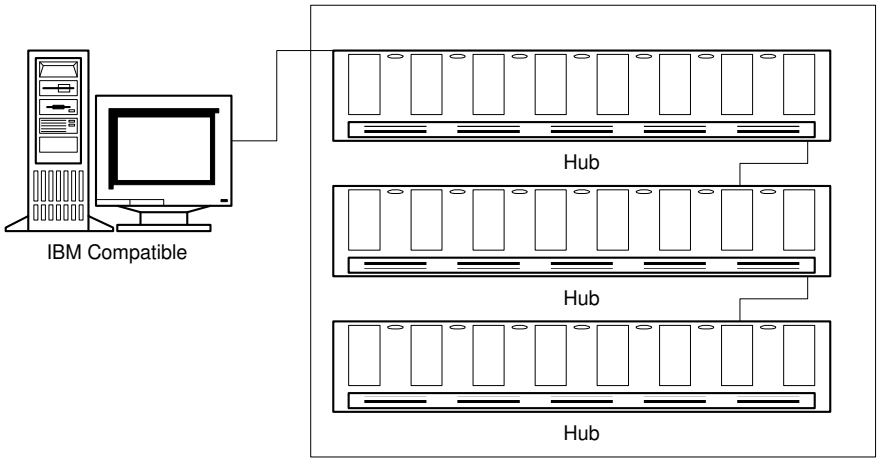
Liên kết hai đoạn mạng bằng Bridge:

Đoạn mạng có cấu trúc mạng khác nhau, mỗi đoạn mạng có thể lắp đặt ở 1 tầng nhà hoặc tòa nhà, mỗi đoạn mạng có thể có số lượng máy khá lớn.



server nối vào các HUB xếp chồng

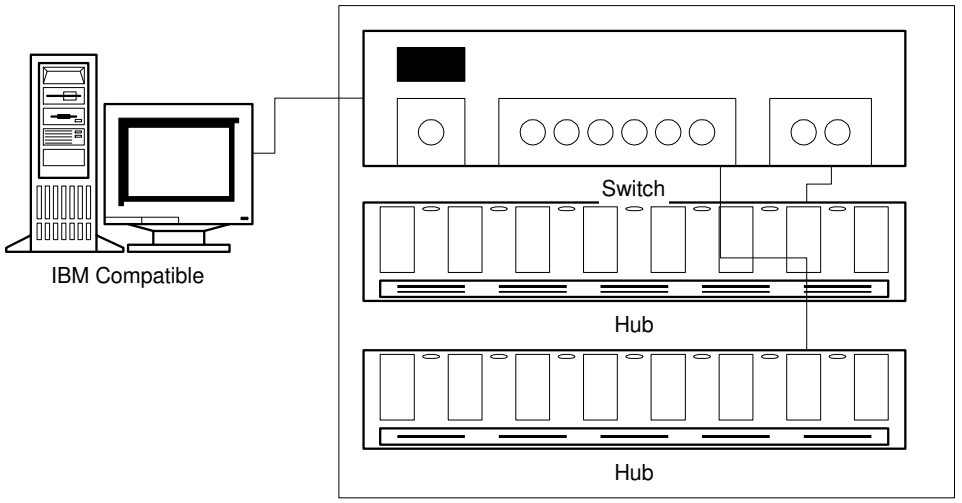
Số lượng máy trạm khá lớn, thường là các máy trạm được để tập trung, khoảng cách máy trong cự ly cho phép, các HUB quản lý tập trung. Tuy nhiên mạng này dùng trong các tổ chức mà lưu lượng thông tin trên mạng không lớn lắm, nếu quá trình truyền tin trên mạng diễn ra liên tục thì người ta thường thiết kế mạng theo mô hình Lan-Switch



Các HUB liên kết qua Switch:

- Switch có nhiệm vụ kết nối các HUB, khi đó Switch có nhiệm vụ quản lý tất cả các đường tín hiệu trên mạng, với cách lắp đặt này có thể cải thiện tốc độ

mạng. Đây là mô hình có thể dùng cho các tổ chức có qui mô máy lớn, các máy trạm đều nằm trong phạm vi nhỏ hơn 150 m so với tủ HUB trung tâm. Theo mô hình này khi cần mở rộng mạng chỉ cần lắp đặt thêm HUB, tuy nhiên số lượng HUB cực đại phụ thuộc vào số cổng ra của Switch. Vì các thiết bị này đặt tập trung nên có thể thiết kế bộ nguồn điện nuôi HUB tập trung, hạn chế sự cố về điện. Tất cả các dây mạng đều tập trung về tủ điều khiển nên cũng dễ dàng xử lý sự cố.



PHẦN II

HỆ ĐIỀU HÀNH MẠNG NOVEL NETWARE

§ 1. GIỚI THIỆU MẠNG NOVELL NETWARE

1.1. Sự phát triển của Novell Netware

NOVELL NETWARE được thiết lập năm 1983 với cấu hình tối thiểu và một số dịch vụ đơn giản.

Novell phiên bản 1.X , 2.X, 3.XX được phát hành cho đến năm 1993

Novell phiên bản 4.X được cài đặt quản trị mạng trên hệ thống nhiều FILE SERVER thông qua hệ dịch vụ NDS (Netware Directory Servise) mạng này có thể quản lý trên hệ thống rộng lớn.

Novell 5.X ra đời vào năm 1999, phiên bản này được thiết kế với nhiều tính năng mới, giúp người dùng dễ dàng sử dụng hơn, dễ tích hợp với mạng diện rộng và Internet.

Tài liệu này đề cập đến những nội dung cơ bản của Novell Netware và các chức năng của phiên bản Novell 4.X.

NOVELL NETWARE được dùng rộng rãi trong:

- Các văn phòng: liên kết các bộ phận rời rạc
- Các công ty: liên kết các xí nghiệp thành viên

Novell Netware có thể chạy trên những máy có cấu hình không cao (cả SERVER và WORKSTATION), cài đặt không phức tạp, quản trị mạng đơn giản và có hiệu quả.

Quản trị Novell Netware bao gồm những công việc như sau:

- Quản trị FILE SYSTEM
- Quản trị USER và GROUP
- Quản trị FILESERVER
- Quản trị PRINT SERVICE
- Quản trị các dịch vụ khác

1.2. Novell Directory Services

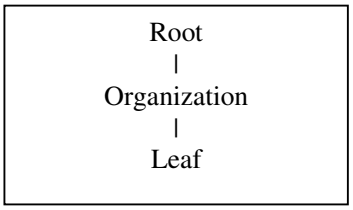
a. Khái niệm

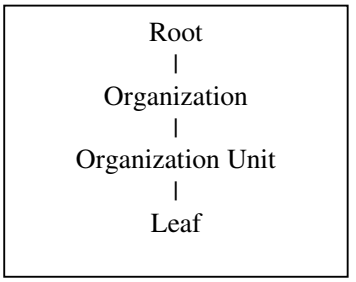
Novell Directory Services (NDS) là dịch vụ đặc trưng của Novell Netware, NDS duy trì cơ sở dữ liệu của tất cả tài nguyên mạng. NDS tạo các mạng thành một mạng thống nhất bằng cách cung cấp một điểm truy nhập và quản lý hầu hết tài nguyên mạng. NDS quản lý tài nguyên theo cấu trúc hình cây, mỗi cây (tree) được đặt tên, nó có thể quản lý nhiều đối tượng (object), có các loại đối tượng sau:

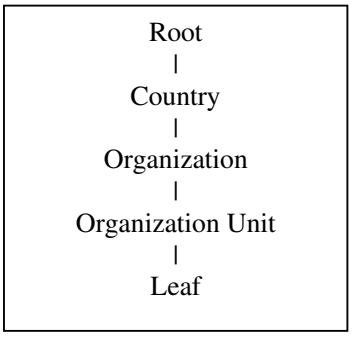
- Đối tượng Root biểu thị cấp cao nhất trong cây thư mục, Root không chứa thông tin, không thể xóa hay đổi tên.
- Đối tượng Container được dùng để tượng trưng cho quốc gia, công ty, phòng ban, nhóm làm việc và tài nguyên dùng chung (Country,

Organization, Organization Unit). Trong đối tượng này có thể chứa các container khác theo 3 mức.

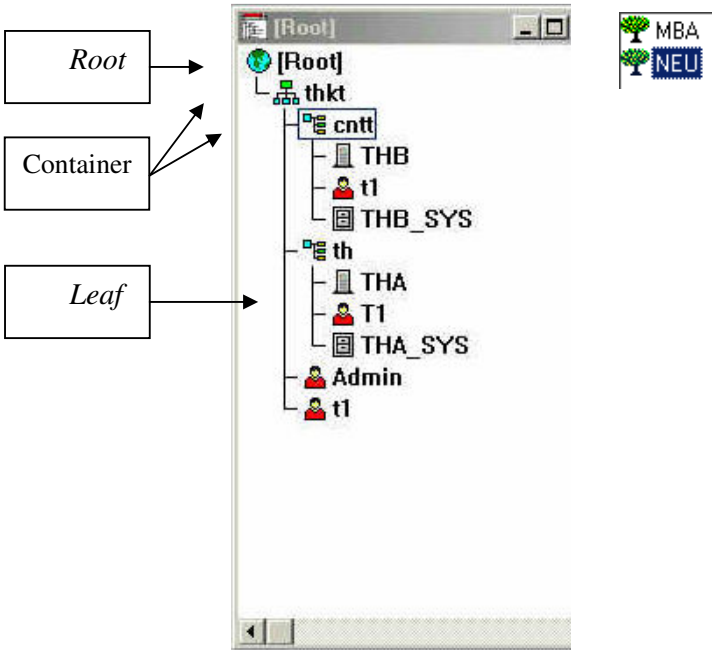
- Đối tượng Leaf đại diện cho tài nguyên mạng như người dùng, máy in, ...





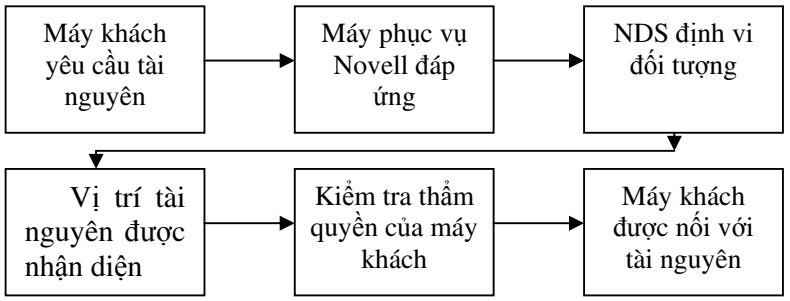


Cấu trúc NDS của tree NEU có Root và container THKT (O), trong đó chứa container CNTT và TH (OU), trong container CNTT chứa server THB user t1 (Leaf), trong container chứa server THB, ... (CN).



b. Vai trò của NDS

Để truy cập tài nguyên của mạng, người sử dụng phải yêu cầu đối tượng theo tên của NDS. Khi NDS trên máy phục vụ nhận biết yêu cầu, nó sẽ kiểm tra trên cây NDS về người sử dụng, quyền sử dụng đối tượng, dựa trên yêu cầu và thẩm quyền người dùng máy dịch vụ sẽ định vị và nối người sử dụng với tài nguyên.



c. Tên của đối tượng

Các đối tượng có thể trùng tên, nằm trên các container khác nhau, NDS không tìm kiếm đối tượng trên toàn cây thư mục, do vậy NDS đòi hỏi thông tin chính xác để tìm đúng đối tượng. Khi truy cập phải cung cấp cho NDS đúng tên đối tượng, có thể cung cấp thông tin bằng: tên phân biệt hay tên phân biệt tương đối.

- Tên chung (common name) của đối tượng leaf là tên nằm bên cạnh đối tượng leaf trong cây thư mục. Ví dụ theo cây NEU có CN=T1, THA, ...

- Ngữ cảnh (contex) là vị trí của đối tượng trong cây thư mục. Đó là danh sách đối tương container bắt đầu từ đối tượng đến root. Ví dụ theo cây NEU có THKT, CNTT,...
- Tên phân biệt: sự kết hợp tên chung và ngữ cảnh. Ví dụ theo cây NEU ta có .CN=T1.OU=CNTT.O=THKT
- Tên phân biệt tương đối (relative distinguished name) liệt kê đường dẫn đối tượng từ đầu đối tương được đặt tên đến ngữ cảnh hiện hành. Tên phân biệt tương đối + ngữ cảnh hiện hành = tên phân biệt.
- Tên có kiểu (Type name): dùng chữ viết tắt để phân biệt các kiểu container và các đối tương leaf
- Cách gọi tên không có kiểu(Typeless name): không bao gồm các kiểu thuộc tính đối tượng. Ví dụ: .T1.CNTT.THKT. Nếu không cung cấp tên đối tượng có kiểu, NDS sẽ tính toán kiểu thuộc tính cho từng đối tượng.

§ 2. QUẢN TRỊ FILE SERVER

- Thực hiện các tiện ích cơ bản của FILESERVER bằng cách nạp các MODUL chính như INSTALL, MONITOR , VREPAIR , REMOTE
- Cài đặt và bổ sung cấu hình mạng (INSTALL)
- Gửi thông điệp hệ thống hoặc cho từng USER (MONITOR)
- Kiểm tra liên kết hoặc ngắt bỏ liên hệ của các WORK STATION
- Sửa lỗi của VOLUME trong một số trường hợp (VREPAIR)
- Cài đặt hay hủy bỏ các dịch vụ, phần mềm hệ thống khác.
- Đóng FILESERVER

 Cách nạp các MODULE trên máy SERVER:

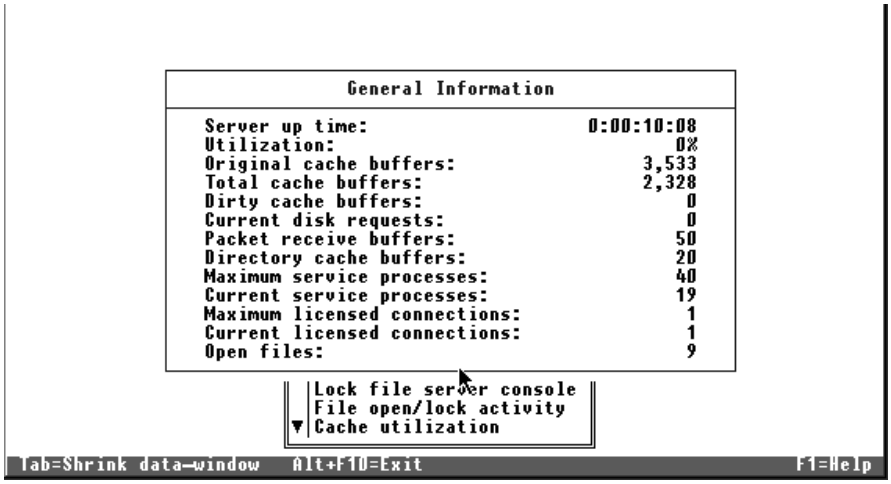
Tại dấu nhắc Novell Netware bấm LOAD <Tên mudul>
Các module sau đây thường được thực hiện:

1. INSTALL

Cài đặt cấu hình hệ thống mạng,tạo các NETWARE PARTITIONS dùng làm các VOLUMES trong mạng. Khởi tạo các VOLUMES, sao chép các FILE hệ thống, cài đặt các phần mềm mạng.

2. MONITOR

Kiểm tra hoạt động của hệ thống, các liên kết trong hệ thống. Ví dụ sau khi load module monitor trên màn hình sẽ hiển thị các thông tin



3. PSERVER

Cài đặt và kiểm soát các hoạt động của PRINT SERVER được cài đặt trực tiếp vào FILESERVER . Điều khiển các máy in trong mạng.

4. VREPAIR

Sửa lỗi khi khởi động mạng. Các lỗi này thường xảy ra khi hệ thống bị tác động đột ngột từ bên ngoài, ví dụ mất điện đột ngột.

```
Options:

     1. Repair a volume
     2. Set VRepair options
     3. Exit

     Enter your choice:
```

Chọn các chức năng để thực hiện chương trình

Đ3. CÀI ĐẶT MẠNG NOVELL NETWARE

3.1. Yêu cầu về phần cứng - phần mềm

- 1 FILE SERVER (máy chủ) : với cấu hình tối thiểu CPU 486 4MB RAM
- Các Work station (máy làm việc)
- Card mạng (ne2000, 3c503 ....)
- Các phụ kiện khác
  + Dây cáp: cáp quang, cáp đồng trục,cáp điện thoại
  + Các thiết bị nối chúng: nối đường thẳng, nối chữ T ...
  + Hub (thiết bị trung gian nối các work station vào FILE SERVER)
- Phần mềm

+ 1 bộ đĩa NOVELL NETWARE (hoặc CD ROM)
+ 1 bộ đĩa DOS
+ Các chương trình sẽ chạy trên mạng

3.2. Các bước cài đặt:

1. Khởi tạo FILESERVER

- Lắp đặt Card mạng vào slot của main-board, lắp dây mạng. Kiểm tra các thiết bị bảo đảm cho máy có thể hoạt động được.
- Khởi động máy chủ, dùng FDISK tạo trên đĩa cứng của máy chủ một DOS PARTITION FORMAT và nạp phần hệ thống

2. Khởi động lại máy chủ

- Chạy tệp install.bat (Trên bộ đĩa cài đặt)
- Hoặc tạo 1 thư mục, chuyển vào thư mục ; sao chép các tệp sau đây:
  + SERVER.EXE
  + Trình điều khiển đĩa cứng (IDE.DSK, ISADISK.DSK, SCSI???.DSK ...)
  + Trình điều khiển CARD mạng (*.LAN)
  + Các tệp khác (*.NLM)
- Bấm SERVER trên màn hình hiển thị dấu nhắc của NOVELL
- Máy thông báo : FILESERVER NAME - Gõ tên của FILE SERVER (tên do mình đặt, nên đặt tên ngắn gọn)
- Máy thông báo : IPX INTERNAL NETWORK NUMBER - Gõ dãy số đặc trưng của card mạng

Thực hiện điền các tham số của Modul Install

- Tại dấu nhắc của Server Novell Netware gõ LOAD INSTALL, trên màn hình hiển thị thực đơn của module install:

Installation Options	
Driver options	(Load/Unload disk and networkdrive)
Disk options	(configure/mirror/test disk partitions)
Volume options	(configure/mount/dismount volumes)
Licence option	(install the server licence)
Copy files option	(install Netware system files)
Directory options	(install Netware Directory services)
NCF files options	(create/edit server startup files)
Product options	(other optional installation items)

a- Chọn thực đơn DRIVER OPTIONS

Cài đặt trình điều khiển đĩa cứng và card mạng

Driver options
Configure disk and storage device Configure network drivers Return to previous menu

- Đĩa cứng: chọn "Configure disk and storage device drivers" , chọn load a driver

Additional Driver Actions
Load a driver Unload a selected driver Return to previous menu

Trên màn hình hiển thị danh mục các driver, chọn loại phù hợp

Select a driver:
IDE.DSK IDE (ATA Compatible) . . . .

Điền các tham số cho đĩa cứng (I/O: 1f0,170,1e8,168; interrupt: A,B,C,E,F)

IDE Parameters
Interrupt number: E Port value: 1F0 Scatter Gather: No Driver Version: version 5.00 (940930)

Driver IDE parameter Actions
Select/Modifi driver parameter Save parameters and load driver

- Card mạng : chọn “Configure network drivers ” , chọn load a driver

Select a driver to install
3C503.LAN 3Com 3c503 EtherLink NE2000.LAN Novell Ethernet NE2000 .....

- Chọn các tham số cho card mạng:

NE2000_1 Protocols ----- IPX (always seleted) [ ] TCP/IP [ ] AppleTalk

Board NE2000_1 (Driver NE2000) Actions
Select/Modifi driver parameters and Save parameters and load driver

IPX là giao thức ngầm định, Novell Netware luôn sử dụng giao thức này, có thể chọn thêm các giao thức TCP/IP và Appletalk.

b. Chọn thực đơn “DISK OPTIONS”

Available Disk Options
Modifi disk partitions and hot fix Mirror/Unmirror disk partitions Perform surface test (optional) Scan for additional devices (optional) Return to the previous menu

- Chọn “Modifi disk partitions and hot fix”

Trên màn hình hiển thị bảng phân chia đĩa: partition thứ nhất do DOS quản lý, cần phải khởi tạo Volumes cho partition thứ hai.

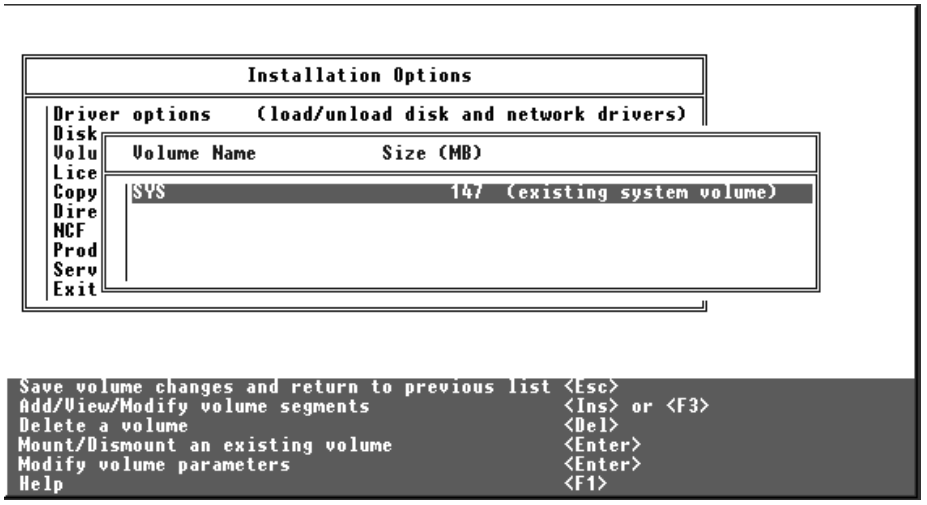
Disk Partition Type	Start	End	Size
Unknown Partition Type 6		0	627 250.2 MB
Netware Partition 628	1008	800.0 MB	

c. Chọn thực đơn "VOLUME OPTIONS "

Khi chưa có volume nào được khởi tạo

Volume Name	Size (MB)

Chọn các phím tương ứng để tạo Volume mới



Các thông tin về partition của đĩa được hiển thị

Volume Disk Segment List				
Device No.		Segment No.	Size(MB)	Volume Assigment Status
0	0	800	(free space)	

Volume đầu tiên được Novell đặt tên là SYS, dung lượng tối thiểu là 75MB.

Disk segment parameters
Disk segment volume name: SYS Disk segment size: 500 MB

Nếu chọn dung lượng của volume SYS cực đại thì sẽ không có volume tiếp theo

Tiếp tục khởi tạo các volume

What would you like to do with this free segment
Make this segment a new volume Make this segment part of another volume

d. Chọn thực đơn "LICENCE OPTIONS"

Cài đặt bản quyền cho Novell Netware thường được cài đặt từ đĩa mềm (đĩa bản quyền), bản quyền quy định số user tối đa được nối vào mạng

e. Chọn " COPY FILES OPTIONS"

Lựa chọn cài đặt files hệ thống, nhóm trình nào được đánh dấu [x] sẽ được cài đặt

Indicate which file groups you want installed
[x] NetWare 4.1 Server executable and boot files (4MB) [x] Update NetWare 4.1 boot directory driver files (1MB) [x] Pre-Install Files (7MB) [x] NetWare System Files (9) [x] NetWare DOS Utilities (12MB) [x] NetWare MS OS/2 Utilities (2MB) [x] NetWare MS Windows Utilities (4MB) [x] NetWare UNIX Utilities (1MB) [x] ETC Files (1MB) [x] Set up a Network Directory for Client Install (7MB) [x] Set up a Network Directory for Server Migration (2MB) [x] NetWare 4.1 English Language-Specific Files (5MB)

f. Chọn thực đơn "DIRECTORY OPTIONS"

Cài đặt NDS và các cập nhật: khi cài phải lựa chọn
Directory name:
Company organization:
+ Level 1:
+ Level 2:
+ Level 3:
- Mật khẩu của ADMIN

Directory Services Options
Install Directory Services onto this server Remove Directory Services from this server Upgrade NetWare 3.x bindery information to the Directory Upgrade mounted volumes into the Directory Return to the previous menu

g. Chọn thực đơn "NCF FILES OPTIONS"

Available NCF Files Options
Create AUTOEXEC.NCF file Create STARTUP.NCF file Edit AUTOEXEC.NCF file Edit STARTUP.NCF file Upgrade a v3.1x AUTOEXEC.NCF Files Return to the previous menu

3.3. Cài đặt WORK STATIONS (DOS)

- Yêu cầu về phần cứng: Máy phải được lắp card mạng, nối dây vào mạng
- Yêu cầu về phần mềm: máy đã được cài đặt hệ điều hành DOS, bộ đĩa cài đặt card mạng.

a. Cài đặt tối thiểu: truy cập mạngmức bindery

Khởi tạo tệp tin IPX.COM

- Tạo 1 thư mục tên là WSGEN và sao chép toàn bộ đĩa WSGEN của Novell Netware vào thư mục này.
- Chạy trình cài đặt WSGEN, chọn danh sách card mạng ứng với card mạng đã cài vào WORK STATIONS, điền các tham số (IRQ, I/O, DMA or RAM ...)

Nếu cài đặt thành công tệp tin IPX.COM sẽ được tạo ra

Nối WORK STATIONS vào mạng: khởi động máy ở DOS, bấm các lệnh sau

IPX
NETX
Chuyển vào ổ đĩa mạng
LOGIN <Tên SERVER >/<Tên USER >

b. Cài đặt đầy đủ: truy cập mạngmức NDS

Sử dụng phần mềm đi kèm card mạng, chạy trình cài đặt INSTALL.EXE, điền các tham số liên quan, nếu quá trình cài đặt thành công, các files sau sẽ được tạo:

LSL.COM
3C508.COM
IPXODI.COM
VLM.EXE
*.VLM
NET.CFG

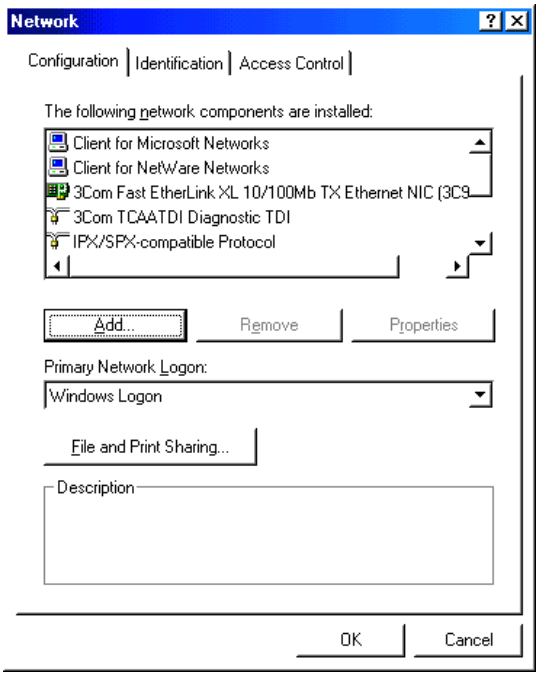
- Khởi động máy từ DOS, chạy lần lượt các tệp trên, có thể chạy các tệp tin trên bằng cách tạo tệp BAT.

Chuyển vào ổ đĩa mạng
LOGIN <Tên SERVER >/<Tên USER >

3.3. Cài đặt WORK STATIONS (Windows 9.X)

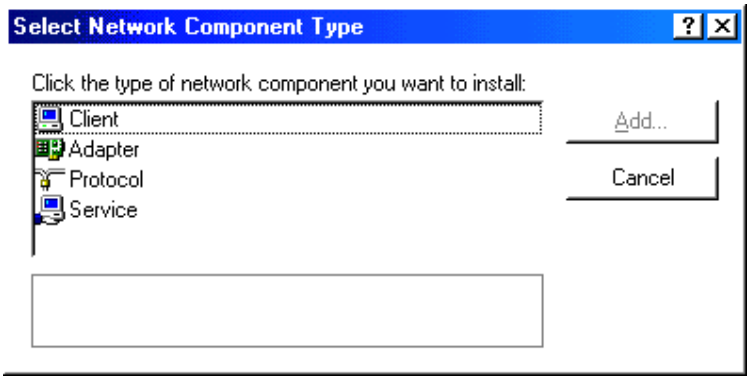
Các máy tính sử dụng hệ điều hành Windows 95, Windows 98, muốn sử dụng tài nguyên của Novell cần phải cài đặt phần mềm client cho máy, có thể dùng hệ điều hành Windows 9.x để cài đặt, các bước cơ bản để cài đặt:

Chọn trong Control Panel – Network: khi đó trên cửa sổ sẽ hiển thị những nội dung được cài đặt



Cài đặt card mạng và các tham số

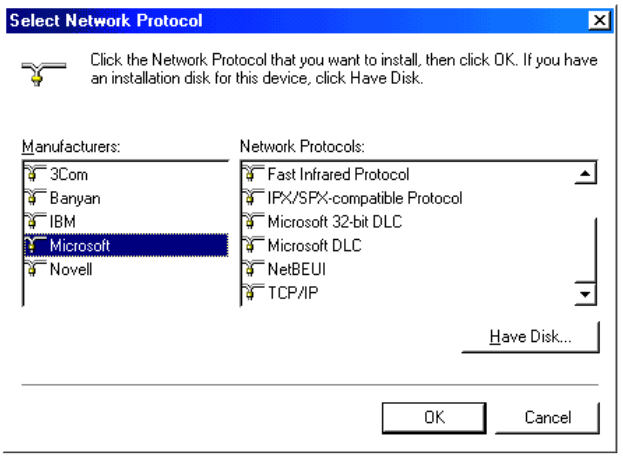
- Chọn ADD – Adapter



- Chọn card mạng trên danh sách card

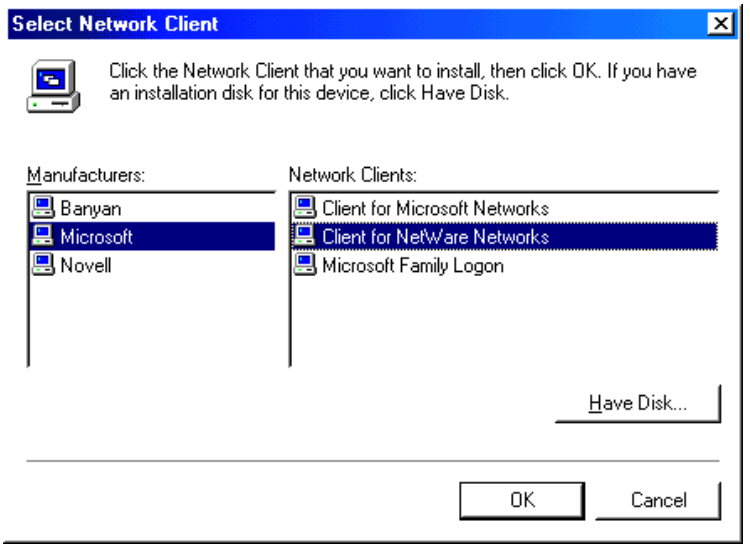
 Cài đặt Protocol

Chọn ADD - Protocol



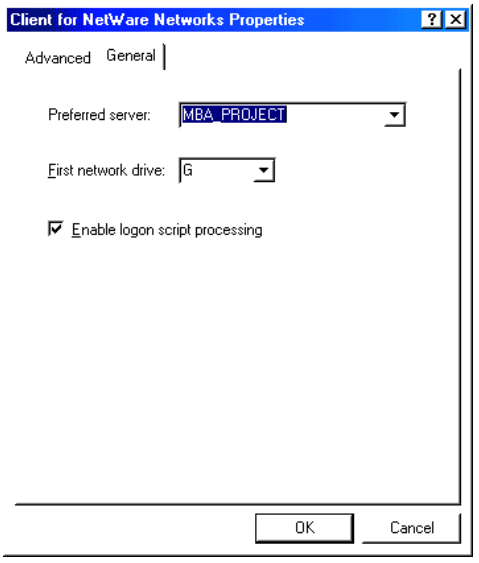
Chọn Microsoft, lựa chọn các Protocol cần thiết, để truy cập mạng Novell NetWare cần cài IPX/SPX

Cài đặt client (máy khách)



- Chọn Microsoft - client for Netware Networks

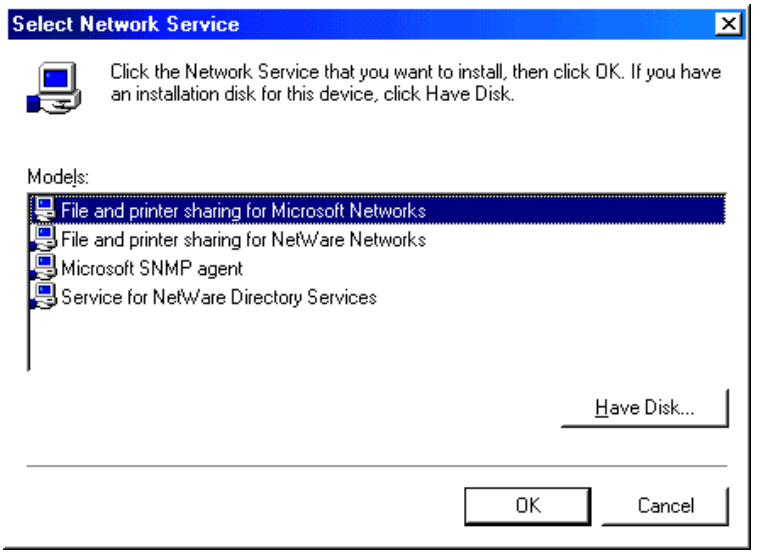
Điền tham số khi truy cập vào mạng:



- Preferres server: Bấm tên Novell Netware Server mà user muốn truy cập
- First network drive: ổ đĩa mạng (logic) đầu tiên được gán khi user truy cập
- Enable logon script processing: cho phép thực hiện login script của Novell

Cài đặt dịch vụ NDS

- Chọn ADD – Service, chọn Service for NetWare Directory Services



§ 4. QUẢN TRỊ HỆ THỐNG THƯ MỤC VÀ FILES

4.1. Cấu trúc thư mục và files của Novell NetWare

Khái niệm VOLUMES

VOLUMES được khởi tạo trong quá trình cài đặt mạng, nó là một phần của Netware partition. Mỗi VOLUMES đều có tên, VOLUMES đầu tiên được khởi

tạo có tên là SYS trên VOLUMES này các FILE hệ thống của NOVELL NETWARE được cài đặt. Các VOLUMES khác có thể đặt tên tùy ý. Để một VOLUMES được sử dụng như một tài nguyên trên mạng VOLUMES này cần được MOUNT.

Cấu trúc của VOLUMES SYS

/LOGIN : Chứa các chương trình dùng để vào ra trên mạng

/PUBLIC: Chứa các tiện ích ở trên mạng, mỗi USER đều có quyền truy nhập đến thư mục này.

/SYSTEM: Chứa các chương trình hệ thống của NOVELL NETWARE. Chỉ có ADMIN mới có quyền đối với thư mục này.

/MAIL : Chứa các thông tin của USER trên mạng

/ETC : Chứa một số FILES ví dụ có thể cài đặt.

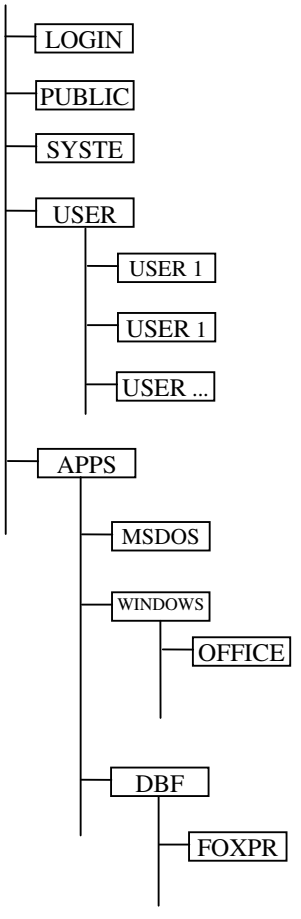
/DELETE.SAV: Chứa các FILE được cứu nằm trong thư mục đã bị xóa.

4.2. Thiết kế hệ thống thư mục và FILES

Nguyên lý chung

- Hệ thống thư mục của NOVELL NETWARE để riêng trên volume SYS
- Hệ thống thư mục của các USER nên tập trung để dễ dàng theo dõi và kiểm soát.
- Hệ thống các chương trình ứng dụng nên phân loại và để theo nhóm.

Ví dụ: nếu chỉ có 1 VOLUMES



Nếu có nhiều VOLUMES nên để mỗi VOLUMES lưu trữ một nhóm chương trình. Ví dụ: một volume dành cho các user, một volume dành cho group, một volume dành cho các ứng dụng khác.

Nếu có nhiều đĩa cứng thì nên tạo trên mỗi đĩa cứng thành 1 volume.

4.3. Quyền hạn

Quyền hạn 1 USER qui định khả năng truy nhập đến các tài nguyên của USER này. Như vậy quyền hạn là một trong những chức năng cơ bản để bảo vệ an toàn cho dữ liệu và hệ thống. Nhiệm vụ của ADMIN là gán các quyền này cho hợp lý. Một hệ thống làm việc tốt là một hệ thống trong đó các USER được trao quyền hạn vừa đủ, có khả năng khai thác tối ưu nhất các tài nguyên trên mạng.

Quyền hạn của các user đối với các tài nguyên được qui định bởi một trong hai cách sau:

TRUSTEE ASSIGNMENT

Quyền được gán trực tiếp bởi ADMIN hoặc các WORK GROUP cho các USER sau khi được gán USER này sẽ được gọi là STRUSTEE của tài nguyên tương ứng. Quyền STRUSTEE được chia làm 2 loại:

- Gán cho thư mục: STRUSTEE DIRECTORY RIGHT
- Gán cho FILES: STRUSTEE FILE RIGHT

INHERITED RIGHT MASK (Quyền thừa kế)

Quyền thừa kế được tự động gán cho FILE và thư mục khi USER tạo ra các thư mục hay FILES. Quyền thừa kế được chia làm 2 loại: quyền thừa kế của thư mục và quyền thừa kế của FILES .

Quyền của USER

- user ADMIN

Quyền có thể gán đối với FILES và thư mục

+ Có quyền tuyệt đối
+ Thay đổi các quyền

Các quyền hạn

- SUPERVISOR tất cả các quyền đối với Thư mục hoặc files. User có quyền này có thể gán quyền cho các user khác về thư mục và files

- READ (D,F) mở FILE hay đọc thư mục; khi muốn chạy một chương trình, xem nội dung của 1 tệp dữ liệu phải có quyền này.
- WRITE (D,F) Cho mở FILE và ghi các thay đổi đối với FILE này.
- CREATE (D) Tạo mới tệp tin và các thư mục con
  (F) Phục hồi các tệp tin sau khi tệp tin bị xóa
- ERASE (D) Xóa 1 thư mục FILE và các thư mục con
  (F) Xóa 1 tệp tin
- MODIFY (D,F) Thay đổi thuộc tính của thư mục và các tệp tin, đổi tên FILES và tên thư mục.
- FILESCAN (D,F) Cho phép liệt kê các tệp tin trên danh sách các thư mục
- ACCESS CONTROL Quyền thay đổi TRUSTEE, INHERITED trừ quyền SUPERVISOR

4.4. Thuộc tính

Thuộc tính là tính chất nội tại bên trong các tài nguyên và không phụ thuộc vào các đối tượng bên ngoài. Thuộc tính là một tác nhân bổ sung cho quyền hạn và cùng với quyền hạn tạo nên quyền thực sự của USER

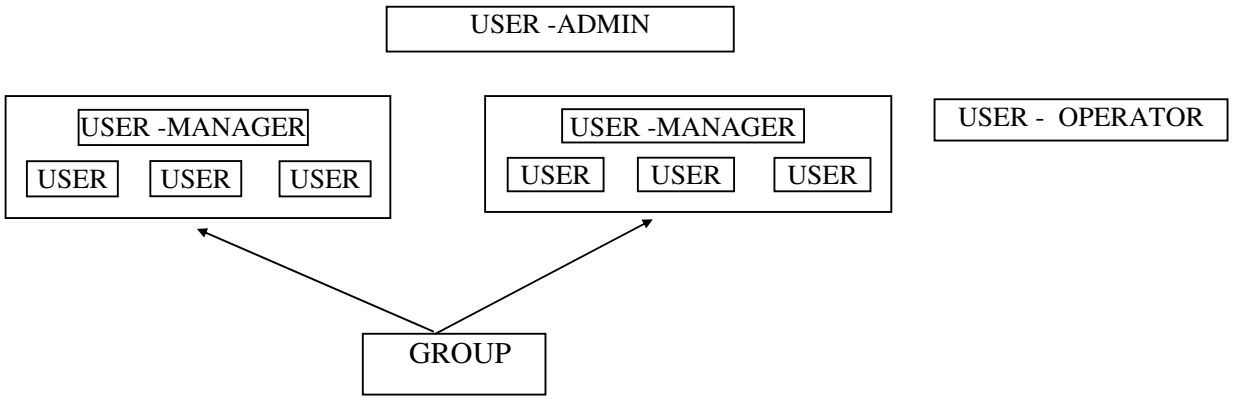
- ARCHIVE NEEDED (A,F,D) Được tự động gán bởi NOVELL NETWARE cho các FILES và thư mục Thuộc tính này là bít ARCHIVE của DOS (DOS có 4 thuộc tính RASH) (Status)
- CAN'T COMPRESS (Cc,F) Không nén files (Status)
- COMPRESSED (C,F) có thể nén files (Status)
- COPY INHIBIT (CI, F) cấm COPY từ Macintosh (Status)
- DELETE INHIBIT (D,F,D) cấm xóa, thuộc tính này mạnh hơn quyền xóa (ERASE) của USER đối với FILES hoặc thư mục (Status)
- DON"T COMPRESS (DC, F,D) Không cho phép nén tệp tin
- DON"T MIGRATE (DM, F,D) Không cho phép di chuyển tệp tin sang thiết bị lưu trữ thứ hai (như đĩa quang)
- EXCUTE ONLY (X,F) Cấm sao chép (*.com, *.exe) , thuộc tính này chỉ có SUPERVISOR có quyền gán và nếu đã một lần gán thì không thể xóa bỏ được nữa
- HIDDEN (H,F,D) ẩn khỏi lệnh DIR của DOS
- IMMEDIATE (IM,F,D) các files được gán thuộc tính này sẽ sớm bị nén
- PURGE (P,F) Thông thường files bị xóa có thể được phục hồi bằng lệnh SALVAGE,
- thuộc tính này sẽ làm mất tác dụng của lệnh SALVAGE. Do đó các files với thuộc tính PURGE sau khi bị xóa sẽ không thể cứu lại được.
- READ ONLY (Ro,F) thuộc tính này luôn đi kèm DELETE INHIBIT và RENAME INHIBIT . Chỉ cho phép đọc, không thể xóa hay đổi tên, cũng không thể thay đổi nội dung files . Thuộc tính này có thể bị loại bỏ bởi user có quyền modify đối với files này
- READ WRITE (Rw,F) nếu thuộc tính Ro bị loại bỏ thì thuộc tính Rw sẽ tự động gán.

- RENAME INHIBIT (RI,F,D) Không cho phép đổi tên files hay thư mục. Thuộc tính này mạnh hơn quyền MODIFY.
- SHAREABLE (S,F) Cho phép được sử dụng như tài nguyên chung trong mạng. Thuộc tính này đặc biệt hữu ích trong trường hợp nhiều user sử dụng chung.

- SYTEM (Sy,F,D) ẩn đối với lệnh DIR, đồng thời không cho phép xóa hay sao chép.
- TRANSACTIONAL (T,F) files được gán thuộc tính này sẽ được bảo vệ trong mạng bởi TTS (TRANSACTIONAL TRACKING SYSTEM)

§ 5. QUẢN TRỊ USER VÀ GROUP

Hệ thống quản trị user trong NOVELL NETWARE được tổ chức:



User là một định danh bao gồm tên và mật khẩu, muốn truy nhập vào mạng phải nhập tên user và mật khẩu. Những thông tin liên quan đến user được tập hợp và gọi là user account bao gồm: tên đầy đủ, mô tả khoản mục, thông tin về môi trường, thời gian được phép làm việc, những hạn chế đối với tài nguyên và những quyền của người sử dụng đối với hệ thống.

- Quản trị user và group bao gồm những công việc cơ bản sau:

1- Khởi tạo USER và GROUP

2- Gán quyền cho USER và GROUP

3- Cài đặt hạn chế của USER

4- Theo dõi và quản trị USER ACCOUNT

5- Khởi tạo các OPERATOR

Các user đặc biệt:

User ADMIN
- Tạo, xóa các USER và GROUP
- Tạo các quyền hạn của từng USER và GROUP
- Chỉ định WORKGROUP MANAGER
- Tạo các OPERATOR
- Kiểm soát USER ACCOUNT
- Tạo các thuộc tính cho toàn bộ tài nguyên được sử dụng trong hệ thống mạng.
- Khởi tạo và điều khiển máy in trong mạng
- Xử lý và điều khiển toàn bộ hoạt động của hệ thống thông qua các lệnh và tiện ích của mạng.

OPERATOR : là 1 user có quyền hạn ngầm định để thực hiện một số nhiệm vụ, có 3 loại OPERATOR
- FILESERVER CONSOLE OPERATOR người có quyền sử dụng FCONSOLE để kiểm soát các liên kết mạng, gửi các thông điệp hệ thống.
- PRINT SERVER OPERATOR quyền được điều khiển công việc in trên PRINT SERVER
- PRINT QUEUE OPERATOR quyền điều khiển các hàng đợi của PRINT SERVICE
- Tất cả các OPERATOR đều do ADMIN khởi tạo và giao quyền.

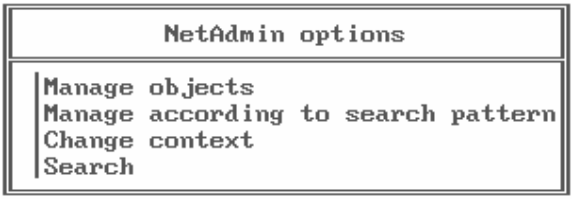
Thiết kế hệ thống users groups

5.1. Quản trị USER

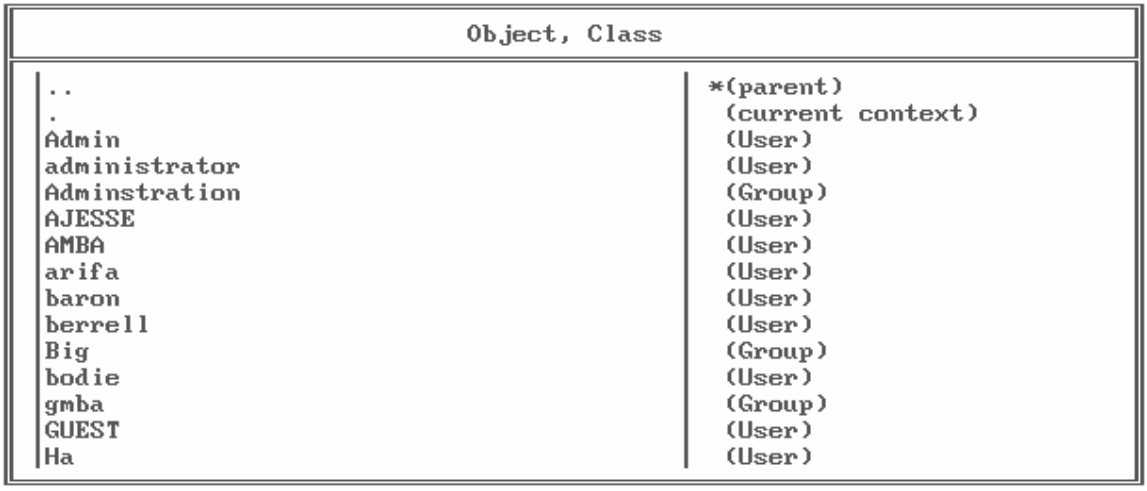
Quản trị user thông qua chương trình NETADMIN

Tạo USER

- Trong thực đơn của NETADMIN :

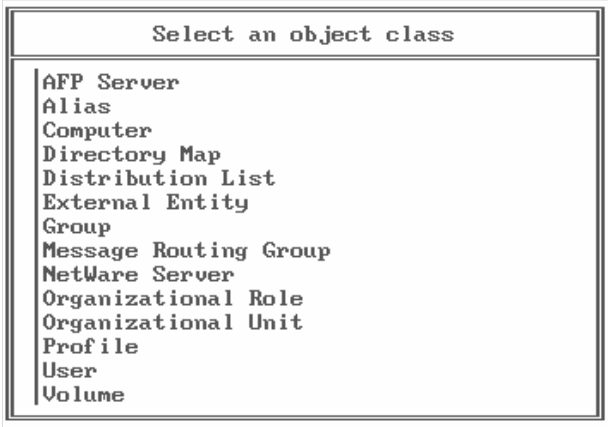


- Chọn Manage object



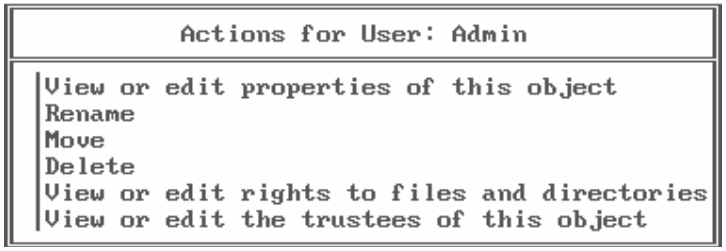
Trên màn hình hiển thị danh sách các object (user, group, printer ...)

Bấm INSERT để tạo user mới, trên màn hình hiển thị:

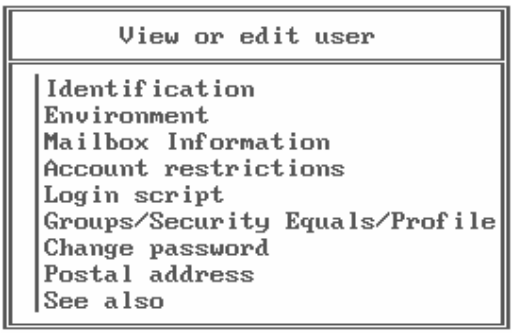


- Chọn "USER", bấm tên user mới

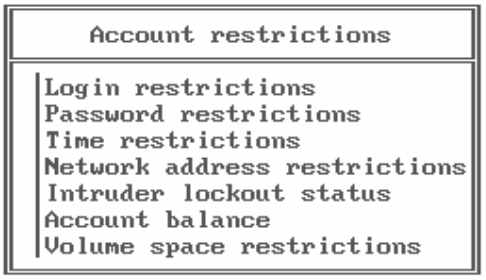
- Cài đặt các thông số cho user: các thông số cần xác định cho user chính là các hạn chế của user bảo đảm cho sự an toàn và bảo mật của hệ thống
- Trong thực đơn của NETADMIN (Object, Class)
  Chọn tên của USER trong danh sách
  Trên màn hình hiển thị thực đơn để có thể cài đặt, hiệu chỉnh các tham số của user



Chọn “View or edit properties or this object “



Các tham số sau đây cần hiệu chỉnh:



ACCOUNT RESTRICTION

- ACCOUNT DISABLE: YES user không được phép vào mạng
  No user được phép vào mạng

- ACCOUNT HAS EXPRIRATION DATE
  NO user vào mạng với tên mình trong cùng một thời gian
  YES user chỉ có thể vào mạng với tên mình tại workstation nhiều nhất là số lần được chỉ ra trong MAXIMUM CONECTION
- ALLOW USER TO CHANGE PASSWORD
  YES user có quyền thay đổi mật khẩu
  NO user không có quyền thay đổi mật khẩu

- MINIMUM PASSWORD LENGTH: độ dài tối thiểu mật khẩu
- FORCE PERIODIC PASSWORD CHANGES
  YES user bắt buộc thay đổi mật khẩu định kỳ

CHANGE PASSWORD thay đổi mật khẩu

LOGIN SCRIPT khởi tạo hay sửa nội dung của USER LOGIN SCRIPT

Time RESCTRICTIONS Khai báo hạn chế về thời gian làm việc của user trong hệ thống

VOLUMES/DISK RESCTRICTIONS khai báo hạn chế về dung lượng không gian trong các VOLUMES mà các user này có thể sử dụng.

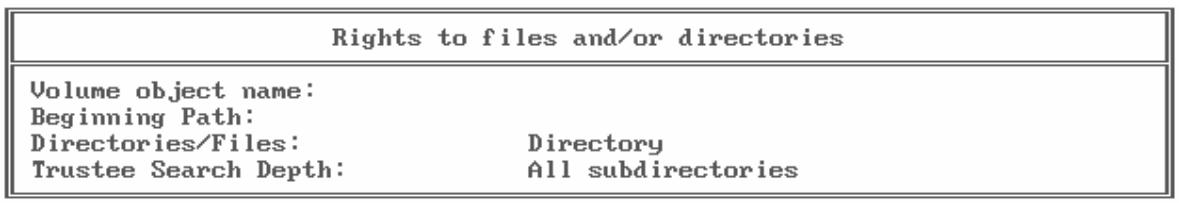
TRUSTEE DIRECTORY ASSIGNMENT

TRUSTEE FILE ASSIGNMENT

Hai lựa trọn trên dùng để gán quyền tương đương cho user sử dụng directory và files

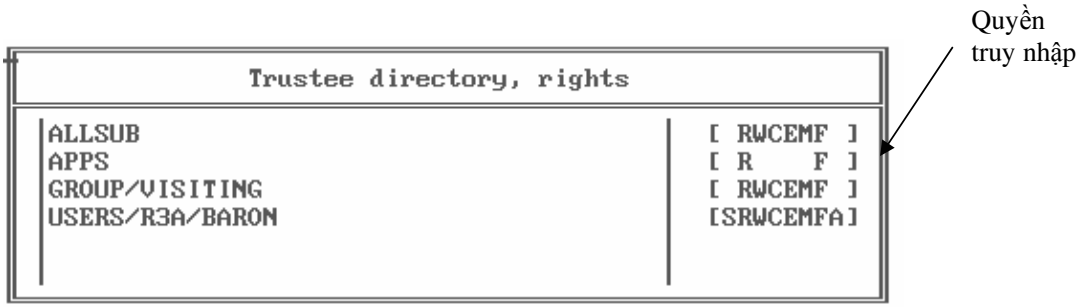
ADMIN hoặc MANAGER có thể gán quyền tương đương cho một user ngang với một user hoặc một nhóm khác.

- Chạy trình NETADMIN
- Chọn USER , Chọn "View or edit right to files and directories "



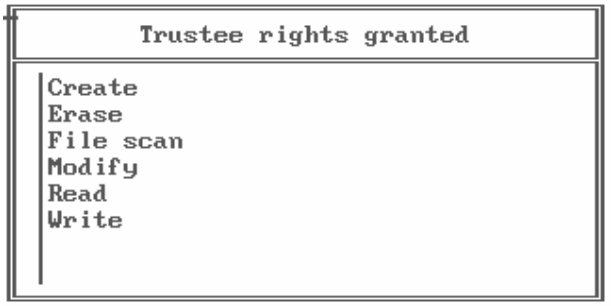
Chọn "Volume object name" để nhập tên volume cần gán quyền (có thể bấm ENTER sau đó INSERT để chọn trên danh sách)

Bấm F10 hiển thị quyền truy nhập các thư mục và các files của user



- Chọn SECURITY EQUIVALENCES

Trong cửa sổ bấm INSERT để gán quyền, hoặc DETETE để hủy bỏ quyền trong danh sách.

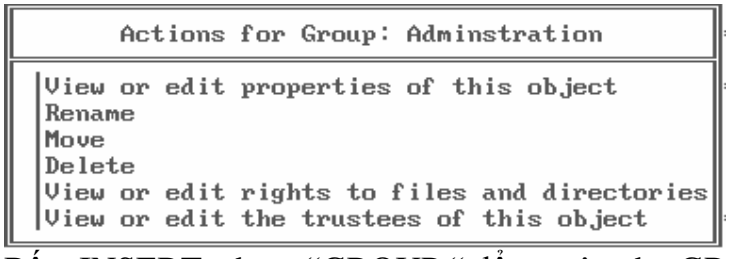


5.2. Quản trị GROUP

Khởi tạo GROUP

- Chạy NETADMIN

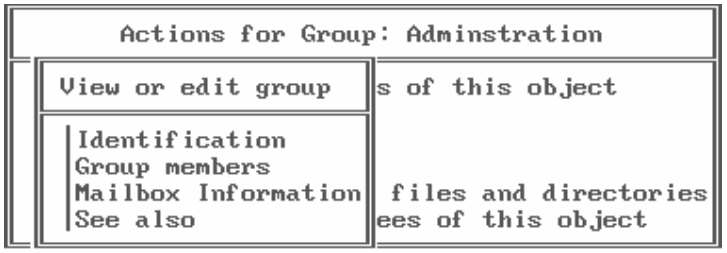
Trên màn hình hiển thị danh sách các object (user, group, printer ...)



Bấm INSERT, chọn "GROUP " để tạo tên cho GROUP

Bổ sung, bớt các thành viên của nhóm

- Chạy NETADMIN ... , chọn group, chọn "view or edit properties of this object"
- Chọn "group members "



Bấm INSERT để thêm user vào group.

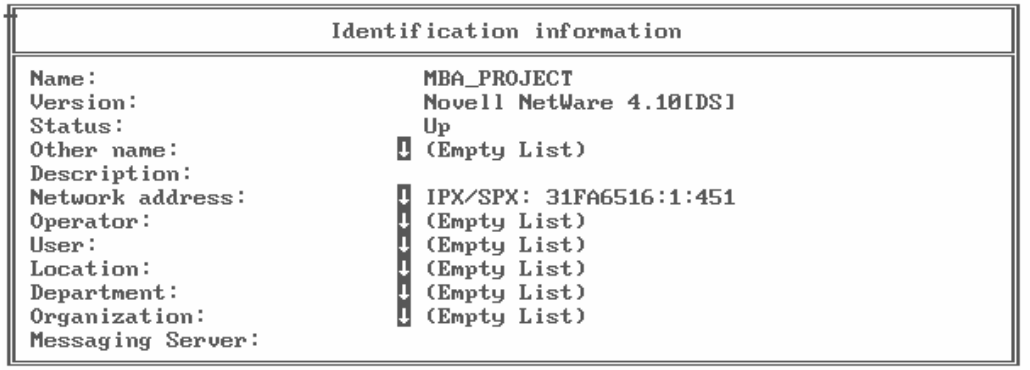
Trên Novell Netware 4.x Group administration là một group đặc biệt, thành viên của group này có quyền tương đương với ADMIN

- Khởi tạo OPERATOR

Tạo FILESERVER CONSOLE OPERATORS

- Chạy NETADMIN
- Chọn object "Netware Server"

Chọn "Identification"



- Chọn "OPERATOR ", chọn user gán thành operator

 Các OPERATOR khác được khởi tạo bằng PCONSOLE

5.3. Quản trị USER ACCOUNT

Quản trị USER ACCOUNT là chức năng cho phép quản trị và theo dõi tiến trình làm việc của từng USER trong hệ thống. Hệ thống tài khoản công nợ ACCOUNTING SYSTEM của NOVELL NETWARE cho phép kiểm tra và tiến hành tính toán cụ thể trên các tham số.

- Đọc dữ liệu: Khối lượng dữ liệu USER đã tiến hành đọc trên các VOLUMES của SERVER

=Σ hệ số * số đơn vị đọc

Số đơn vị đọc: BLOCK/30" (1BLOCK=4KB NOVELL NETWARE)

Hệ số có thể thay đổi theo từng đơn vị thời gian là 30 phút trong ngày (48 hệ số/ngày)

- Ghi dữ liệu: Khối lượng dữ liệu được ghi trên các VOLUMES của SERVER

Cách tính tương tự như đối với đọc dữ liệu

- Thời gian làm việc: Tổng thời gian user làm việc trên mạng, đơn vị tính là 30 phút

=Σhệ số * số đơn vị làm việc

-Thời gian kết nối trong mạng: tổng số thời gian user kết nối trong hệ thống, đơn vị tính là 30 phút

=Σhệ số * số đơn vị làm việc

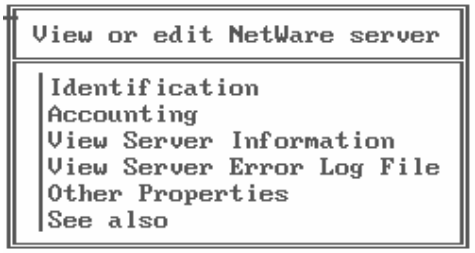
- Dung lượng đĩa đã được sử dụng: Tổng dung lượng đĩa đã được sử dụng

- Các dịch vụ mà user đã tiến hành trên SERVER. Các dịch vụ này được thể hiện thành những request (yêu cầu) gửi lên SERVER như E-mail, in ấn trên mạng ...

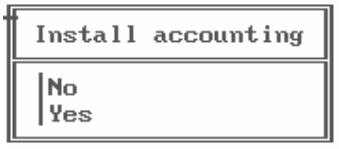
5.4. Thao tác với USER ACCOUNT

Cài đặt Accounting system

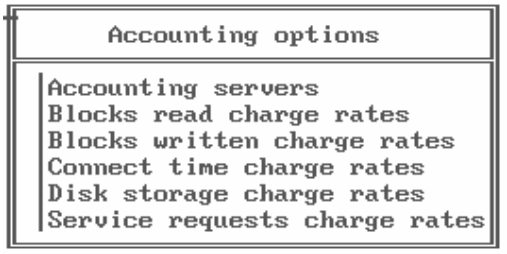
- Chạy NETADMIN, chọn server cần cài đặt, chọn "View or edit properties of this object"



- Chọn ACCOUNTING



- Chọn YES để cài đặt

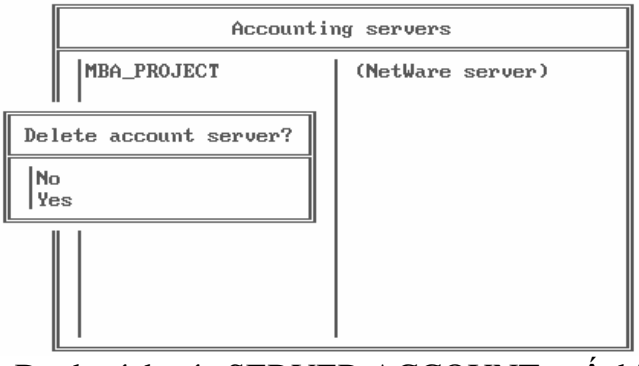


Đặt các yêu cầu:

- BLOCK READ CHARGE RATES: cài đặt hệ số tính toán khối lượng đọc đĩa
- BLOCK WRITTEN CHARGE RATES: cài đặt hệ số tính toán khối lượng ghi đĩa
- CONNECT TIME CHARGE RATES: cài đặt hệ số tính toán thời gian truy cập
- SERVICE REQUEST CHARGE RATES : cài đặt hệ số dịch vụ mạng đã yêu cầu

Xóa SERVER ACOUNT

- Chạy NETADMIN
- Chọn các bước giống trên, chọn "Acounting server ", bấm DELETE

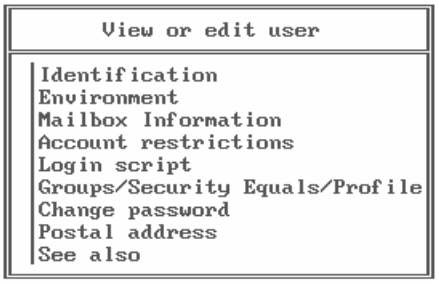


- Danh sách các SERVER ACCOUNT xuất hiện, bấm DELETE để xóa

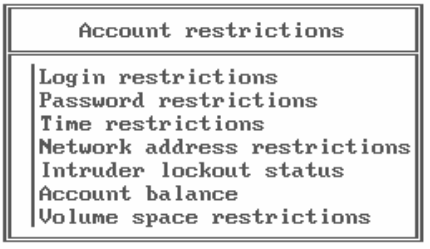
Tính công nợ

Để tính công nợ (BALANCE) của một user :

- Chạy NETADMIN
- Chọn user trong danh sách, chọn "View or edit properties of this object "



- Thực hiện ACCOUNT BALANCE



Trên màn hình hiển thị thực đơn, cần điền các tham số:

Account balance:

Điền số

Allow unlimited credit: (Yes: không giới hạn tài khoản, No: giới hạn)

§ 6. LOGIN SCRIPTS

6.1. Khái niệm:

Những lệnh cần thực hiện một cách tự động khi USER nối vào mạng được tập hợp và ghi trong một tệp được gọi là LOGIN SCRIPT. LOGIN SCRIPT đóng vai trò như AUTOEXEC của DOS .

NOVELL NETWARE phân biệt 3 LOGIN SCRIPT

- SYSTEM LOGIN SCRIPT

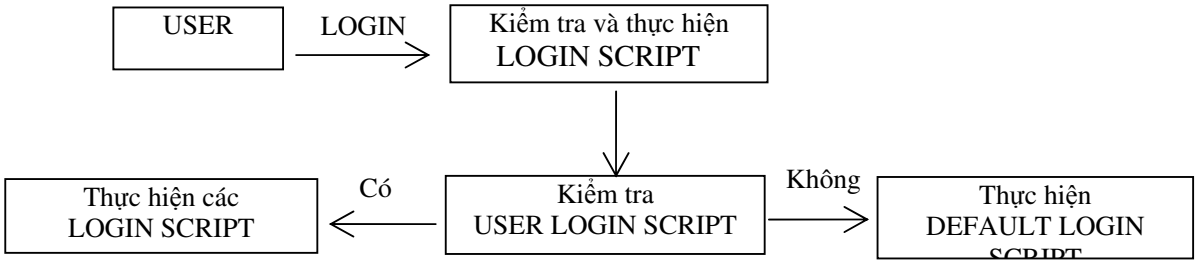
Chứa các lệnh mà mọi user thuộc organization này khi vào mạng sẽ được thực hiện. LOGIN SCRIPT được khởi tạo bằng ADMIN

- PROFILE LOGIN SCRIPT chứa các lệnh trong một object, có thể gán để một số user có thể thực hiện LOGIN SCRIPT này.

- PERSONAL LOGIN SCRIPT

Chứa các lệnh riêng biệt cần thực hiện của user khi vào mạng . Mỗi user có thể khởi tạo riêng cho mình một USER LOGIN SCRIPT.

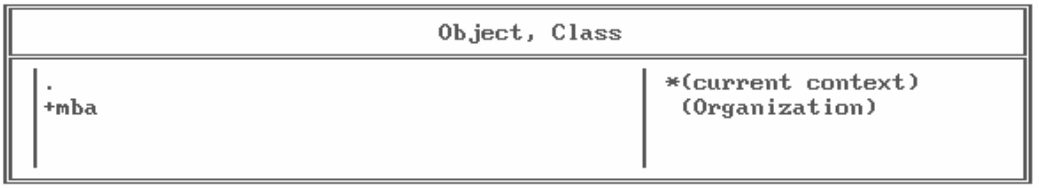
- DEFAULT LOGIN SCRIPT ngầm định của mọi user khi USER LOGIN SCRIPT của user này chưa được khởi tạo



6.2. Khởi tạo LOGIN SCRIPT

Khởi tạo SYSTEM LOGIN SCRIPT

- Chạy NETADMIN



- Chọn tên Oganizational Unit
- Bấm F10, chọn "View or edit properties of this object "
- Chọn SYSTEM LOGIN SCRIPT

Khởi tạo hoặc hiệu chỉnh SYSTEM LOGIN SCRIPT viết giống như trên hệ soạn thảo văn bản

Khởi tạo PERSONAL LOGIN SCRIPT

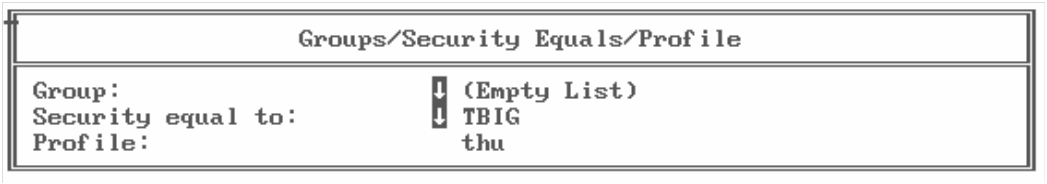
- Chạy NETADMIN
- Chọn tên user, bấm F10, chọn "View or edit properties of this object "
- Chọn LOGIN SCRIPT

Khởi tạo hoặc hiệu chỉnh LOGIN SCRIPT viết giống như trên hệ soạn thảo văn bản

 Khởi tạo PROFILE LOGIN SCRIPT

- Tạo một PROFILE (là một object, giống như tạo một user)
- Gán quyền, viết LOGIN SCRIPT ...
- Gán quyền sử dụng PROFILE này cho user

+ Trong NETADMIN chọn user , chọn "View or edit properties of this object ", chọn "Group/Security/Equals/Profile"



+ Chọn PROFILE và gán tên PROFILE

6.3. Các lệnh hệ thống trong LOGIN SCRIPT

1. COMSPEC

Chức năng: Khi kết thúc 1 ứng dụng bao giờ DOS cũng nạp lại COMMAND.COM, lệnh này chỉ đường dẫn hệ thống chứa tệp COMMAND.COM để hệ điều hành nạp lại tệp tin này.

Cú pháp: COMSPEC <D:> <PATH> COMMAND.COM

2. DRIVE

Chức năng: Chuyển ổ đĩa làm việc, đặt ổ đĩa mạng (được ánh xạ-MAP) như là ổ đĩa DOS mặc nhiên.

Cú pháp:

DRIVE <D:>

DRIVE *n:

n thứ tự ổ đĩa mạng

3. # (exec)

- Chức năng: Thi hành một lệnh ngoại trú ngay trong LOGIN SCRIPT sau đó quay trở về

- Cú pháp: # <Tên lệnh>

Chú ý: Chỉ thực hiện các lệnh ngoại trú

- Không nạp các TSR (thường trú)

- # nằm trên 1 dòng

4. EXIT

- Chức năng: Ngừng thi hành các lệnh còn lại của LOGIN SCRIPT và chuyển về DOS thực hiện tiếp

- Cú pháp: EXIT

EXIT "Lệnh"

- Chú ý: SERVER, STATION phải tương thích với IBM

- "Lệnh" sau EXIT phải nằm trong ổ đĩa đã được ánh xạ

5. IF ... THEN

- Chức năng: cho phép LOGIN SCRIPT thực hiện quyết định dựa trên 1 số điều kiện

Cú pháp:

IF điều kiện THEN lệnh

IF điều kiện AND điều kiện THEN lệnh

IF điều kiện THEN BEGIN

    Lệnh

(ELSE)

Lệnh
END
6. WRITE
- Chức năng: hiển thị một đoạn văn bản lên màn hình
- Cú pháp :
WRITE “Text”;”more text “; identifier (biến hệ thống)
WRITE “Text %Biến”
(Tên biến phải viết chữ in hoa)
* Danh sách các biến :
DAY Số ngày trong tháng (1-31)
DAY_OF_WEEK Tên ngày trong tuần (MONDAY, TUESDAY..)
HOUR giờ hiện hành (1-12)
GREETING_TIME “morning”, “afternoon “, “evening” theo đồng hồ
MONTH Tháng (1-12)

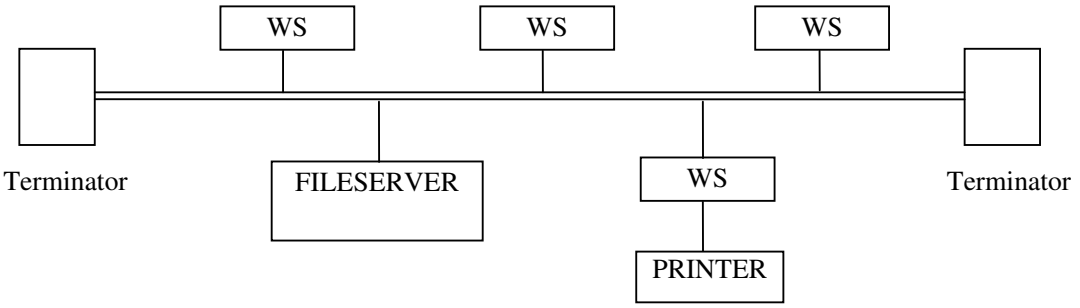
FULL_NAME Tên đầy đủ của USER
LOGIN_NAME Tên user khi nối vào mạng

STATION Chỉ số của trạm làm việc
P_STATION Vị trí vật lý của trạm làm việc
OS Tên của hệ điều hành
Các biến của DOS

7. Lệnh MAP

§ 7. QUẢN TRỊ DỊCH VỤ IN TRÊN MẠNG

7.1.Tổng quan về công việc in trên mạng (DOS client)



- PRINT SERVICE đã được cài đặt
- ở FILESERVER đã được nạp PSERVER

- ở WORKSTATION : đã được chuẩn bị khởi tạo cổng máy in vào mạng và chuyển dữ liệu ra PRINT SERVER

- ở WORKSTATION có nối máy in: máy in đã được nối vào mạng

- ở các chương trình ứng dụng: Khởi tạo máy in, khởi tạo cổng in (LPT logic)

7.2. Cài đặt PRINT SERVER , cài đặt máy in, cài đặt PRINT QUEUE,

Người thực hiện (user): ADMIN

Chạy PCONSOLE

```
Available Options

Print Queues
Printers
Print Servers
Quick Setup
Change Context
```

- Cài đặt PRINT SERVER

+ Chọn PRINT SERVERS : Khai báo tên PRINT SERVER

```
Print Server Information

Printers
Information and Status
Users
Operators
Description
Password
Audit
```

- Cài đặt máy in

+ Chọn PRINTER, khai báo tên máy in

```
                Printer HP_5POffice Configuration

Printer number:          4
Printer status:          (See form)
Printer type:            Parallel
Configuration:           (See form)
Starting form:           0
Buffer size in KB:       3
Banner type:             Text
Service mode for forms:  Minimize form changes within print queues
Sampling interval:       5
Print queues assigned:   (See list)
Notification:            (See list)
```

Tiếp theo nhập vào các thông số kỹ thuật của máy

- Cài đặt print queue

Trong thực đơn PCONSOLE

+ Chọn PRINT QUEUES : gõ tên queue

+ Chọn QUEUE SERVERS để gán queue cho print server đã được cài đặt ở phần trên

Thực đơn QUEUE USER dùng để gán các user được quyền sử dụng các queue này

Thực đơn QUEUE OPERATOR dùng để gán USER điều khiển hàng đợi in; các user này có quyền xem và điều khiển print jobs hủy hay thay đổi thứ tự in

- Gán các printer được cài đặt cho print queue

+ Chọn PRINT SERVERS

+ Chọn PRINTER

Một danh sách các printer đã cài đặt xuất hiện, với mỗi máy in làm như sau:

+ Printer queues assigned để gán QUEUE

+ Gõ thứ tự ưu tiên PRIORITY

Khởi động PRINT SERVER tại FILE SERVER

Trong tệp AUTOEXEC.CNF viết dòng lệnh

LOAD PSERVER <Tên PRINT SERVER>

Khai báo, chuẩn bị đường truyền dữ liệu cho WORK STASTION

- Dùng lệnh CAPTURE

Có thể dùng lệnh này ở autoexec hoặc trong login script

Cú pháp : CAPTURE q=<tên print queue> L=<từ 1 đến 3 đây là tên cổng LPT> ....

Có thể có thêm một số tham số:

NB Bỏ qua trang tiêu đề của công việc in

NT Không thay đổi ký tự TAB

NF Máy in không kéo trang trước khi in

...

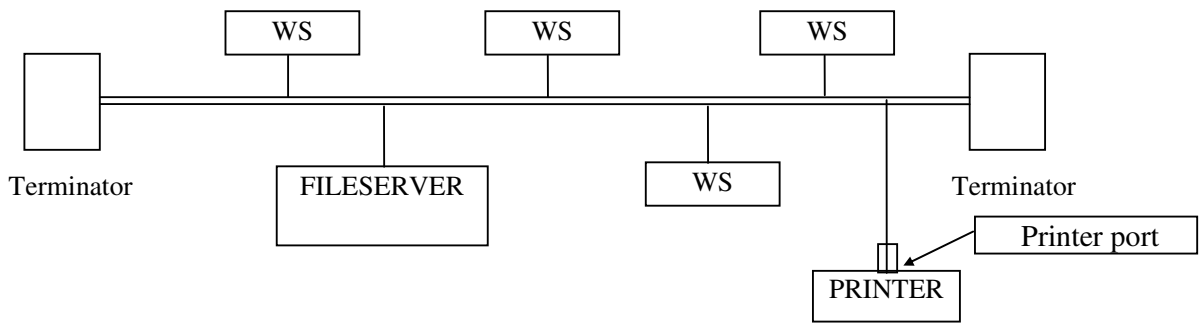
7.3. Ở work station : nối máy in với mạng

- Chạy NPRINTER
- Chọn PRINT SERVER chọn máy in cần nối vào mạng

Ở work station : chuẩn bị cho việc in, như khai báo cổng máy in, máy in ...

Thông thường các chương trình ứng dụng thường được chuẩn bị cổng in để có thể gửi tín hiệu ra máy in. Cổng này có thể là LPT1, LPT2 cổng in vật lý LPT1, LPT2, LPT3 cổng in LOGIC. Trước khi gửi in phải khai báo cổng in sẽ in ra, cổng in được chọn phải tương ứng với cổng in được khai báo trong CAPTURE

7.4. Tổ chức máy in PRINT SERVER



- Máy in được nối trực tiếp với mạng qua bộ điều khiển cắm ở cổng máy in (Printer connection)
- Tạo PRINT SERVER : thao tác giống như trường hợp máy in nối vào máy tính
- Chạy chương trình SETUP của Printer connection): chọn và điền các tham số
    + Type: manual load (remote)
    + Port: other

§ 8. MỘT SỐ LỆNH CƠ BẢN CỦA NOVELL NETWARE

8.1. Lệnh NDIR

```
NDIR                          General Usage Help                          4.25

Purpose: View information about files, directories, and volumes.

For help on:                                    Type:
  Display format                                  NDIR /? FOR
  Sorting features                                NDIR /? SORT
  Search filters (restrictions)                   NDIR /? RES
  Attribute filters                               NDIR /? AT
  Other options                                   NDIR /? OPT
  Syntax                                          NDIR /? SYN
  All help screens                                NDIR /? ALL

For example, to:                                Type:
  See all files in current directory              NDIR *.*

  See only directories on drive C:                NDIR C:\*.* /DO
```

Chức năng: Xem các thông tin về tệp, thư mục và volume.
Cú pháp:
NDIR / Tham số
Ví dụ:
NDIR
Hiển thị danh sách tệp tin, các thông tin về tệp ở thư mục hiện thời.

8.2. Lệnh NCOPY

```
NCOPY                              General Help                              4.13

Purpose:  To copy files and directories.
Syntax:   NCOPY source path [target path] [options]

To:                                                                      Use:
  Copy subdirectories                                                    /S
  Copy subdirectories including empty directories                        /S/E
  Copy files with archive bit set                                        /A
  Copy files with archive bit set, then clear the bit                    /M
  Copy sparse files                                                      /F
  Inform when non-DOS file information will be lost                      /I
  Copy only DOS information                                              /C
  Read after write verify on local drives (DOS only)                     /V
  Retain compression on supported media                                  /R
  Retain compression on unsupported media                                /R/U
  Display version information                                            /VER

For example:                                  Type:
  To copy all files and subdirectories from      NCOPY SYS:*.* G:*.* /S
   volume SYS to drive G:
```

Chức năng: sao chép tệp và thư mục

Cú pháp: NCOPY Tham số nguồn Tham số đích [Options]

Ví dụ: NCOPY *.* C:\ /S

Copy tất cả các tệp, các thư mục con ở thư mục hiện thời sang ổ đĩa C:\

8.3. Lệnh Login.

Chức năng: cho phép user truy nhập vào mạng Novell Netware

Cú pháp:

Ví dụ:

Login [/Ver] [[Server|Tree]/user name] [/optional]

Các option:

/Ver: Cho biết version mạng ta login

/NS: Khi vào mạng không chạy các lệnh trong login script.

/CLS Khi vào mạng, xoá màn hình sau khi chạy các lệnh trong login script.

/S filename Chạy các lệnh login script trong file có tên là filename.

/NB Không hiển thị bảng (banner) trên màn hình khi login.

8.4. Lệnh Logout.

Chức năng: ra khỏi mạng, trạm làm việc chỉ nhìn thấy thư mục LOGIN của máy chủ, trước khi tắt máy người sử dụng cần thực hiện lệnh này.

Cú pháp:

LOGOUT [server name | T | /Ver]

Các option:

Server name Ra khỏi máy chủ có tên là server name.

/T Ra khỏi mạng để lại dịch vụ thư mục bindery connection

/Ver Ra khỏi mạng và cho biết version của mạng đang sử dụng.

Ví dụ:

Máy đang làm việc (login) với 3 máy chủ có tên là FS1, FS2 và FS3.

Muốn ra khỏi máy chủ FS1, ta gõ lệnh: **Logout FS1**

Muốn ra khỏi tất cả các máy chủ, ta gõ: **Logout**

8.5. Lệnh NLIST.

a. Chức năng: cho biết các thông tin về các user, group và các đối tượng khác.

Cú pháp:

NLIST Class type [property search option] [display option] [basic option]

Classtype: có các loại như sau:

Server	Profile
Computer	Alias

Directory map	Print queue
Organization Unit	Printer
User	Print server
Group	Organization
Volume	AFP server

Ví dụ:

Nlist Server	Xem các server trong context hiện tại.
Nlist "directory map"	Xem ánh xạ ổ đĩa của trạm làm việc hiện tại.
Nlist "organization"	Xem các mức tổ chức của mạng.

b. Chỉ định các option trong khi xem các đối tượng.

Cú pháp:
NLIST Classtype [=object name] [basic option]
Các option:
/A Xem các user và server.
/Ver Cho biết version mạng ta đang làm việc.
/Tree Xem các cây thư mục.
/B Xem các Server trong chế độ bindery
/S Xem các Server trong context hiện tại.
Ví dụ

Nlist server /S	Liệt kê các Server trong context hiện tại
Nlist server /B	Liệt kê các Server trong chế độ Bindery
Nlist User /A	Xem các user đang login
Nlist Volume /CO "O = FPT"	Xem các volume trong context O = FPT

c. Chọn dữ liệu nào sẽ xem.

Cú pháp:
NLIST Classtype [=object name] [display option]
Các option:
/D Xem chi tiết toàn bộ (detail).
/N Chỉ xem tên đối tượng thôi.
Show property chỉ định thuộc tính cần xem là property
Ví dụ:

Nlist Group = Manager /D	Xem thông tin chi tiết trong nhóm Manager
Nlist User = ThangC /D	Xem thông tin chi tiết của user ThangC
Nlist User Show "telephone", "street address"	Xem địa chỉ, số điện thoại của toàn bộ các user.

8. 6. Lệnh MAP.

Chức năng: ấn định tên ổ đĩa logic tới một đường dẫn (hay một thư mục).

Cú pháp:

MAP [option | /ver] [search:=[driver:=]] [driver:=] [path] [/w]

Các option:

INS	Thêm một đường dẫn (insert).
DEL	Xoá một đường dẫn (delete).
N	Thêm một đường dẫn với tên ổ đĩa hợp lệ tiếp theo.
R	Thêm ổ đĩa như là thư mục gốc.
/VER	Cho biết version mạng ta đang sử dụng.
/W	Không thay đổi (ghi lại) môi trường chính.

Ví dụ

Map N TH/SYS:Login	Thêm một đường dẫn với tên ổ đĩa hợp lệ tiếp theo, chỉ tới thư mục Login trong Volume SYS của máy chủ tên là TH.
Map S10:=W:=TH/Data:setup	Thêm đường dẫn thứ 10 với tên ổ đĩa logic là W chỉ tới đường dẫn là thư mục setup trong volume có tên Data của máy chủ TH.
MAP R H:= TH/data:User/KT	Ấn định ổ đĩa logic gốc H là thư mục KT trong volume có tên

Data của máy chủ TH.

8.7. Lệnh WHOAMI

Chức năng: cho biết người đang vào mạng tại trạm làm việc hiện tại là ai.

Cú pháp:

Whoami [Server] [/S] [/G] [/W] [/R] [/O] [/ALL] [/C] [/VER]

Các option:

Server: Chỉ định tên máy chủ, để trống nếu muốn xem toàn bộ các máy chủ có liên quan.
/S Cho biết quyền tương đương (với ai) của người đang login.
/G Cho biết người đang login thuộc nhóm nào.
/W Cho biết thông tin về nhóm của người đang login.
/R Cho biết các quyền sử dụng mạng của người đang login.
/C Cho phép xem từng trang màn hình.
/Ver Cho biết version mạng đang sử dụng.

8.8. Lệnh SETPASS.

Chức năng: Thay đổi mật khẩu của user đang trong mạng.
Cú pháp:
Setpass [Ver] [[Server name/] [username]
Các option:
/Ver cho biết version mạng đang sử dụng.
Servername thay đổi mật khẩu trong máy chủ có tên là Servername
Username thay đổi mật khẩu của user khác có tên Username (nếu ta có quyền)

8.9. Lệnh SEND.

Chức năng: gửi một thông báo tới một đối tượng.
Cú pháp:
Send "message" [TO] <object>
Các option:
message dòng thông báo mà ta muốn gửi.
object đối tượng mà ta cần gửi thông báo tới (user, group, server).
Ví dụ:

Send "Hi" to Guest Gửi thông báo "Hi" tới user tên là Guest

Send "Good Morning" to FIS Gửi thông báo " Good Morning " tới group tên là FIS

8.10. Lệnh Rconsole.

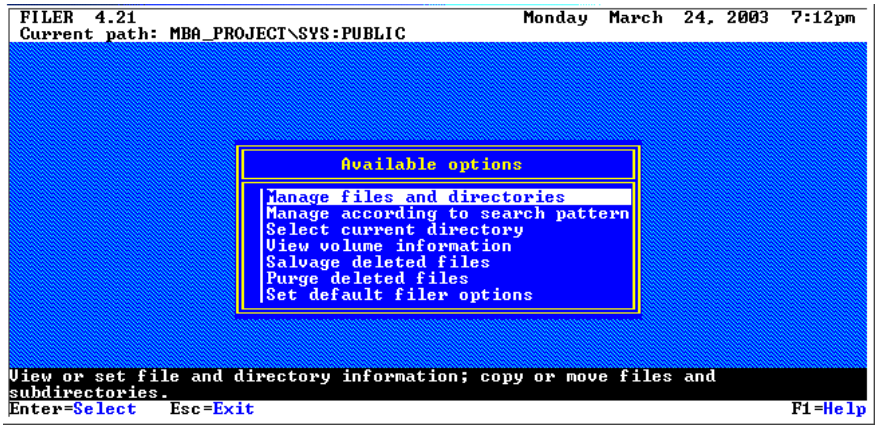
Chức năng: Dùng để thao tác với máy chủ và các lệnh trên nó ở trạm làm việc (remote console).

Cú pháp: Rconsole Tên máy chủ

Nếu không vào tên máy chủ, máy sẽ đưa ra danh sách các máy chủ đang hoạt động và hỏi ta muốn làm việc với máy chủ nào, và dạng connection của trạm làm việc đang dùng.

8.11. Lệnh Filer

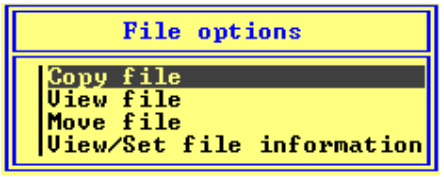
Chức năng: tiện ích về thư mục và tệp tin



Cú pháp: Đây là một trình làm theo giao diện thực đơn, các chức năng cơ bản được thể hiện như sau:

- Manage files and directories: Sao chép, xem, thay đổi, quyền hạn và thuộc tính của tệp tin, thư mục.

Sau khi lựa chọn thư mục và tệp tin, thực đơn cuối cùng cho phép thực hiện các chức năng như sau:



- Manage according to search pattern: tìm kiếm tệp tin
- View volume information: Xem các thông tin về volume
- Salvage deleted files: Khhoi phục tệp tin bị xóa, nếu chưa dùng lệnh Purge

Purge deleted files: xóa vĩnh viễn tệp tin

PHẦN III

HỆ ĐIỀU HÀNH MẠNG WINDOWS 2000

§ 1. TỔNG QUAN VỀ MICROSOFT WINDOWS 2000

1.1. Windows 2000 Server

Windows 2000 Server là hệ điều hành trên các File Server để quản lý mạng cục bộ, mạng diện rộng cũng như phục vụ các ứng dụng, đây là phiên bản kế tiếp của Windows NT Server. Windows 2000 Server cho phép quản lý, chia sẻ thông tin, truy nhập máy in và các thiết bị ngoại vi trên mạng. Nó cung cấp cơ sở hạ tầng ứng dụng, hỗ trợ nhiều giải pháp nghiệp vụ cho các hệ quản trị cơ sở dữ liệu, các ứng dụng khác của Oracle, Sybase, Btrieve, v.v. ... Windows 2000 Server có nhiều tính năng mới so với phiên bản Windows NT Server.

Có bốn phiên bản của hệ điều hành Windows 2000, một phiên bản dành cho các máy trạm (workstation) và các phiên bản khác là các cấp độ phức tạp dành cho server khác nhau.

Windows 2000 Professional

Phiên bản "Professional" được dành cho các máy trạm kế tiếp bản Windows NT4 Workstation. Phiên bản này có thể sử dụng trong văn phòng các doanh nghiệp và các chuyên gia với các tính năng về cấu hình và bảo mật mở rộng. Giao diện Windows 2000 Professional tương tự như Windows 95/98 trên màn hình nhưng hoàn toàn khác về mặt kỹ thuật. Nó hoàn toàn không phải là một sản phẩm kế thừa trực tiếp với Windows 95 hay 98 và không dành để sử dụng tại nhà. Phiên bản Professional sẽ có thể dùng đồng thời hai chip xử lý (nếu được cài đặt).

Windows 2000 Server

Đây là tùy chọn dành cho server cơ sở trong các doanh nghiệp trung bình và nhỏ và kèm thêm một Web server, thiết bị đầu cuối, và các dịch vụ truy cập từ xa. Nó sẽ có thể hỗ trợ tới bốn bộ xử lý được cài đặt đồng thời trên một máy tính.

Windows 2000 Advanced Server

Advanced Server giữ vai trò của "Phiên bản NT Enterprise" với việc có thể hỗ trợ đến tám bộ vi xử lý cài đặt đồng thời. Nó dành để thực hiện các cơ sở dữ liệu chuyên sâu.

Windows 2000 Datacenter Server

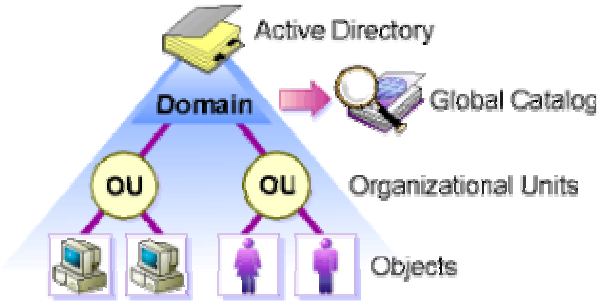
Xuất hiện vào cuối năm 2000 là phiên bản về Windows 2000 rất mạnh đối với các mạng lớn như là công ty, trong các trường đại học, học viện. Có thể sử dụng đồng thời đến 16 bộ xử lý và 64 GB bộ nhớ.

1.2. Activate Directory (AD)

a. Khái niệm

AD là một dịch vụ của Windows 2000 Server, là cơ sở dữ liệu của hệ thống có chức năng nhằm quản lý toàn bộ hệ thống tài nguyên của mạng. Thiết lập, quản lý và cung cấp các dịch vụ mạng cho người dùng.

- Vấn đề bảo mật: AD cung cấp các công cụ nhằm xác minh (authentication) và cấp phép (authorization) cho người dùng mạng để có thể truy cập mạng, đồng thời ngăn chặn những truy cập bất hợp pháp.
  + Duy trì một danh bạ về người dùng và các đối tượng khác trên mạng.
  + Tập trung các danh bạ để hình thành một Server truy nhập.
- Vấn đề tìm kiếm thông tin trên mạng:
  + Tìm kiếm tên các server thông qua tên
  + Tìm kiếm, giải đáp tên qua IP
- Phân cấp quản trị mạng
- Liên lạc và sao chép thông tin trên mạng lớn
- Có thể quản lý một hệ thống mạng rất lớn



b. Các nội dung, đặc điểm của AD

- *Miền (domain)*

Đơn vị bảo mật của Windows 2000 dùng lưu trữ cơ sở dữ liệu người dùng mạng, Server lưu trữ CSDL này được gọi là domain controller, Active Directory server hay có thể được gọi là Logon server. Nhóm các máy phải thông qua máy domain controller để biết thông tin xác minh về người dùng được gọi là một domain (miền), Microsoft cho phép lưu trữ 1,5 triệu tài khoản người dùng trên một domain. Mỗi một máy server W2k có chứa một bản sao tài khoản người dùng mức mạng thì được coi là một domain controller.

- *Nhóm (group)*

Tập hợp một nhóm đối tượng nào đó trên mạng cùng một số quyền, hoặc cùng chịu chung một chính sách quản lý được coi là một nhóm (group), quyền của đối tượng này sẽ được gán gián tiếp thông qua group, đối tượng nào là thành viên của nhóm sẽ có tất cả các quyền của nhóm này. Để áp dụng các chính sách lên nhóm, Windows 2000 cho phép có thể tạo nhóm chứa người dùng, nhóm chứa các máy trên mạng. Các nhóm có thể được lồng vào nhau sâu hơn một cấp, song không được phép đệ qui. Windows 2000 cung cấp bốn loại nhóm:

- Machine local group (nhóm máy cục bộ): nhóm này có chức năng cho máy cục bộ, nó có thể chứa các nhóm miền cục bộ, nhóm toàn miền, nhóm chung.
- Domain local group (nhóm miền cục bộ): nhóm có chức năng cho một miền, nó có thể chứa các nhóm khác miễn là chúng cùng một miền với domain local group này.
- Nhóm toàn miền (global group): nhóm có chức năng cho nhiều miền
- Nhóm chung (universal group): có thể chứa bất kỳ một global hoặc một universal group nào

Để tương thích với tên miền ở phiên bản trước Windows NT, khi cài đặt Windows 2000 để ngầm định chế độ quản lý tên miền hỗn hợp (mixed mode), nó không tạo ra các nhóm universal. Nếu tất cả các máy domain controller trong mạng đều cài windows 2000 thì có thể chuyển sang chế độ native mode và có thể thực hiện được các nhóm universal.

Kích thước các nhóm được windows 2000 giới hạn ở mức 5000 thành viên, nếu muốn tạo nhóm có số thành viên lớn hơn thì cần phải tạo ra một số nhóm nhỏ sau đó đặt số nhóm này vào một nhóm khác.

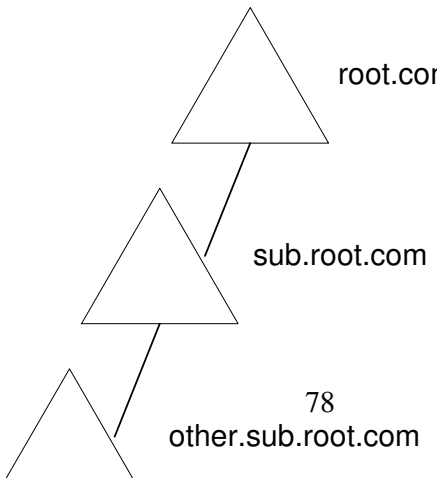
 Đơn vị tổ chức (Organizational unit – OU)

Windows 2000 chia miền thành đơn vị nhỏ hơn đó là OU, OU được tạo ra và có thể được trao quyền kiểm soát gọi là "sự ủy quyền kiểm soát" (delagating control). Một hoặc một tập hợp người dùng có thể được trao quyền kiểm soát này.

 Địa bàn (site)

Si te được AD dùng để quản lý theo giác độ địa lý của mạng, mỗi khu vực được kết nối bằng LAN thì được gọi là một site. Để quản lý windows 2000 cần được cùng cấp thông tin chi tiết về cách bố trí vật lý của mạng, nó có thể tính ra nơi nào có những đường liên kết LAN, WAN. Khi cần sao chép dữ liệu qua lại giữa các domain controller, nó sẽ nén dữ liêu, dùng những thông tin về route để tính ra đường truyền dữ liệu tốt nhất.

 Cây của các miền (tree of domains)



Hệ thống tên miền trong các doanh nghiệp có mạng đa miền được xây dựng theo cấu trúc hình cây. Miền đầu tiên được gọi là gốc (root), các miền dưới nó được gọi là miền con (child domain), dưới nó có thể có một số cấp nữa. Windows 2000 có thể tự động tạo các quan hệ ủy quyền giữa mỗi miền và miền con.

 Rừng của các miền (forest of domains)

Tập hợp các cây (tree) được khớp lại gọi là rừng (forest), các cây trong rừng có cấu trúc giống như một cây, nếu xét theo các mối quan hệ ủy quyền. Có thể quyết định miền đầu tiên được tạo ra sẽ là gốc (root).

1. 3. Địa chỉ trong giao thức TCP/IP

a. Khái niệm

Địa chỉ của nút là địa chỉ thực thể hay địa chỉ của thiết bị trên mạng, để xác định vị trí của nút mạng TCP/IP sử dụng nguyên tắc đánh số duy nhất gọi là địa chỉ IP. Tất cả các thiết bị trên mạng nào sử dụng bộ giao thức TCP/IP đều cần một địa chỉ IP độc nhất. Địa chỉ IP được dùng phổ biến hiện tại gọi là (IPv4 - Internet Protocol Address Version 4), về cấu tạo IPv4 có 32 bit, trên lý thuyết IPv4 có thể cung cấp 2^{32} = 4 294 967 296 địa chỉ. Tuy nhiên với số lượng địa chỉ trên thì chỉ có thể cung cấp cho hệ thống mạng doanh nghiệp. Ngày nay khi Internet đã trở nên phổ biến, tốc độ tăng trưởng của mạng rất cao, IPv4 không đáp ứng được. Nhóm chuyên trách về kỹ thuật của tổ chức Hiệp hội Internet IETF (Internet Engineering Task Force) đã đề xuất và thực hiện địa chỉ thế hệ mới IPv6 (IP Address Version 6). Về cấu tạo IPv6 có 128 bit, trên lý thuyết IPv6 cung cấp 2^{128} = 340 282 366 920 938 463 463 374 607 431 768 211 456 địa chỉ.

b. Xác định địa chỉ IP(v4)

- Cách cấu tạo: (4 bytes)

X.X.X.X

X=0 ÷ 255

Cách nhau dấu chấm (.)

IP chia làm 5 lớp: theo giá trị của X

X	Lớp (class)	Subnet mask
1 - 126	A	255.0.0.0
128 – 191	B	255.255.0.0
192 – 223	C	255.255.255.5
224 - 239	D	Địa chỉ đặc biệt
240 - 247	E	Chưa dùng tới

- Địa chỉ mắt lưới: IP được chia làm 2 phần: địa chỉ mắt lưới (Network portion or netid), địa chỉ máy tính (host portion or hostid).

Lớp	Địa chỉ mắt lưới	Địa chỉ máy tính trong lưới
A 9.67.5.12	X.0.0.0 9.0.0.0	X.X.X 67.5.12
B 150.5.7.9	X.X.0.0 150.5.0.0	X.X 7.9
C 212.15.17.19	X.X.X.0 212.15.17.0	X 19

Với nguyên tắc này các Routers căn cứ vào địa chỉ mắt lưới và sẽ chuyển tiếp bó tin từ mắt lưới này qua mắt lưới khác cho tới khi đúng địa chỉ của mắt lưới đó. Khi bó tin đã tới đúng mắt lưới Routers chuyển bó tin cho máy nhận căn cứ vào phần địa chỉ của máy nhận.

§ 2. CÀI ĐẶT WINDOWS 2000 SERVER

2.1. Một số vấn đề chung

 Hệ thống files trên của MicroSoft

- FAT của DOS, dùng trên DOS, NT, Windows 2000; có thể định dạng khi setup. FAT cho phép truy nhập từ các hệ điều hành khác như Windows NT, Windows 9.X, DOS. Windows NT, W2000 không hỗ trợ chế độ bảo mật trên FAT

- FAT32 dùng trên Windows 98 và các phiên bản tiếp theo, Windows 2000 không hỗ trợ chế độ bảo mật trên FAT32

- NTFS 4 của NT, NTFS 5 của Windows 2000; có thể định dạng khi setup. Các hệ điều hành khác không thể truy nhập. Windows NT, Windows 2000 hỗ trợ chế độ bảo mật trên NTFS, chế độ bảo mật đã được nâng cấp trên NTFS 5.

- HPFS hệ thống files có chế độ thực hiện cao dùng trên NT và OS/2 ; không định dạng khi setup

 Tên SERVER

- Là một dãy ký tự đại diện cho máy tính
- Qui ước đặt tên: Theo qui định của hệ điều hành, ngắn gọn, có qui luật

 Chọn lựa cách kết nối mạng

- Giao thức: TCP/IP, NetBEUI
- Quan hệ thành viên của server
  + Xác định: thành viên của nhóm công tác, thành viên của miền, hay một Domain Controller
  + Có thể không cần quyết định vai trò khi cài đặt, chỉ cần lựa chọn cho máy tham gia một nhóm công tác hay một miền, sau khi cài đặt xong có thể thăng cấp máy thành DC.
- Các thành phần bổ xung
  Dịch vụ khác như Internet Information Server, dịch vụ DHCP, … có thể không cần phải lựa chọn khi cài đặt, những dịch vụ này sẽ được bổ xung khi mạng có nhu cầu

 Vấn đề cấp phép: lựa chọn một trong những cách cấp phép sau

- Cấp phép theo chỗ ngồi (per-seat licensing): Đòi hỏi mỗi máy khách trên mạng khi truy cập vào server đều phải có giấy phép riêng của nó. Cần phải tính trên mạng có bao nhiêu máy khách và khai báo số chổ ngồi. Theo cách này sẽ không cần tính đến có bao nhiêu mối liên kết (connection) đồng thời từ các máy khách vào cùng một server, hoặc mỗi máy khách giữ bao nhiêu mối nối kết với các server.
- Cấp phép theo server (per-server licensing): đòi hỏi mỗi nối kết nối giữa máy khách với server phải có một giấy phép, nếu một máy khách nối với 5 server khác nhau thì sẽ cần 5 giấy phép.

2.2. Cài đặt windows 2000 server

a. Yêu cầu cơ bản

 Yêu cầu về phần cứng:Tối thiểu

- Pentium 166
- 64 MB RAM
- HD 850 MB thêm 100 MB ứng với 64 MB RAM bộ nhớ bị thiếu
- Các phần cứng khác phải tương thích với Windows 2000, danh sách phần cứng được ghi trên đĩa CD cài đặt và trên trang Web www.microsoft.com/hwtest/hcl
- Card mạng, dây mạng đã được lắp đặt, lưu ý về kiểu card mạng (ISA, PCI), cấu hình card mạng như IRQ, địa chỉ I/O, những thông số khác về card mạng.
- Các thiết bị ngoại vi khác như máy in: kiểu, cổng máy in; MODEM ...

- Loại vi xử lý (Intel, Alpha, ...) thư mục I386 chứa phần SETUP cho Intel và tương thích. Khi cài đặt chạy SETUP ở thư mục thích hợp, hoặc bộ đĩa cài đặt được dùng cho bộ vi xử lý phù hợp.

 Phần mềm:

- Đĩa cài đặt từ CD ROM
- Các Driver cho thiết bị

 Chuẩn bị

- Thiết lập cấu hình trong BIOS của máy tính: nếu để chế độ PnP, thì cần dành IRQ cho các thết bị cũ theo chuẩn ISA như card mạng, card âm thanh.
- Chuẩn bị đĩa cứng:

+ Nếu dùng toàn bộ đĩa cứng cho Windows 2000: không cần phân chia Partition, định dạng (format) trước.

+ Nếu chỉ dùng 1 phần của đĩa cứng: cần phân chia Partition, có thể định dạng (format) trước.

b. Một số phương pháp cài đặt:

 Cài đặt từ đĩa cứng

- Giai đoạn 1: chuẩn bị và bắt đầu cài đặt

Chuẩn bị đĩa cứng: Tạo trên đĩa cứng 2 Primary Partition (FAT16 hoặc FAT32) trong đó có 1 Partition có thể khởi động được từ DOS (ổ đĩa C:), Partition thứ hai có thể sử dụng để ghi chép các tệp tin cài đặt (ổ đĩa D:). Sao chép toàn bộ thư mục I386 ở đĩa cài đặt vào ổ đĩa D:

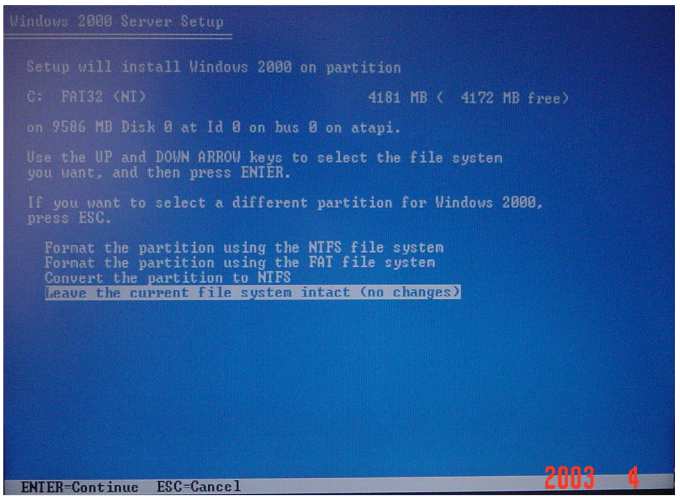
Sau khi máy tính đã khởi động từ DOS, chạy tệp Winnt.exe trong thư mục I386, trên màn hình hiện thị hộp hội thoại để xác định đường dẫn của tệp các tin cài đặt (Enter the path where Windows 2000 files are to be found), nếu trên màn hình đã hiển thị đúng đường dẫn , bần Enter để tiếp tục.

Chương trình setup của Windows bắt đầu thực hiện việc cài đặt: sao chép các tệp tin cần thiết lên đĩa cứng, trên màn hình hiển thị quá trình sao chép tệp tin (Setup is copying files …), sau khi sao chép xong máy tính yêu cầu khởi động lại.

- Giai đoạn 2

Sau khi khởi động lại chương trình setup tìm kiếm các thiết bị trên máy , qua giai đoạn này máy dừng lại để yêu cầu xác định: cài đặt Windows 2000 từ đầu hoặc cài đặt kiểu sữa lỗi (repair) nếu trên đĩa cứng đã cài Windows 2000. Bấn Enter để tiếp tục cài đặt, trên màn hình hiển thị các thông tin về bản quyền (Windows 2000 Licensing Agreement), bấm F8 để tiếp tục. Tiếp theo trên màn hình hiển thị tất cả các Partition của đĩa cứng mà có thể cài đặt Windows 2000,

cần xác định các Partition sẽ cài đặt. Sau khi chọn Partition tiếp tục lựa chọn cấu trúc hệ thống files:



+ Convert the partition to NTFS: Chuyển cấu trúc tệp tin sang dạng NTFS

+ Leave the current file system intact (no changes): Giữ nguyên cấu trúc tệp tin cũ

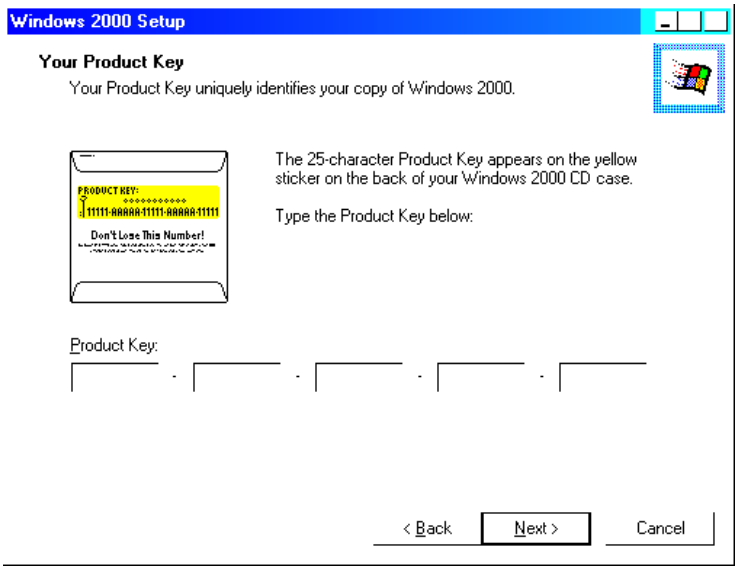
Thực hiện một trong hai lựa chọn trên, máy sẽ khởi động lại và chuyển sang giai đoạn 3, giai đoạn màn hình Graphic.

- Giai đoạn 3

Sau khi khởi động lại, trình setup thực hiện việc xác định và cài đặt thiết bị trên máy (Installing devices), công việc này diễn ra khá lâu nếu gặp các thiết bị mà Windows 2000 không thể cài đặt được có thể dẫn đến treo máy, kết thúc quá trình cài đặt mà không thành công. Tiếp theo thực hiện các lựa chọn về ngôn ngữ, bàn phím sử dụng (Regional settings), bấm Next để tiếp tục. Các lựa chọn tiếp theo:

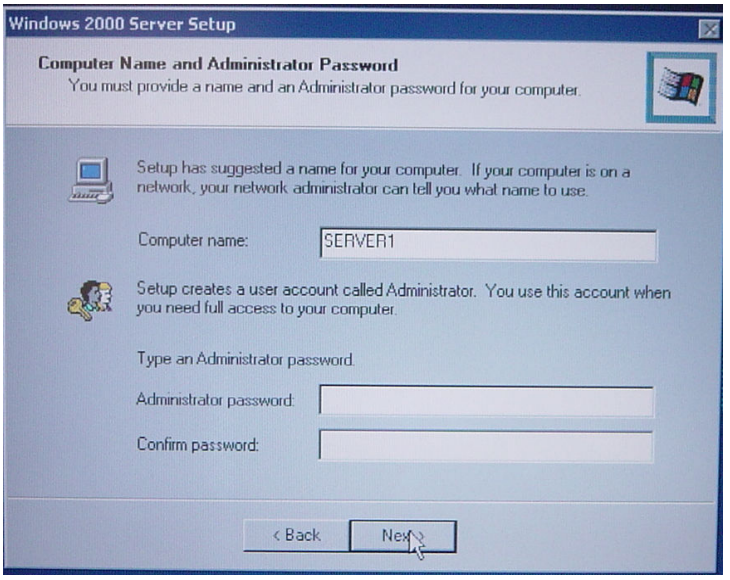
+ Xác định tên người và tổ chức (cơ quan) sử dụng hệ điều hành.

+ Xác định Product key: bấm chính xác các ký tự vào các ô theo mẫu sau, sau đó chọn Next



+ Xác định kiểu cấp phép (licensing) có hai kiểu để có thể lựa chọn per-server và per-seat

+ Xác định tên máy tính (Computer name) và mật khẩu của user quản trị mạng (Administrator)



+ Xác định các thành phần của Windows (Add or remove components of windows 2000), nếu để các tham số ngầm định, bấm Next.

+ Xác định ngày, giơ , múi giờ quốc tế của hệ thống

+ Cài đặt các thông số mạng (Networking settings): có hai lựa chọn Typical setting hoặc Custom settings, bấm Next. Tiếp theo xác định các tham số trên Workgroup or computer domain. Chọn "NO, … " và bấm trên ô workgroup một dãy ký tự để có thể làm tên cho một workgroup, lựa chọn này để có thể cấu

hình máy thành máy Server. Nếu chọn "YES, … " cần phải điền tên domain đã có ở trên mạng, và làm các bước tiếp theo theo hướng dẫn trên màn hình.

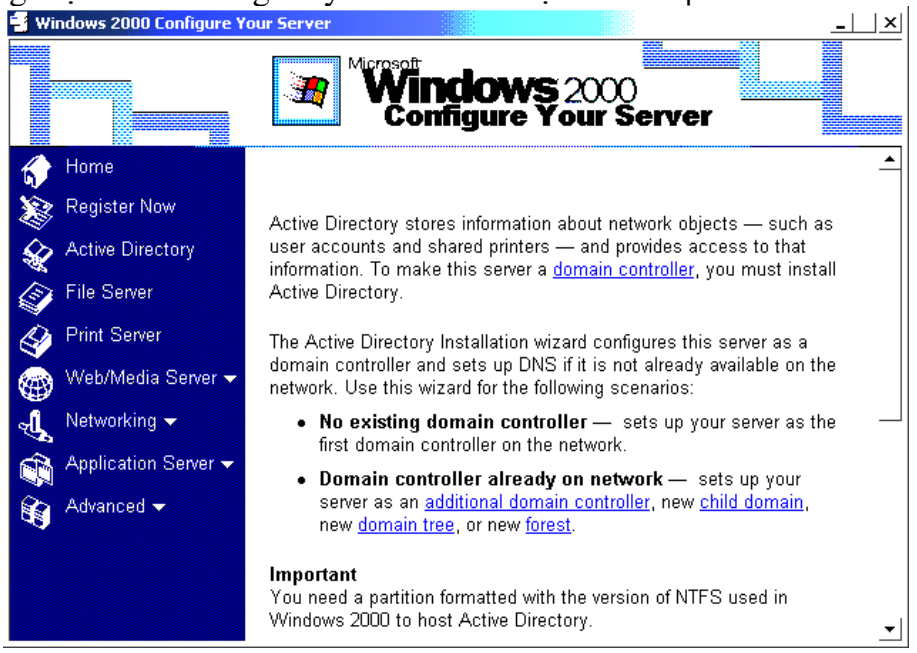
Trình Setup tiếp tục thực hiện quá trình cài đặt, nếu thiết bị phần cứng không có lỗi và nếu các thiết bị đó được Windows 2000 chấp nhận, thì quá trình cài đặt thành công, máy sẽ khởi động lại.

 Cài đặt từ bộ đĩa CD ROM Windows 2000

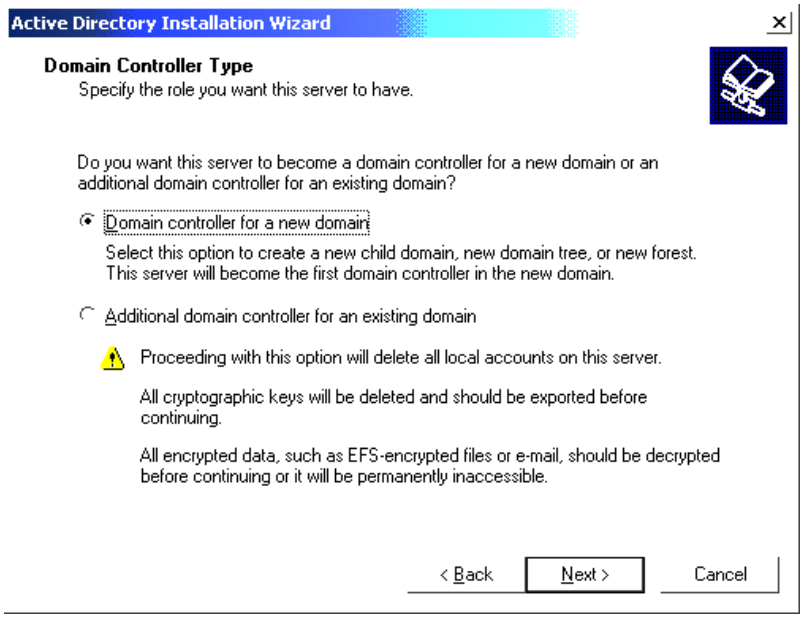
- Yêu cầu:
  + Máy tính có ổ đĩa CD ROM, có thể khởi động được từ ổ CD
  + Đĩa CD ROOM cài đặt Windows 2000.
- Cài đặt
  + Thiết lập chế độ khởi động máy tính từ ổ đĩa CD
  + Đưa đĩa CD ROOM cài đặt Windows 2000, khởi động máy
  + Nhập các tham số cần thiết

2.3. Cài đặt Activate Directory

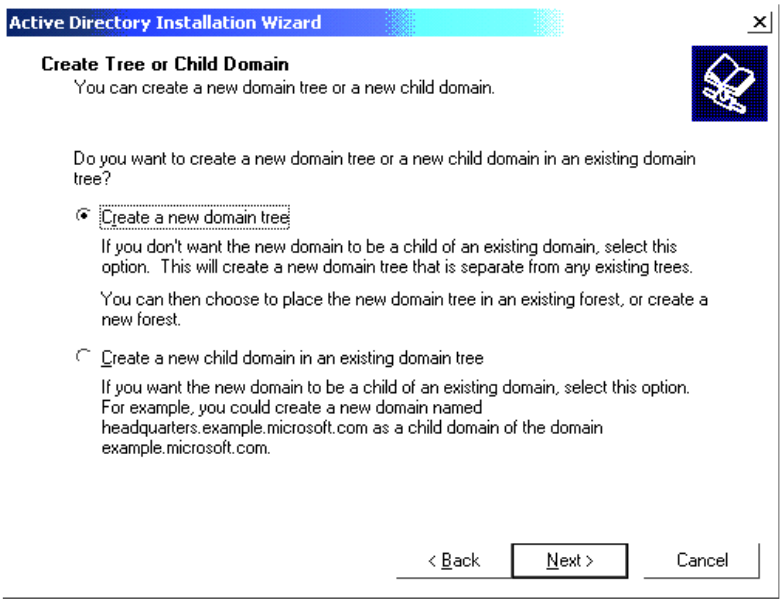
Sử dụng thực đơn "configure your server" hoặc trình dcpromo.exe



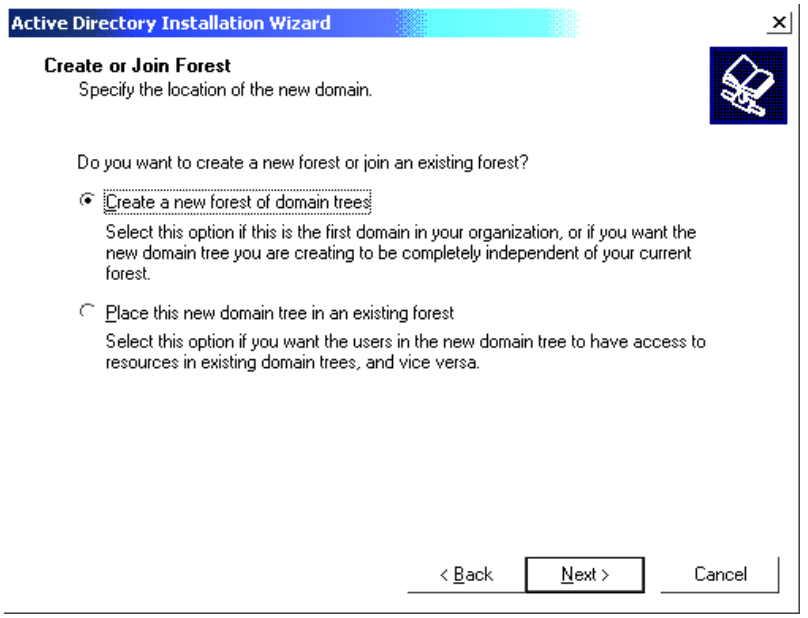
Lựa chọn tạo miền mới



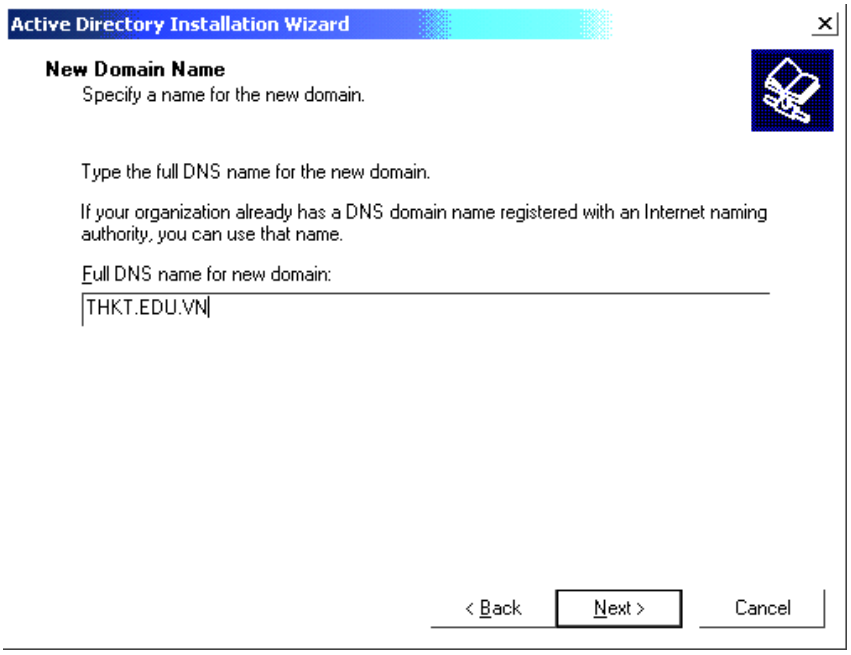
Mỗi miền phải thuộc về một cây nào đó, đây là miền đầu tiên trong một cây mới chọn "create a new domain tree"



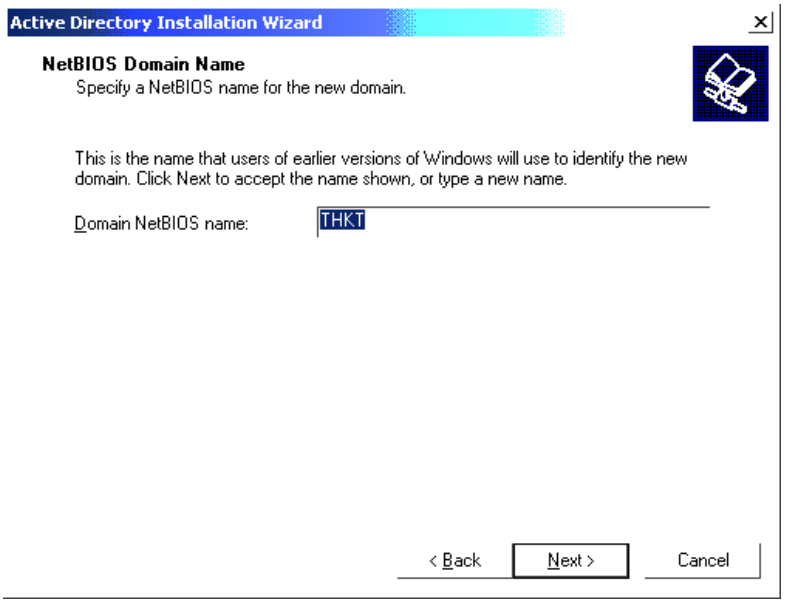
Xác định cây mới này trong rừng mới, chọn "create a new forest of domain tree"



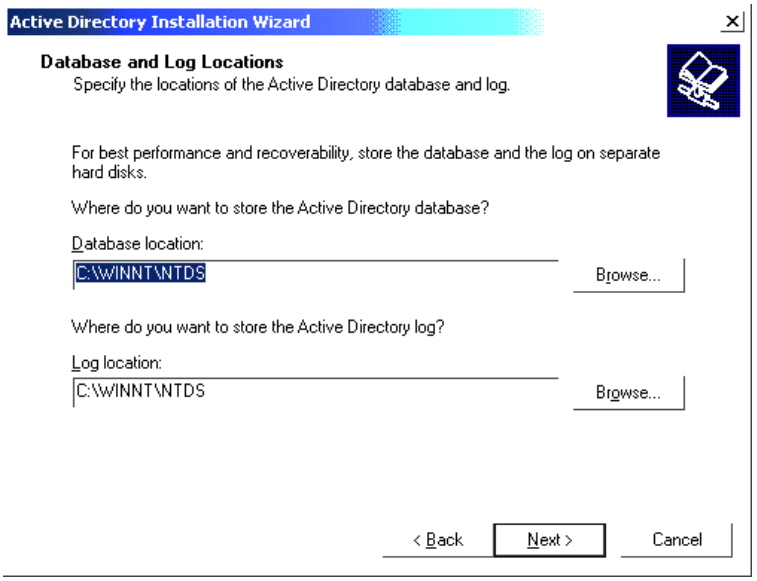
Xác định tên đầy đủ của miền mới



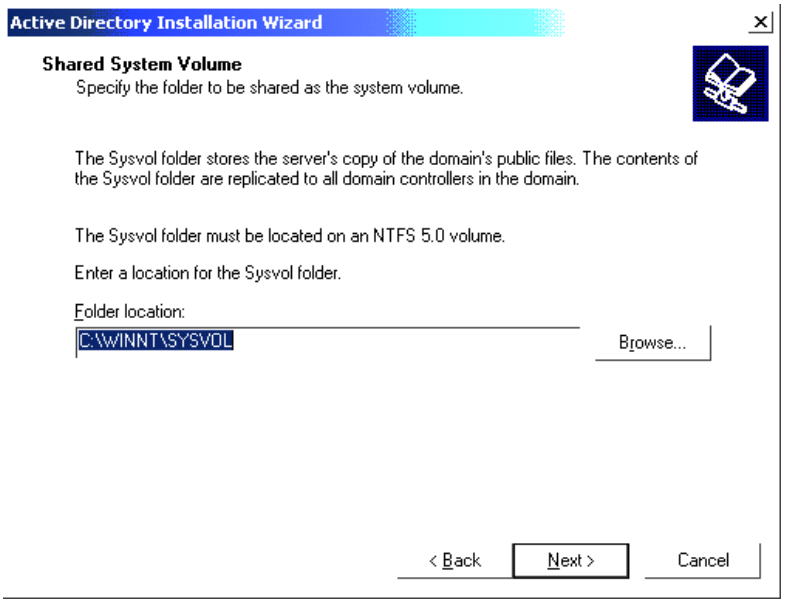
Tên kiểu cũ sẽ được xác lập



Chọn chỗ đặt tệp tin cơ sở dữ liệu AD và tệp tin ghi chép giao dịch, chọn theo chế độ ngầm định của Windows 2000

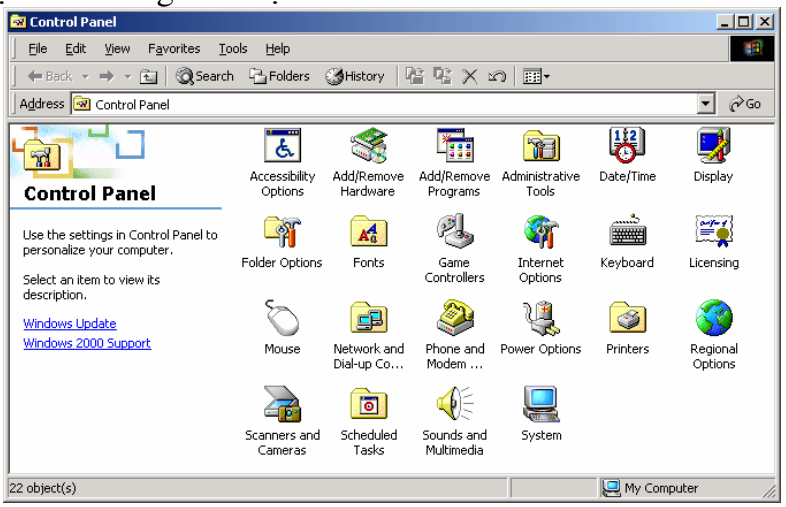


Chọn chỗ đặt folder SYSVOL



2.4. Hiệu chỉnh các tham số

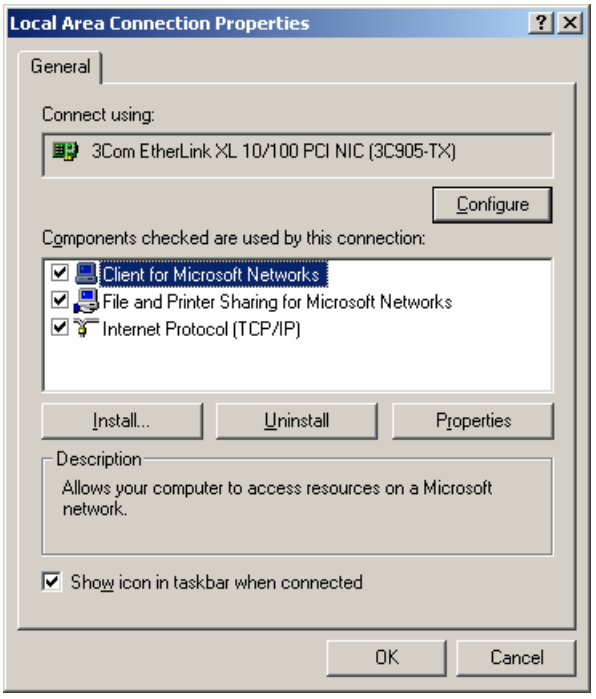
Sử dụng CONTROL PANEL , đây là một nhóm phần mềm cung cấp các công cụ cho người quản trị mạng, có thể hiệu chỉnh các tham số liên quan về phần cứng phần mềm. Mỗi một biểu tượng trên cửa sổ CONTROL PANEL sẽ đảm nhiệm một chức năng nhất định.



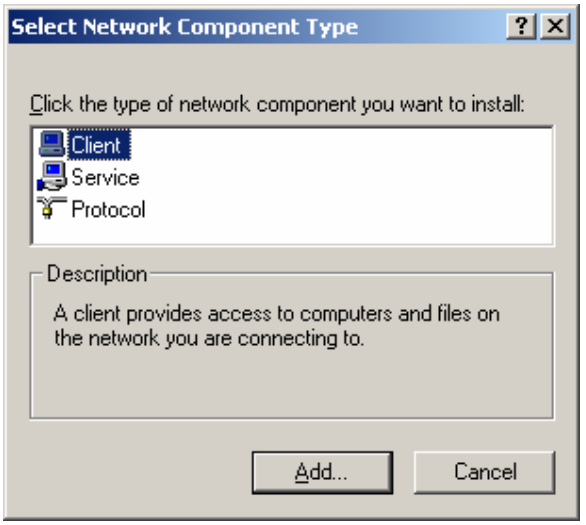
a. Hiệu chỉnh các tham số về mạng

Bao gồm các công việc

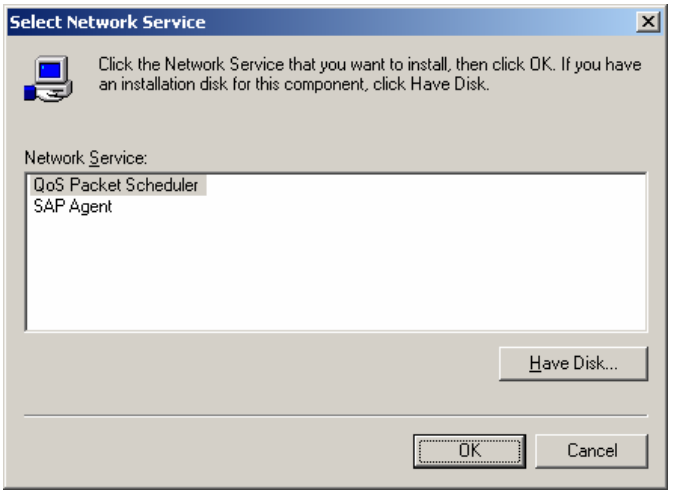
Chọn biểu tượng NETWORK and Dial-up, tiếp theo bấm phải chuột vào dòng Local Area Connection và chọn Properties, trên màn hình hiển thị cửa sổ sau:



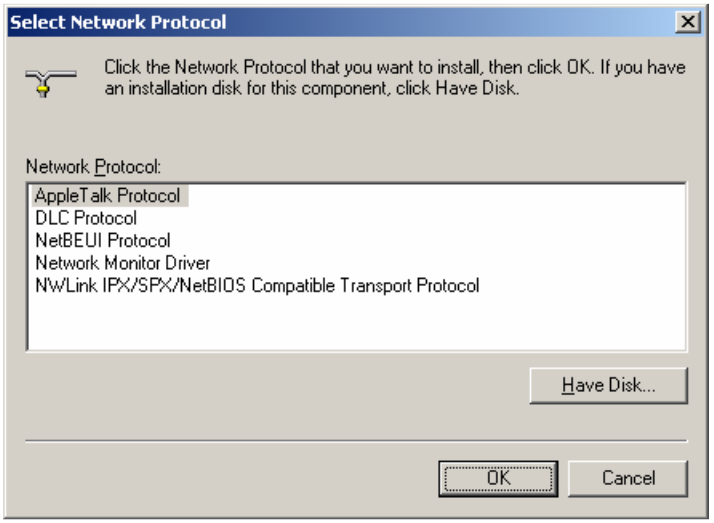
Chọn Install để thêm hoặc loại bỏ các dịch vụ (Service), các bộ giao thức (Protocol), các client cho Server. Chọn Uninstall để gỡ bỏ các dịch vụ, giao thức và client. Chọn Properties để điền các tham số chi tiết.



Các dịch vụ mạng Windows 2000 Server cho phép các máy tính trong mạng truy cập và sử dụng nguồn tài nguyên của mạng.



Tùy từng loại giao thức, người quản trị mạng cần chọn Properties để điền, hiệu chỉnh các thông tin chi tiết.



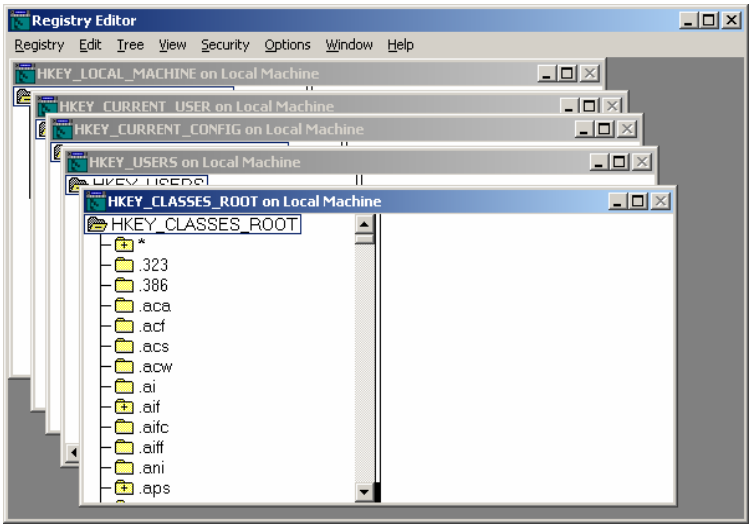
b. Hiệu chỉnh các tham số khác

2.5. Cơ sở dữ liệu Registry

a. Khái niệm

Registry của Windows 200 là một có sở dữ liệu có cấp bậc của những thiết định dùng để mô tả tài khoản người dùng, phần cứng của máy, và các ứng dụng. Mỗi khi thay đổi một lựa chọn về phần cứng hay phần mềm, thì những thay đổi này thường được ghi lại trong registry. Registry còn lưu nhiều thiết định của hệ thống ví dụ như khi khởi động máy Windows 2000 sẽ tự kiểm tra những thiết bị phần cứng được gắn vào máy. Có thể xem và sửa registry bằng trình REGEDT32.EXE (nằm trong thư mục \winnt\system32).

b. Nội dung Registry



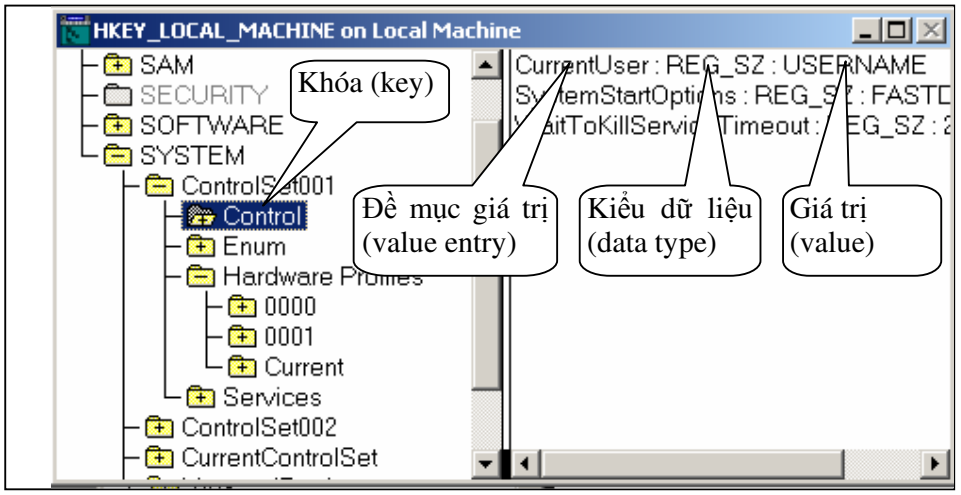
- *Cây con (subtree)*

Registry lưu trữ tất cả những thông tin bằng cách chia chúng ra làm năm cây con:

- HKEY_LOCAL_MACHINE: Chứa thông tin về phần cứng, những thiết định dành cho các hệ thống đang chạy trên máy.
- HKEY_CURENT_USER: Chứa thông tin về người dùng (user profile) được truy cập server, chứa các cấu hình, các thiết định dành cho các desktop đang chạy trên server.
- HKEY_USER: Chứa con trỏ (poiter) chỉ đến cây con HKEY_CURENT_USER và đến một default profile. Default profile chứa các thiết định ngầm định.
- HKEY_CLASSES_ROOT: Giữ các thông tin về file association , giúp hệ điều hành biết cách mở tin bằng chương trình nào.

- *Khóa (key), đề mục giá trị,tên, giá trị, kiểu dữ liệu*

Khóa là một đề mục có tác dụng chia tách thông tin ra thành các đoạn, có thể có các khóa bên trong các khóa, gọi là các khóa con (subkey), khóa con có thể có nhiều cấp. Khóa có tên, tên khóa được viết liền và thường được đặt ngắn gọn để có thể viết vừa một hàng chữ.



Dữ liệu trong registry được qui định như sau:

- REG_BINARY: dữ liệu nhị phân thô
- REG_DWORD: Kiểu nhị phân khác dài 4 bytes
- REG_EXPAND_SZ: chuỗi ký tự có kích thước thay đổi
- REG_MULTI_SZ: một kiểu dữ liệu chuỗi khác, cho phép nhập thêm tham số
- REG_SZ: chuỗi đơn giản

 Các tổ của registry (hive)

Registry chủ yếu được chứa trong các tệp tin gọi là tổ (hive), các tổ là các tệp nhị phân do vậy muốn đọc nó cần có trình soạn thảo phù hợp. Việc chia các tổ nhằm nạp, sao lưu dự phòng một cách thuận tiện. Các tệp tin tổ được tổ chức như sau:

Cây/khóa	Tệp tin
HKEY_LOCAL_MACHINE\SAM	SAM (chính) và SAM.LOG (dự phòng)
HKEY_LOCAL_MACHINE\SECURITY	SECURITY (chính) SECURITY.LOG (dự phòng)
HKEY_LOCAL_MACHINE\SOFTWARE	SOFTWARE (chính) SOFTWARE.LOG (dự phòng)
HKEY_LOCAL_MACHINE\SYSTEM	SYSTEM (chính) SYSTEM. ALT (dự phòng)
HKEY_USER\DEFAULT	DEFAULT (chính) DEFAULT. LOG (dự phòng)
HKEY_USER\SECURITY ID	NTUSER.DAT
HKEY_CURENT_USER	NTUSER.DAT
HKEY_CLASSES_ROOT	Được tạo lúc boot máy

 Lưu dự phòng và khôi phục

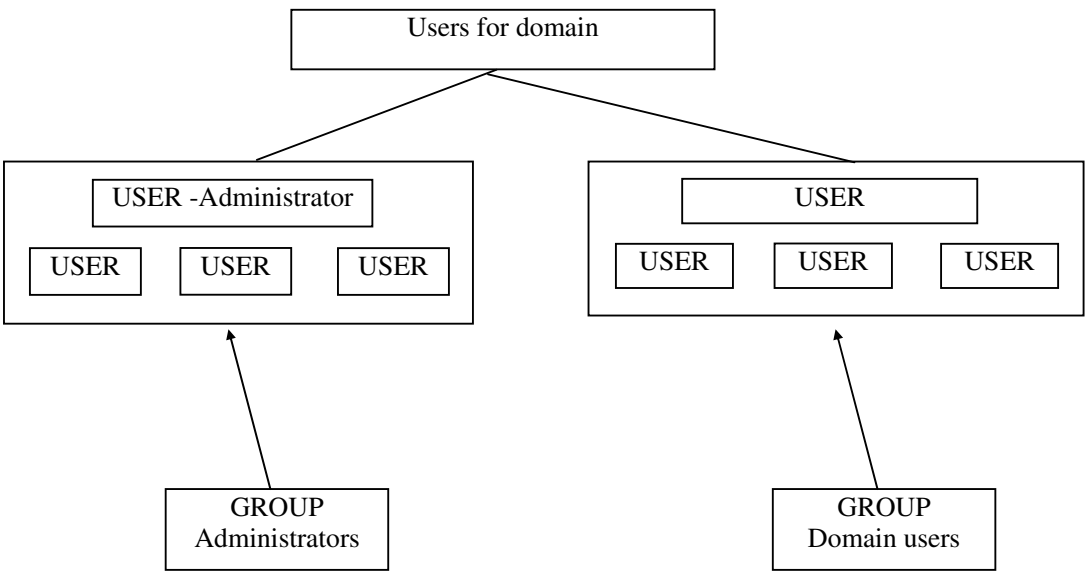
Mỗi khi tệp tin tổ bị thay đổi, trước tiên nó sẽ ghi vào các tệp tin dự phòng (LOG) của nó, khi sự mô tả về những thay đổi đối với tệp tin tổ đã hoàn tất, nó mới thực sự ghi chính tệp tin tổ. Nếu hệ thống gặp sự cố trong

thao tác ghi vào tệp tin tổ, thì hệ thống có thể lấy thông tin từ tệp dự phòng để khôi phục tệp tin tổ trở lại tình trạng trước đó. Trong một số trường hợp registry bị lỗi thì có khả năng gây lỗi cho Windows 2000, do vây cần phải sao lưu registry. Để sao lưu toàn bộ registry có thể dùng trình tiện ích ntbackup.exe (trong thư mục system32). để sao lưu một phần của registry có thể dùng luôn trình REGEDT32.EXE

§ 3. QUẢN TRỊ USER VÀ GROUP

3.1. Khái niệm user và group

User là một định danh, thông qua định danh này người dùng có thể truy cập và sử dụng các dịch vụ mạng, user có thể gọi là "tài khoản người dùng". Windows 2000 Server có khả năng cung cấp một hệ thống user rất lớn và được quản lý bởi AD. Các tài khoản người dùng khi được tạo ra đều tự động được cấp cho một mã nhận diện bảo mật (security indentifier – SID) Mỗi SID là một con số duy nhất để nhận diện một tài khoản, các SID không bao giờ được tái sử dụng, khi một tài khoản bị xóa, nó cũng bị xóa theo. Khi có tài khoản mới sẽ được cấp một SID mới không trùng lặp, nhờ đó tài khoản mới sẽ không có quyền hạn và quyền truy cập trùng với tài khoản cũ và hệ thống vẫn giữ được tính bảo mật. Hệ thống tài khoản được tổ chức theo mô hình:

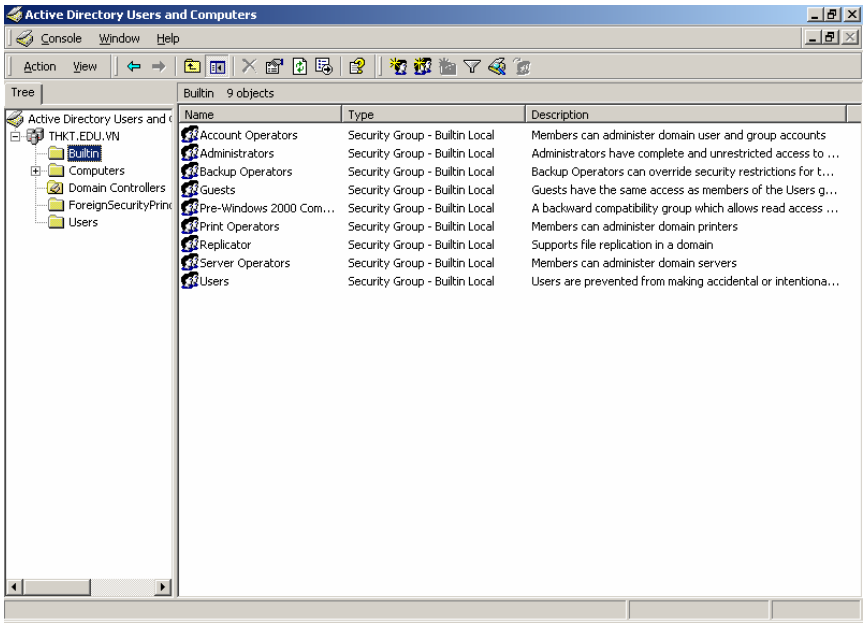


Sau khi cài đặt Windows 2000 tạo ra một user đặc biệt có tên Administrator, tài khoản này có toàn bộ quyền hạn trên một máy hay một miền, không thể xóa tài khoản này nhưng có thể đổi tên.

Một số user có cùng quyền hạn trên mạng có thể được xếp vào một nhóm gọi là group, group được đặt tên (group name) . Một user có thể là thành viên của

nhiều group, khi là thành viên của group nào thì user sẽ có quyền hạn của nhóm đó. Windows 2000 server cung cấp hai loại nhóm, nhóm bảo mật (security group) và nhóm phân phối (distribution group). Nhóm bảo mật là nhóm được dùng để cấp các quyền hạn và quyền truy cập, mỗi nhóm bảo mật đều được cấp một mã nhận diện bảo mật (SID). Có ba kiểu nhóm bảo mật là nhóm bảo mật tại chỗ (local group), nhóm toàn miền (global group), nhóm chung (universal group). Nhóm phân phối được tạo ra không vì mục đích bảo mật, nó không có (SID) , nó có thể được dùng để cấp phát thư tín, dùng để thông báo cho một nhóm người. Windows 2000 server cung cấp các nhóm tạo sẵn như sau:

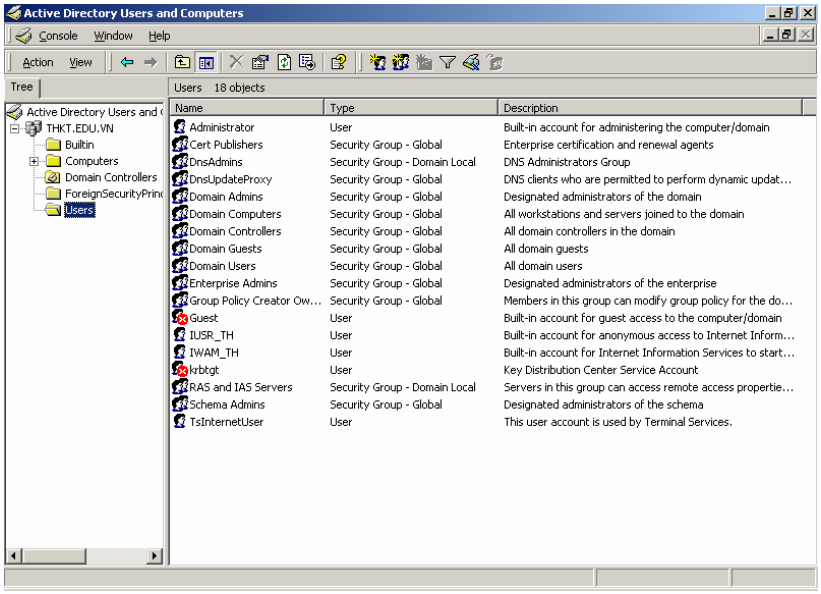
- *Các domain local group trong container builtin*



Quyền hạn các người dùng của nhóm ngầm định	**Họ cũng có thể**
Administrators: Truy nhập tại chỗ Truy cập máy này từ mạng Chiếm quyền sở hữu các tập tin Quản lý bản ghi chép kiểm toán và bảo mật Thay đổi giờ giấc của máy Tắt máy Buộc tắt máy này từ một máy ở xa Lưu dự phòng các tập tin và thư mục Khôi phục lại các tập tin và thư mục Thêm và bớt các device driver Tăng độ ưu tiên của một quá trình xử	Tạo ra và quản lý các tài khoản người dùng Tạo ra và quản lý các global group Trao quyền hạn cho người dùng Quản lý chính sách kiểm toán và bảo mật Khoá chặt server console Mở khoá server console Định dạng đĩa cứng của server Tạo ra các nhóm chương trình chung Giữ riêng một profile tại chỗ Chia sẻ và chấm dứt chia sẻ các thư mục Chia sẻ và chấm dứt chia sẻ các tài

lý	nguyên khác
Server Operators Truy nhập tại chỗ Thay đổi giờ giấc của máy server này Tắt máy server này Buộc tắt máy server này từ một máy ở xa Lưu dự phòng các tập tin và thư mục Khôi phục lại các tập tin và thư mục	Khoá chặt server Phủ quyết khoá của server Định dạng đĩa cứng của server Tạo các nhóm chung Giữ riêng một Profile tại chỗ Chia sẻ và chấm dứt chia sẻ các thư mục Chia sẻ và chấm dứt chia sẻ các máy in
Account Operators Truy nhập tại chỗ Tắt máy server này	Tạo ra và quản lý các người dùng, các global, các local group, không có quyền sửa đổi tài khoản của Administrator, global group Domain Admins hoặc các local group Administrator, Server Operators và Backup Operators Giữ riêng một Profile tại chỗ
Print Operators Truy nhập tại chỗ Tắt máy	Giữ riêng một Profile tại chỗ Chia sẻ và chấm dứt chia sẻ các máy in
Backup Operators Truy nhập tại chỗ Tắt máy Lưu dự phòng các tập tin và thư mục Khôi phục lại các tập tin và thư mục	Giữ riêng một Profile tại chỗ
Everyone Truy cập máy này từ mạng	Khoá chặt server – nếu có quyền truy nhập tại chỗ
Users (Không có quyền gì)	Tạo ra và quản lý các local group- nếu có quyền truy nhập tại chỗ
Guets (Không có quyền gì)	(Không có quyền gì)
Replicator (Không có quyền gì)	(Không có quyền gì)

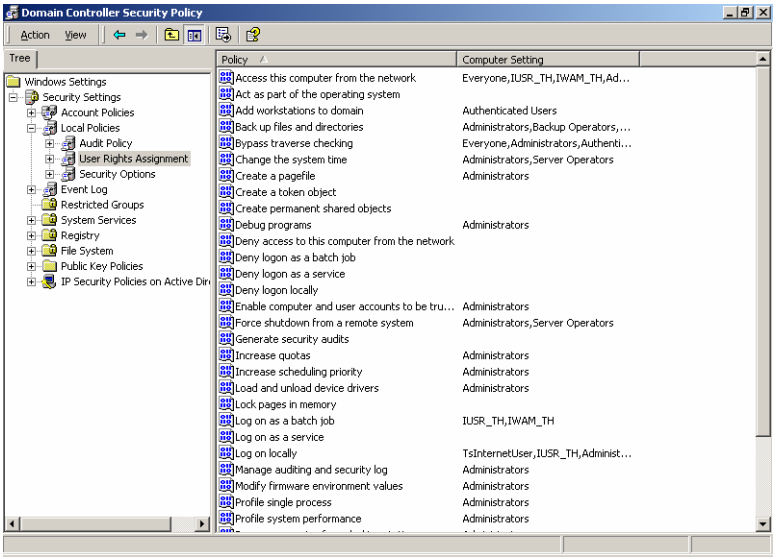
- *Các domain global group trong container users*



Nhóm	Công dụng của nó
Domain Admins	Bằng cách đặt một tài khoản người dùng vào trong global group này, bạn cung cấp được các năng lực ở mức độ quản trị cho người dùng đó. Các thành viên của Domain Admins của một miền có thể quản trị miền nhà, các máy trạm của miền ấy và mọi miền được uỷ quyền khác nếu đã lồng global group Domain Admins của miền ấy vào các Local Administrators của chúng. Theo mặc định global group Domain Admins được tạo sẵn của một miền sẽ là một thành viên của cả Local Administrators của miền ấy lẫn các Local group Administrators của mọi máy trạm NT hoặc Win2K Pro trong miền ấy. Global group Domain Admins của một miền sẽ tự động có một thành viên là tài khoản Administrators được tạo sẵn của miền ấy.
Domain Users	Các thành viên của global group Domain Users của một miền có quyền truy cập và quyền hành của người dùng bình thường đối với cả miền ấy lẫn mọi máy trạm NT/ Win2K trong miền ấy. Nhóm này chứa tất cả các tài khoản của người dùng của miền ấy, và theo mặc định, là một thành viên của mọi Local group Users trên mọi máy trạm NT/ Win2K trong miền ấy.
Domain Guests	Miền này cho phép các tài khoản khách vãng lai (Guests) truy cập được các tài nguyên ngang qua ranh giới miền nếu họ đã được các quản trị viên miền này cho phép làm như thế

3.2. Khái niệm quyền hạn

Windows 2000 cung cấp cho người dùng khả năng truy cập mạng để quản lý hoặc sử dụng tài nguyên. Để kiểm soát truy cập windows 2000 chia ra hai nhóm quyền, thứ nhất đó là quyền hạn (right) đối tượng nào có quyền hạn thì có khả năng gán hoặc hạn chế khả năng truy cập vào một vài đối tượng hệ thống. Thứ hai là quyền truy cập (permission) hay giấy phép truy cập, đối tượng nào có quyền này thì được phép sử dụng đối với tài nguyên của mạng. Theo nguyên tắc, các quyền hạn được ưu tiên hơn quyền truy cập, các quyền hạn thường được gán cho những công việc liên quan đến việc quản lý hệ thống. Windows 2000 đã gán ngầm định các quyền hạn cho các nhóm tạo sẵn, ngoài ra quản trị mạng có thể tạo nhóm mới rồi gán quyền hạn phù hợp cho nhóm này. Các quyền hạn được tạo sẵn bao gồm:



Quyền hạn người dùng	Giải thích ý nghĩa
Access this computer from the network	Nối kết vào máy này ngang qua mạng
Act as part of the Operation system	Đóng vai trò như một phần được uỷ quyền của hệ điều hành; một số tiểu hệ thống được cấp quyền hạn này.
Add wortation to domain	Làm cho các máy trạm trở thành viên của miền
Back up files and directoies	Lưu dự phòng các tập tin và thư mục. Như đã nói ở trên quyền này đã phủ quyết các quyền truy cập tập tin và thư mục.
Bypass traverse checking	Duyệt lưới qua một cây thư mục, cho dù người dùng đó không có

	quyền truy cập nào đó với thư mục đó.
Change the system time	ấn định giờ giấc đồng hồ bên trong máy tại chỗ.
Create a pagefile	Tạo một tập tin phân trang (bộ nhớ ảo)
Create a token object	Tạo các thẻ hiệu truy cập (access token) chỉ bộ phận Local security Authority mới có quyền này.
Create permanent shared objects	Tạo những đối tượng vĩnh viễn đặc biệt
Debug programs	Gỡ rối các ứng dụng
Deny access to this computer from the network	Ngược lại với quyền Access this computer from the network; thu hồi riêng quyền này đối với những người hay nhóm mà bình thường vẫn có nó.
Deny logon as batch job	Thu hồi quyền Log on as a batch job
Deny logon as a service	Thu hồi quyền Log on as a service
Deny logon locally	Thu hồi quyền Log on locally
Enable computer and user accounts to be trussted for delegation	Chỉ định các tài khoản có thể được uỷ quyền
Force shutdown from a remote system	Quyền shutdown máy từ xa
Generate security audits	Tạo ra các đề mục ghi chép kiểm toán
Increase quotas	Tăng các hạn ngạch về dung lương đĩa cứng
Increase scheduling priority	Tăng cường độ ưu tiên lịch biểu của một quá trình xử lý.
Load and unload device drivers	Nạp (hoặc bớt) driver vào (hoặc ra khỏi) hệ thống.
Lock pages in memory	Khoá chặt các trang vào trong bộ nhớ để ngăn không cho chúng bị đưa vào trong bộ lưu trữ dự phòng(như Pagefile. Sys chẳng hạn)
Log on as a batch job	Đăng nhập vào hệ thống như một phương tiện hàng đợi theo lô (Batch queue facility)
Log on as a service	Thực hiện các dịch vụ bảo mật (vd; người dùng mà thực hiện việc sao chép sẽ đăng nhập với tính cách

	như một dịch vụ)
Log on locally	Đăng nhập tại chỗ, tại chính máy server này.
Manager auditingand security log	Chỉ rõ những sự kiện và kiểu truy cập tài nguyên gì sẽ được kiểm toán. Ngoài ra còn cho phép xem và xoá sạch bản ghi chép bảo mật (security log)
Modify firmware envionment valuas	Sửa đổi các biến môi trường của hệ thống (không phải biến môi trường của người dùng)
Profile single process	Sử dụng những khả năng ghi chép hoạt động(profiling) của Win2K để quan sát, nhận xét hoạt động của một quá trình xử lý.
Profile system perfromance	Sử dụng các khả năng ghi chép hoạt động của Win2K để quan sát nhận xét hoạt động của hệ thống.
Remove computer from docking station	Tháo gỡ một máy laptop ra khỏi hộp nối ghép vào mạng (docking station) của nó.
Replace a process level token	Sửa đổi thẻ hiệu truy cập của một quá trình.
Restore files and directories	Khôi phục lại các tập tin và thư mục, quyền này phủ quyết các quyền truy cập tập tin và thư mục.
Shut down the system	Tắt máy Win2K
Synchronize directory service data	Cập nhật thông tin của Active directory
Take ownership of files or other objects	Chiếm quyền sở hữu của các tập tin, thư mục, và các đối tượng khác, vốn trước đó được các người dùng khác sở hữu.

- *Quyền truy cập (permission)*

Quyền tuy cập hay giấy phép truy cập (permission) chỉ định user nào, group nào được phép sử dụng tài nguyên và với điều kiện gì.

3.3. Thiết kế hệ thống user và group

Trước khi tạo hệ thống user và group cho một mạng máy tính cần thiết kế hệ thống user cho tổ chức đó. Thiết kế này là cơ sở để thực hiện các tác vụ cụ thể liên quan đến user, nó cũng là cơ sở để quản lý hệ thống user trong quá trình hoạt động mạng. Khi thiết kế cần tuân thủ một số nguyên tắc sau:

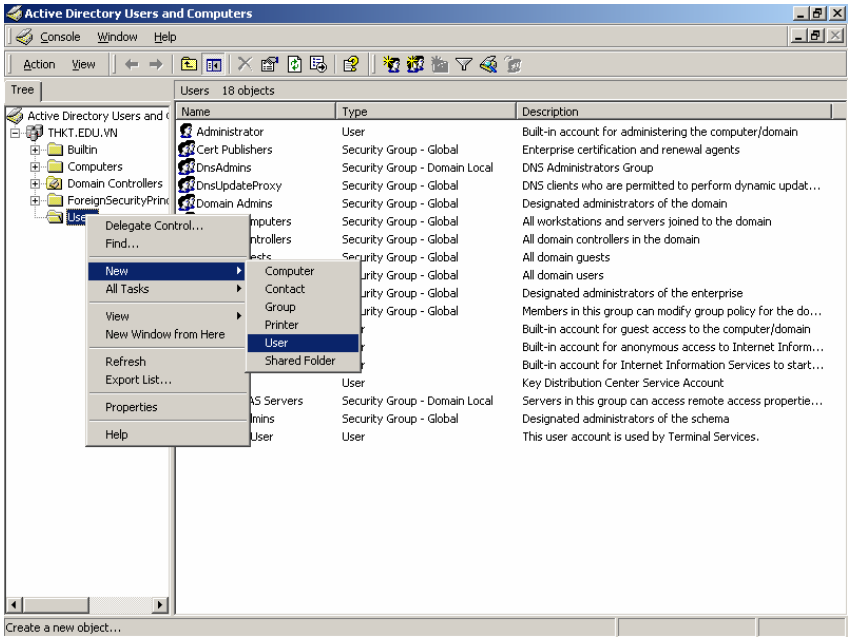
- Tên phải là tên duy nhất, tên user không được trùng tên nhóm.
- Độ dài của tên dài nhất đến 20 ký tự, có thể chứa chữ hoa, chữ thường hoặc cả hai.
- Tên không được chứa các ký tự: " / \ : | + * ? < >' ; các ký tự mở rộng.
- Phân loại user quản trị mạng và user sử dụng tài nguyên mạng
- Sắp đặt các user vào các group phù hợp
- Lập bảng thiết kế users

3.4. Tạo user và group

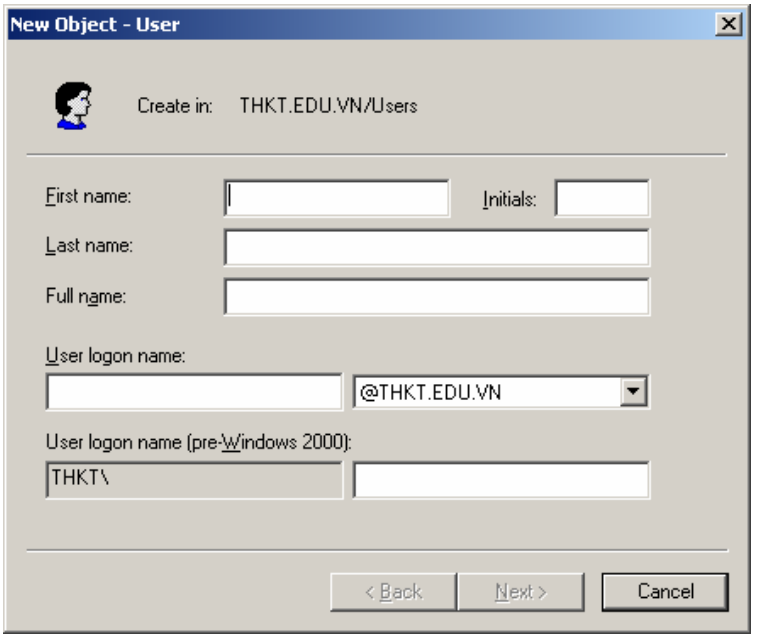
a. USER

 Tạo user

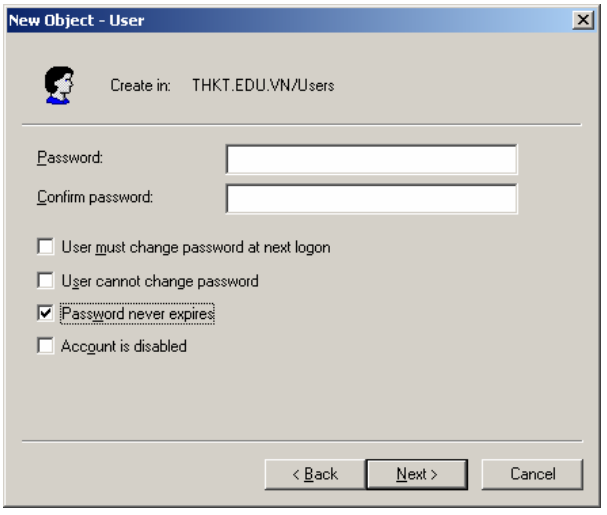
- Chạy trình Activate Directory Users and Computer, bấm phải chuột vào OU users, chọn new – user. Trên màn hình sẽ hiển thị cửa sổ NEW OBJECT – USER



- Trên cửa sổ này cần điền các tham số

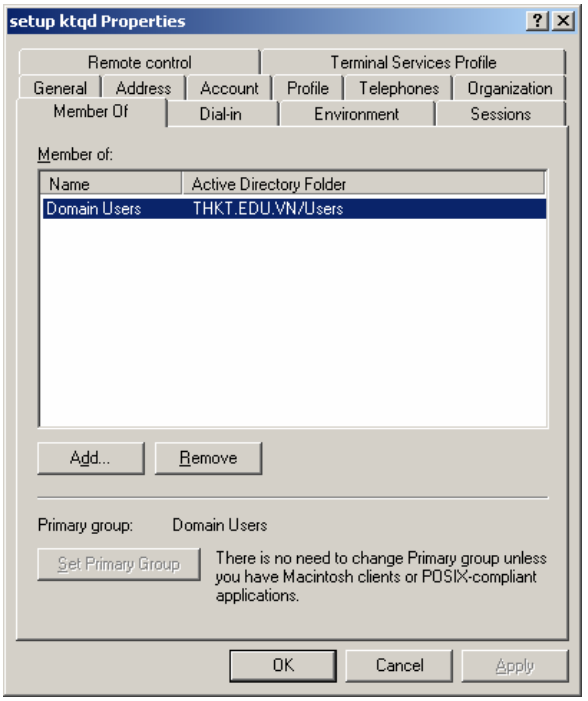


Bấm next để tiếp tục, trên màn hình hiển thị cửa sổ để điền tiếp các tham số



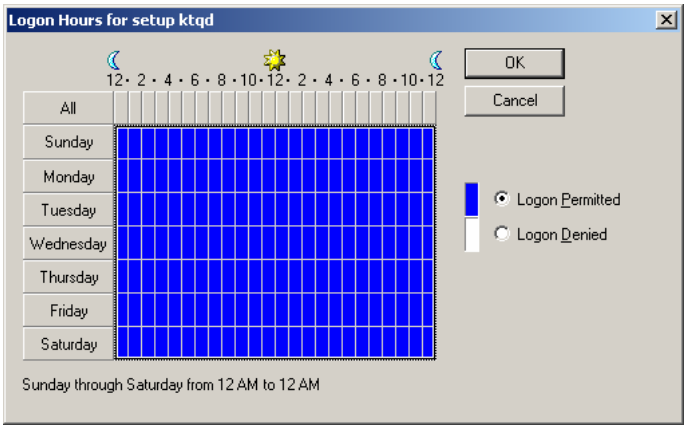
- *Các đặc tính của user, xem thông tin và hiệu chỉnh*

- Chọn user – Properties

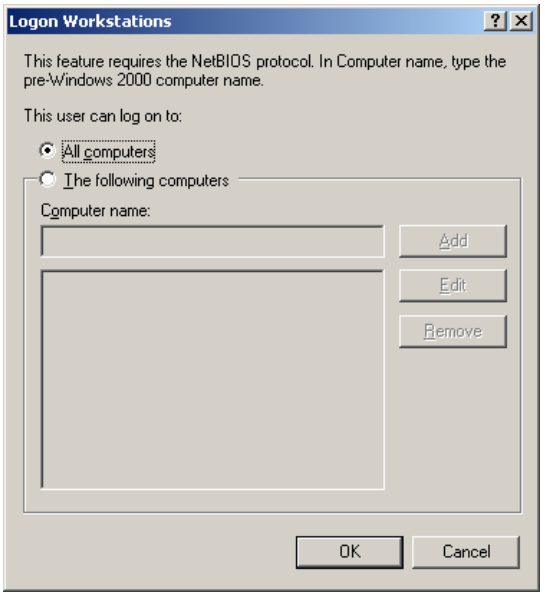


- *Các thiết định về tài khoản*

- Chọn user – Properties, chọn mục Account
- Chọn hạn chế về thời gian cho việc truy nhập



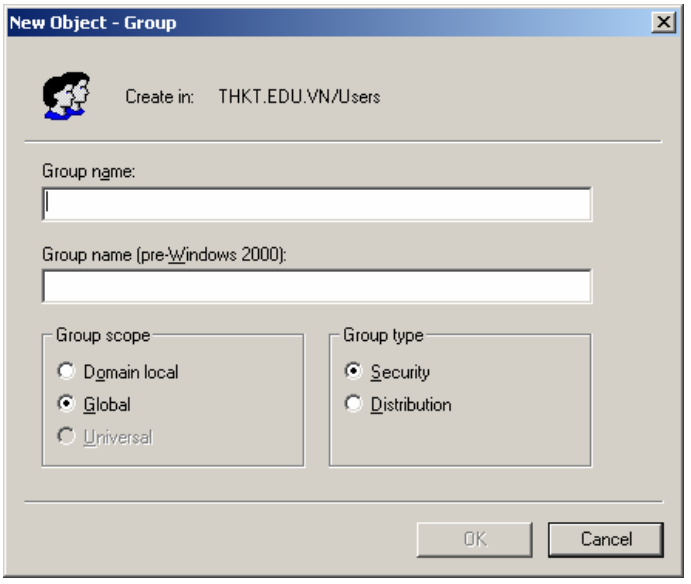
- Chỉ định các máy trạm được phép truy nhập



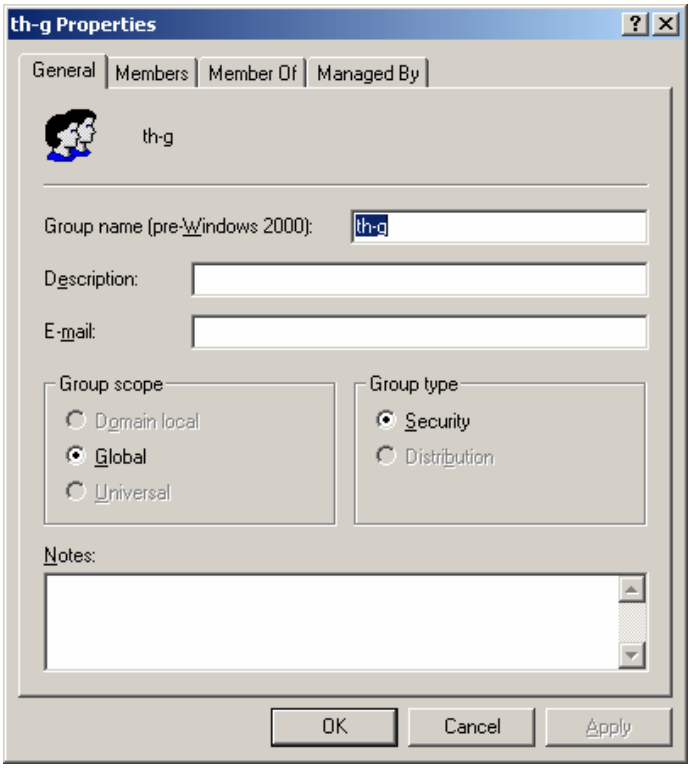
b. Group

 Tạo group

- Chạy trình Activate Directory Users and Computer, bấm phải chuột vào OU users, chọn new – group. Trên màn hình sẽ hiển thị cửa sổ NEW OBJECT – GROUP



- Đưa user vào nhóm phù hợp



3.5. Login Script

Login script là một chương trình máy tính được sử dụng để định hình cho môi trường làm việc, Login script được thực hiện khi user truy nhập mạng. Login script có thể viết bằng nhiều nhôn ngữ như các lệnh shell của DOS/NT/Windows 2000, Windows Scripting Host (WSH), KiXtart, XLNT, Perl, VBScript, Jsscript, …. Login script được viết phụ thuộc vào yếu tố: người xây dựng kịch bản (quản trị mạng), máy khách. Để sử dụng các ngôn ngữ khác thì máy khách cần phải hiểu được ngôn ngữ này, tức là phải cài đặt các trình thông dịch. Để đơn giản có thể chọn các lệnh shell của Windows 2000 / NT/ Windows 9x. Có thể tạo login script bằng các trình soạn thảo như NC, Notepad, … , dùng trình soạn thảo này tạo tệp có cấu trúc TEXT có phần mở rộng là BAT (DOS, Windows 9X) , hoặc có phần mở rộng là CMD (Windows 2000 / NT) và ghi tệp tin này lên thư mục

C:\WINNT\SYSVOL\sysvol\domainname\scripts

Ví dụ: domain có tên thkt.edu.vn thì login script phải được ghi lên thư mục C:\WINNT\SYSVOL\sysvol\THKT.EDU.VN\scripts

- *Một số biến sử dụng:*

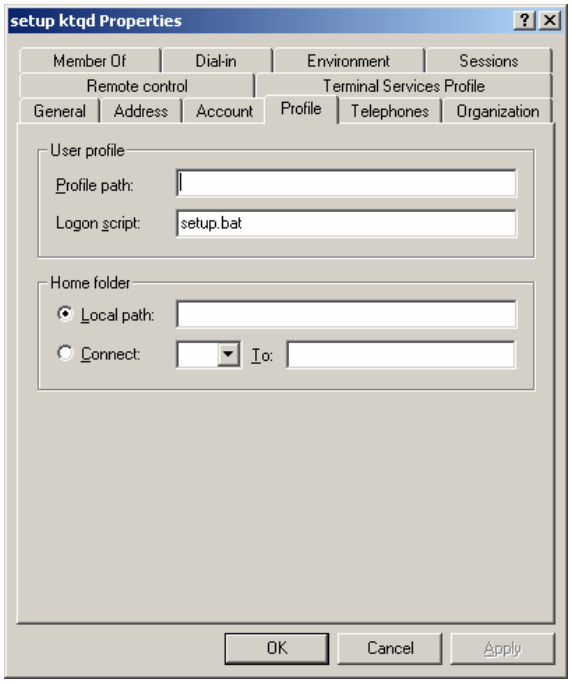
Biến hệ thống	Kết quả
%HOMEDRIVE%	Cho tên ổ đĩa cục bộ ở máy trạm nối với user's home directory
%HOMEPATH%	Cho tên đường dẫn thư mục ngầm định của user (user's home directory)
%OS%	Cho tên hệ điều hành ở workstation
%PROCESSOR_LEVEL %	Cho kiểu processor ở workstation
%USERDOMAIN%	Cho tên domain chứa tài khoản của user
%USERNAME%	Cho tên user

- *Sử dụng trình NET.EXE : Dùng NET USE theo chế độ dòng lệnh*

NET.EXE có trong các máy đã cài Dos client, Windows 9X, Windows 2000/NT

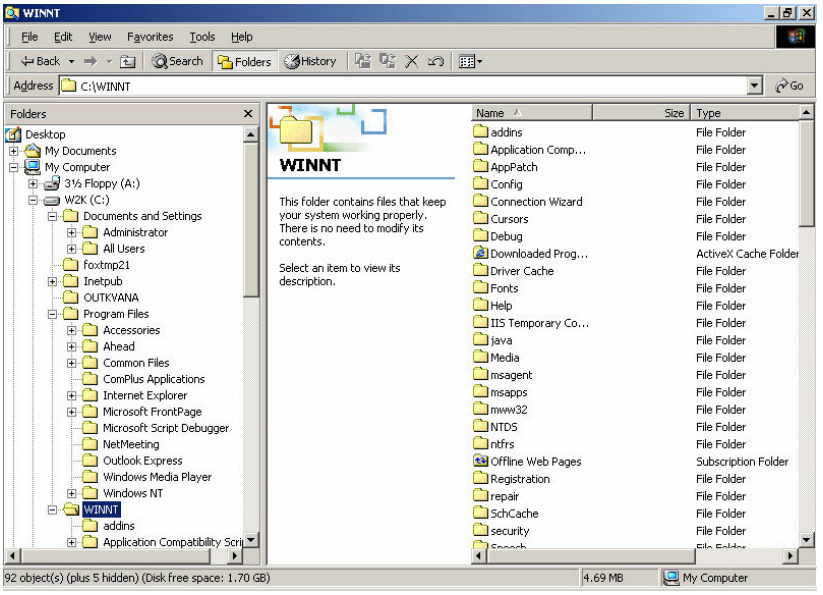
- *Khai báo login script cho user*

Chạy trình Activate Directory Users and Computer, chọn user cần khai báo login script, chọn Properties của user này, sau đó điền tham số.



§ 4. QUẢN TRỊ HỆ THỐNG THƯ MỤC VÀ FILES

4.1 Cấu trúc thư mục của Windows 2000



4.2 Thiết kế hệ thống thư mục cho người dùng

Nguyên lý chung

Tùy qui mô và nhu cầu cung cấp dịch vụ của mạng mà từng hệ thống hệ thống thư mục có thể để riêng trên ổ đĩa logic hoặc để riêng trên một máy.

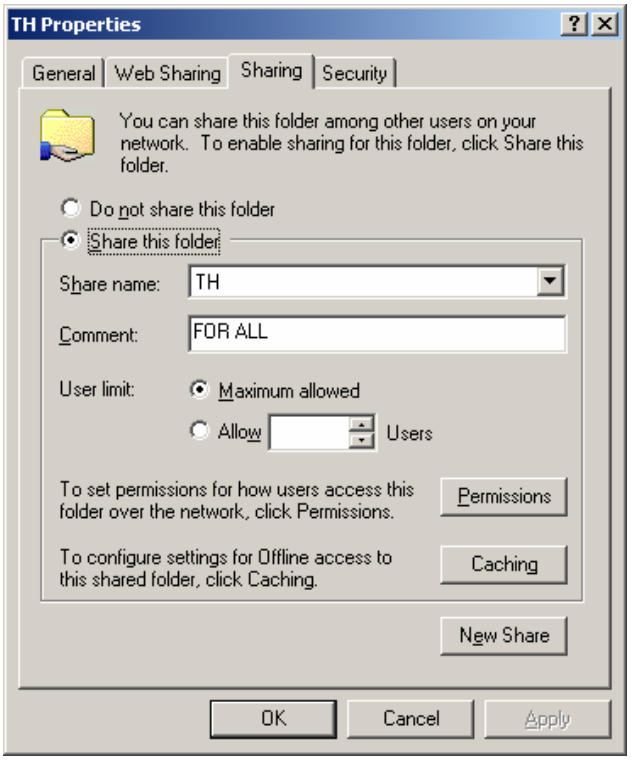
- Dành riêng ổ đĩa logic (c:) cho hệ thống thư mục của Windows 2000.
- Hệ thống thư mục của các chương trình ứng dụng nên phân loại và để theo nhóm, có thể để trên ổ đĩa của windows 2000 nếu dung lượng cho phép, nếu dung lượng lớn cần để riêng trên ổ đĩa logic.
- Hệ thống thư mục của các USER nên tập trung để dễ dàng theo dõi và kiểm soát.
- Hệ thống thư mục của các group nên tập trung để dễ dàng theo dõi và kiểm soát.
- Hệ thống các trình cài đặt của mạng nên để riêng trên ổ đĩa logic
- Tên của các thư mục nên đặt ngắn gọn và có qui luật

4.3 Quản lý quyền truy cập

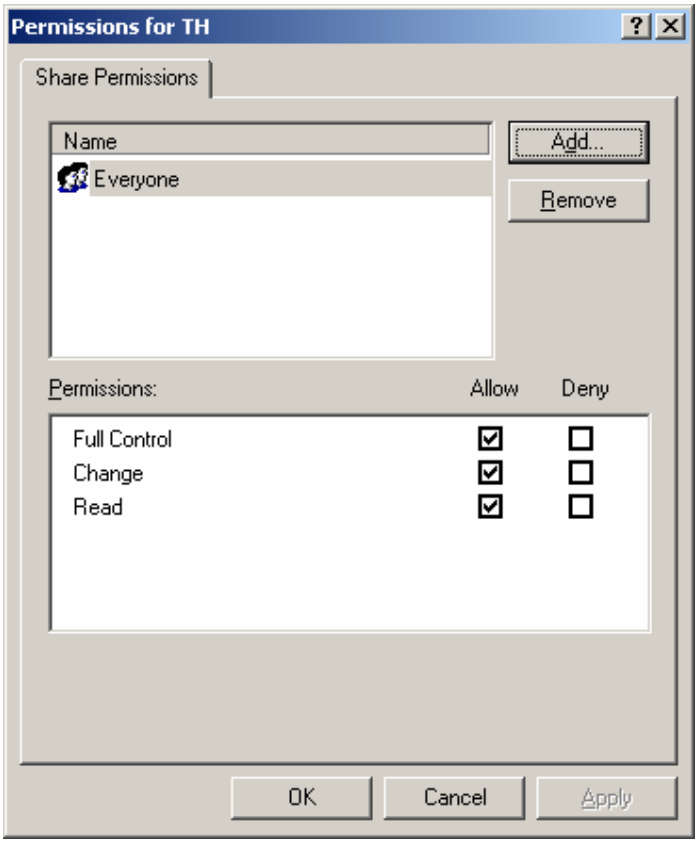
a. Quyền truy cập ở cấp share

Share là hình thức cung cấp dịch vụ ghi chép tệp tin lên ổ đĩa cứng của mạng. Để cung cấp dịch vụ này, quản trị mạng cần tạo share và gán quyền truy cập phù hợp cho user.

- Tạo share: đặt tên share



- Chọn permission để gán quyền truy cập cho user



Full Control: Quyền thực hiện tất cả các công việc trên thư mục và tệp tin

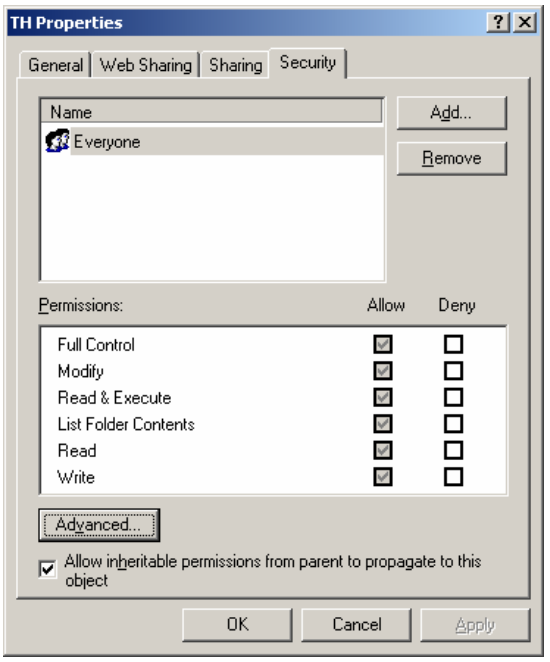
Change: Quyền đọc, thi hành, thay đổi, xóa thư mục và tệp tin
Read: Quyền đọc và thi hành thư mục và tệp tin không có khả năng sửa đổi

b. Quyền truy cập ở cấp thư mục và tệp tin

Quy định permission ở mức thư mục và tệp tin, các quyền được chia nhỏ mở mức chi tiết.

- Chọn Properties của thư mục, chọn security

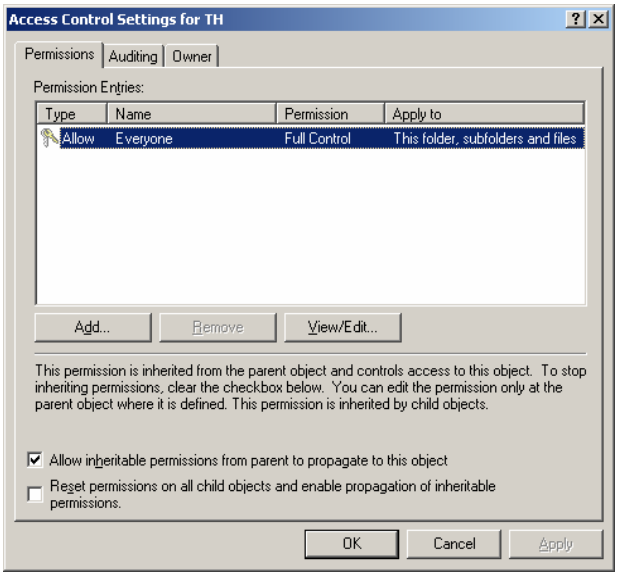
 Các permission mức phân tử (molecular level):



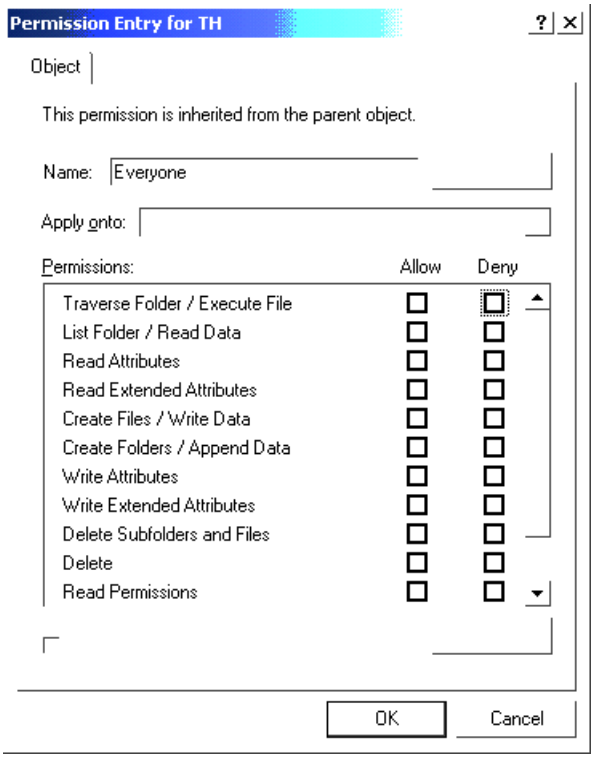
Allow inheritable permissions parent to propagale to this object: Cho phép các thư mục con thừa hưởng các quyền của thư mục mức cao hơn nó.

 Các permission mức nguyên tử (automic level)

- Chọn Properties của thư mục, chọn Advanced



- Chọn View/Edit



Write Traverse Folder/Execute File

List Folder/Read Data

Read Attributes

Read Extended Attributes

Create Files/Write Data

Create Folders/Append Data

Write Attributes

Write Extended Attributes

Delete Subfolders and Files

Delete

Read

Change

Take Ownership

Synchronize

- *Quyền sở hữu (ownership)*

  Chủ nhân của một đối tượng (tệp tin, thư mục, ...) có một thuộc tính gọi là ownership, owner tách biệt với permission. Nếu có owner đối với một đối tượng thì có thể phân bố lại permesion cho đối tượng này. Người tạo ra đối tượng sẽ là owner ngầm định của đối tượng.

c. Các share hệ thống

Các share được tạo sẵn được dùng chung cho hệ thống, cuối tên share chứa ký tự $, các share này đều là các share ẩn.

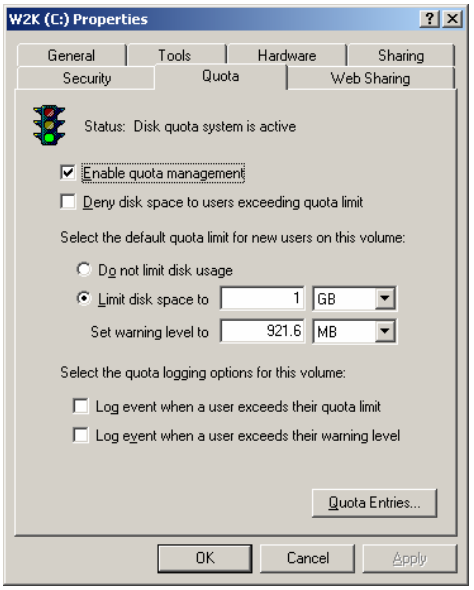
Tất cả các ổ đĩa đều có một share ẩn ứng với thư mục gốc của mỗi ổ đĩa đó. Các share này gọi là share phục vụ việc quản trị (administrative share). Không thể thay đổi các permission hoặc đặc tính của các share này, nhưng có thể chấm dứt hoàn toàn việc chia sẻ chúng, ví dụ các share C$, D$, ...

Các share khác được dùng vào mục đích cụ thể:

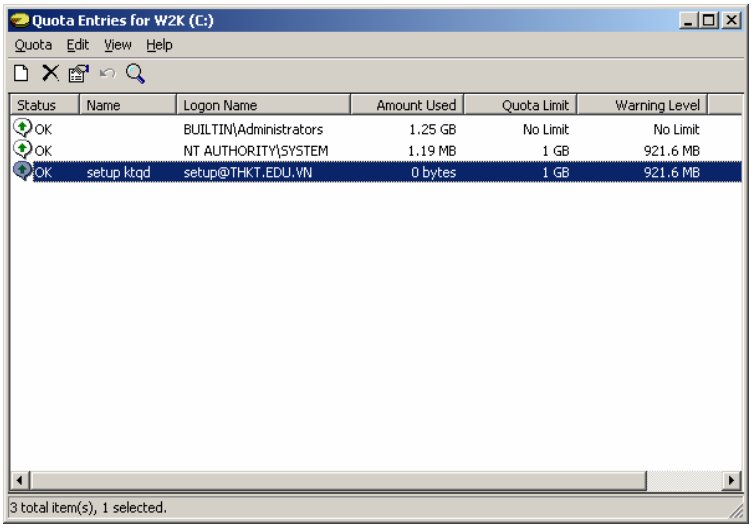
- Admin$: ứng với thư mục gốc của hệ thống (thư mục chứa tệp tin của windows 2000 ví dụ c:\winnt)
- Print$: Khi tạo một máy in chung, hệ thống sẽ đặt các driver cho máy in đó vào trong share này.
- IPC$ Dùng để truyền thông giữa các máy tính
- Repl$: Sao chép dữ liệu giữa các server khi các máy dùng dịch vụ sao chép (replication service)
- Các share có thể ẩn nếu khi đặt ký tự $ vào cuối tên share

4.4 Cấp phát hạn ngạch đĩa (quota)

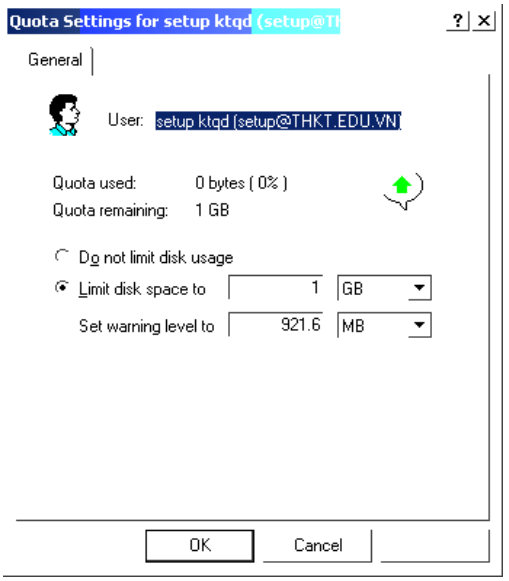
Để kiểm soát dung lượng đĩa cứng trên server, windows 2000 cung cấp một công cụ nhằm chỉ định dung lượng đĩa mà người sử dụng có thể ghi lên server. Để thực hiện chức năng này chọn properties của ổ đĩa trong cửa sổ explore.



Chọn ô quota, trên màn hình hiển thị cửa sổ properties, chọn enable quaota management, chọn quota entries



Trên màn hình hiển thi danh sách các user được cấp quota, click chuột vào tên user để hiệu chỉnh tham số. Thêm user vào danh sách cấp hạn ngạch, chọn trên thực đơn quota – new quota entry. Chon user của domain trên danh sách hiên thị trên màn hình, tiếp theo xác định dung lượng đĩa cần hạn chế.



§ 5. CÀI ĐẶT CLIENT, TRUY NHẬP VÀO MẠNG

Sau khi Windows 2000 được cài đặt trên SERVER cần phải được SHARE đĩa cứng, gán quyền truy cập và quyền sử dụng cho các USER và GROUP. Windows 2000 cho phép các Workstation (DOS, WINDOWS For Workgroup, WINDOWS 9X, Windows NT Workstation, Macintosh) truy nhập các tài nguyên hệ thống. Tuy nhiên tùy theo từng hệ điều hành mà trên SERVER và Workstation cần phải cấu hình phù hợp.

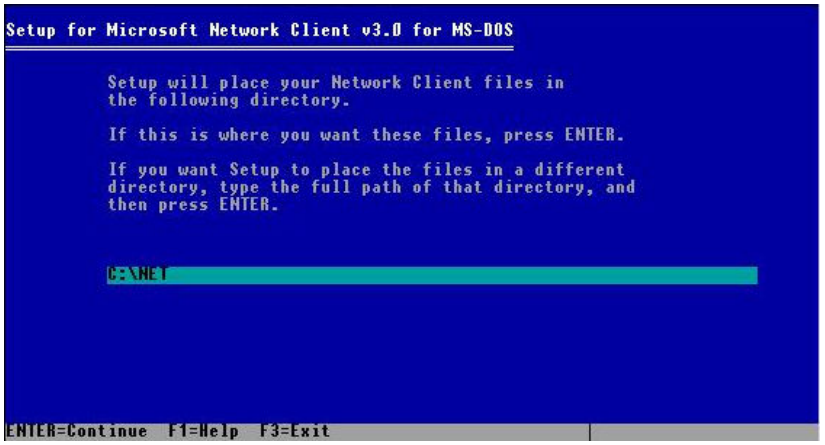
5.1. Hệ điều hành DOS

Các máy sử dụng điều hành DOS có thể truy cập vào Windows 2000 bằng các phần mềm Microsoft Network Client hoặc Microsoft LAN Manager.

 Cài đặt Microsoft Network Client

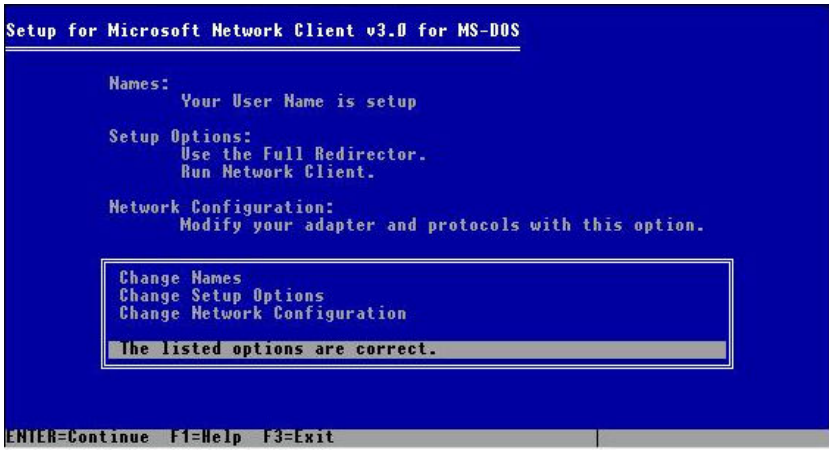
Sử dụng đĩa cài đặt (trên đĩa CD Windows NT Server, có thể copy ra đĩa mềm)

- Khởi động DOS và chạy trình SETUP

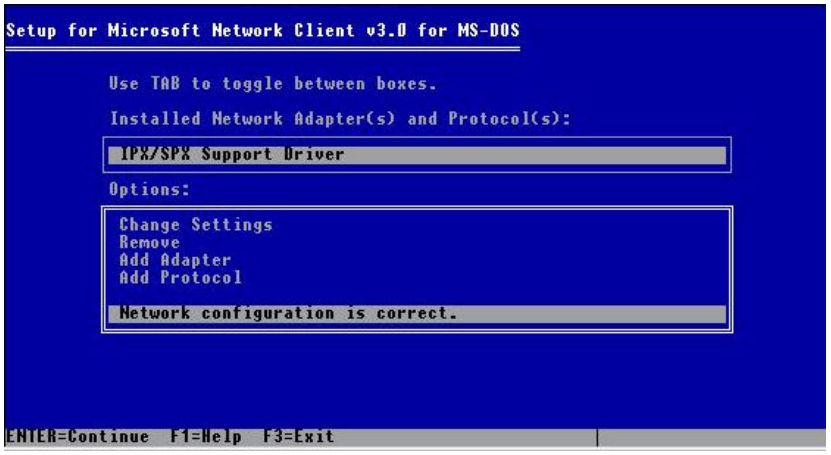


+ Chọn thư mục cài đặt (ngầm định c:\net)

- Bấm enter để tiếp tục



+ Chọn change name: để hiệu chỉnh các tham số về
    + User name
    + Computer name
    + Workgroup name
    + Domain name
+ Chọn change setup options: để hiệu chỉnh các tham số về
    + Redir options: nếu chọn Full Redirector thì có khả năng
        + Truy cập vùng và Login scrript
        + Remote Access Service
        + Gửi thông báo
        + Các cơ chế truyền thông liên kết xử lý như Remote procedure calls, Windows Sockets (Winsock- giao diện giữa chương trình và giao thức vận chuyển).
    + Logon validation: chọn logon to domain
+ Chọn network configuration: để hiệu chỉnh các tham số về



+ Add adapter: Chọn card mạng, chọn trên danh sách, nếu MS client không hỗ trợ thì cần khai báo thư mục chứa driver card mạng.

+ Add protocol: Các giao thức hỗ trợ
  + NetBEUI
  + IPX Compatible Transport
  + TCP/IP: có hỗ trợ DHCP, không hỗ trợ phân giải tên DNS

Sau khi chọn, hiệu chỉnh các tham số, hệ thống sẽ được cài đặt và yêu cầu khởi động lại máy tính.

 Truy nhập vào mạng:

Nếu quá trình cài đặt thành công, các tệp tin của hệ thống đã được tạo ra, nội dung của các tệp tin như sau:

+ Tệp tin CONFIG.SYS

```
FILES=45
device=a:\himem.sys
device=a:\MSCLIENT\ifshlp.sys
dos=high
LASTDRIVE=Z
```

+ Tệp tin AUTOEXEC.BAT

```
SET PATH=a:\MSCLIENT
a:\MSCLIENT\net start
```

Sau khi các tệp tin đã được nạp thành công (Đã login vào mạng), có thể sử dụng dịch vụ đơn giản của mạng .

Có thể dùng tệp lệnh NET.EXE để thực hiện một số công việc về mạng như ánh xạ ổ đĩa (map), khai báo máy in mạng.

 Một số hướng dẫn về NET.EXE

- Chức năng: hỗ trợ user sử dụng tài nguyên của mạng như đĩa cứng, máy in, hiển thị thông tin kết nối.

NET: thực hiện theo giao diện thực đơn

NET HELP: trợ giúp
NET USE
NET USE [drive: | *] [\\computer\directory [password | ?]]
[/PERSISTENT:YES | NO] [/SAVEPW:NO] [/YES] [/NO]
NET USE [port:] [\\computer\printer [password | ?]]
[/PERSISTENT:YES | NO] [/SAVEPW:NO] [/YES] [/NO]

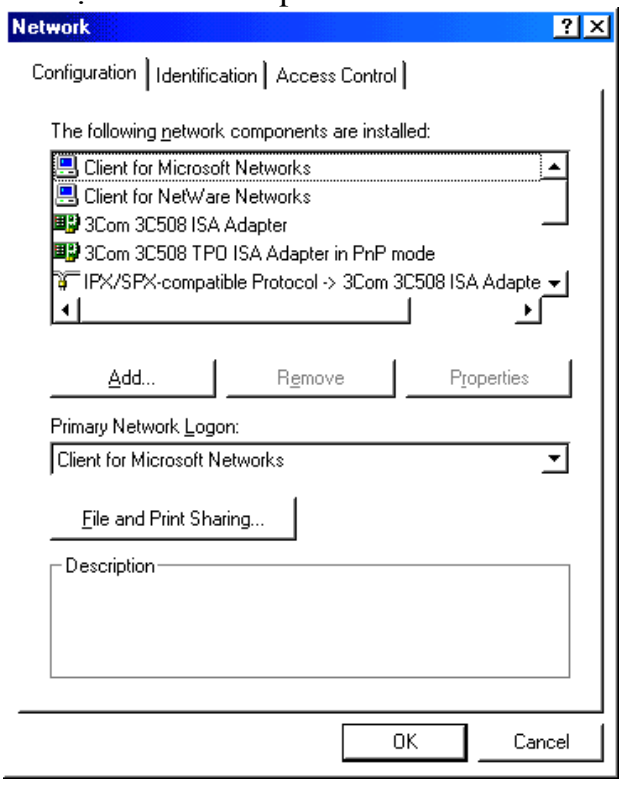
NET USE drive: | \\computer\directory /DELETE [/YES]
NET USE port: | \\computer\printer /DELETE [/YES]
NET USE * /DELETE [/YES]

drive Ký tự chữ cái tên ổ đĩa
* Tự gán chữ cái tên ổ đĩa tiếp theo. Nếu dùng /DELETE, ngắt các kết nối
port Cổng LPT cho máy in
computer Tên SERVER
directory Tên thư mục được shared
printer Tên máy in

5.2. Hệ điều hành WINDOWS 9X

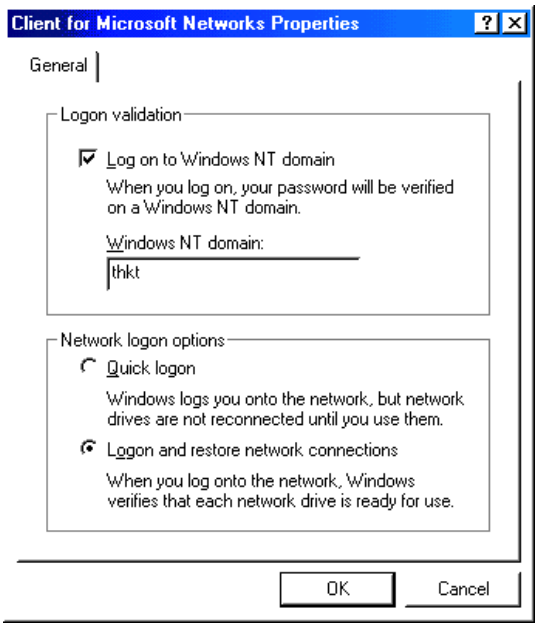
- *Cài đặt*

- Máy phải được cài đặt phần cứng như dây cáp mạng, card mang
- Máy phải được cài đặt và cấu hình phần mềm



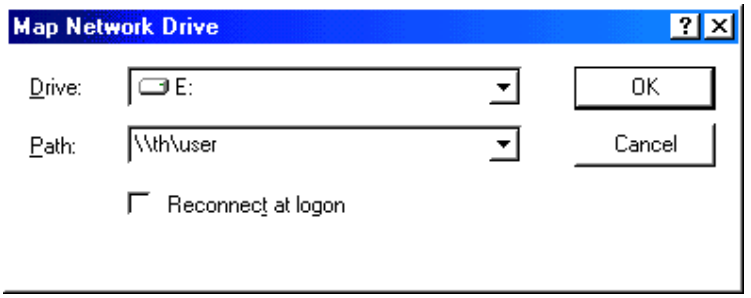
+ Card mạng
+ Dịch vụ: Client for Microsoft Network
+ Protocol: TCP/IP

- Khai bao tên domain trong Properties của dịch vụ Client for Microsoft Network



 Sử dụng

- Khi khởi động máy: nhập tên và mật khẩu hợp lệ
- Có thể sử dụng dịch vụ ghi chép thư mục, tệp tin lên ổ đĩa mạng bằng cách ánh xạ ổ đĩa mạng thành ổ đĩa logic của máy.
  Chọn Map Network Driver (trong EXPLORE)



+ Driver: đặt tên ổ đĩa logic
+ Path: \\server_name\Share_name

- Sử dụng dịch vụ in ấn
- Có thể sử dụng các dịch vụ khác như: mail, internet, ... nếu trên máy server có cung cấp những dịch vụ này.

5.3. Truy nhập vào Workstation khác trong mạng (mạng ngang hàng)

Tùy từng hệ điều hành được cài đặt ở máy trạm mà cách cài đặt cụ thể có khác nhau, đối với mạng ngang hàng chi có thể cung cấp một số dịch vụ mạng đơn gian như chia xẻ tệp tin, thư mục, chia xẻ máy in. Nếu các máy trạm đề cài đặt windows 9x, windows NT, hoặc windows 2000 thì về nguyên tắc có thể cấu hình như sau:

- Ở máy cần truy nhập (đích, chứa tài nguyên) đã được cài đặt phần cứng, cầu hình đầy đủ phần mềm cho mạng. Máy này cần chia xẻ tài nguyên để dùng chung trên mạng bằng cách chọn Share và điền các tham số:

+ Điền Share Name: Đặt tên cho ổ đĩa, hoặc thư mục để máy khác có thể truy nhập

+ Gán quyền truy cập hoặc những hạn chế nếu có, đối với windows 9x thì chỉ có thể đặt password, đối với windows NT thì có thê gán quyền truy nhập chi tiết hơn.

- Ở máy truy nhập (sử dụng tài nguyên) đã được cài đặt phần cứng, cầu hình đầy đủ phần mềm cho mạng. Login vào mạng với đủ thẩm quyền.

+ Có thể sử dụng dịch vụ ghi chép thư mục, tệp tin lên ổ đĩa mạng bằng cách ánh xạ ổ đĩa mạng thành ổ đĩa logic của máy.

Chọn Map Network Driver (trong EXPLORE), điền các tham số

+ Driver: đặt tên ổ đĩa logic

+ Path: \\server_name\Share_name

§ 6. QUẢN TRỊ PHƯƠNG TIỆN LƯU TRỮ

6.1. Một số khái niệm

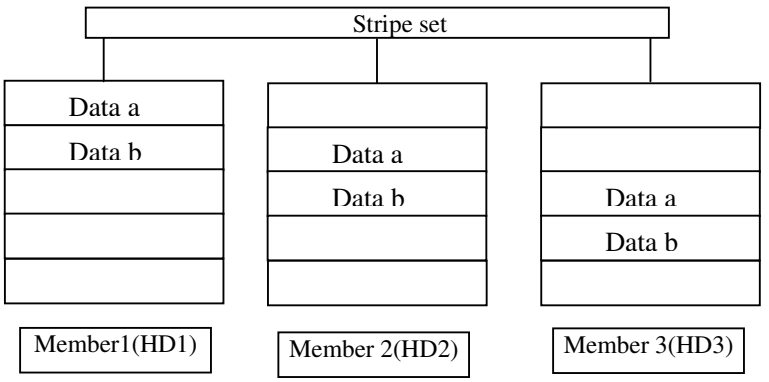
a. Raid

Phương pháp bảo vệ dữ liệu của hệ thống bằng cách kết hợp các đĩa cứng để tăng tính an toàn dữ liệu. Có sáu kiểu thực hiện raid (raid level), windows 2000 hỗ trợ các mức raid 0, 1, 5 .

 Raid 0: chia dải không chẵn lẻ (striping withou parity)

Kỹ thuật chia dải không chẵn lẻ là kỹ thuật liên kết các chỗ trống trên các đĩa vật lý thành một bộ đĩa chia dải (stripe set). Mỗi thành viên trong stripe set được chia thành các dải (stripe) có kích thước bằng nhau, khi hệ thống ghi dữ liệu vào stripe set, dữ liệu sẽ được phân bố trên các dải này. Để thực hiện kỹ

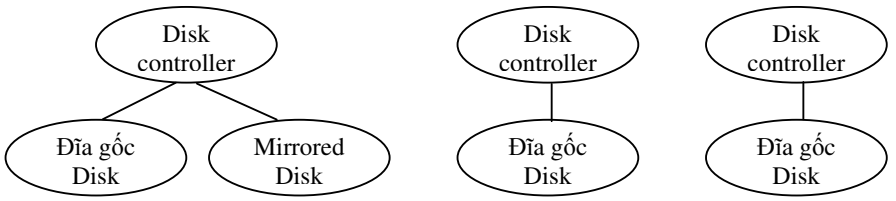
thuật này cần ít nhất hai đĩa, tối đa 32 đĩa, có thể mô tả kỹ thuật stripe set như sau:



Khi hệ thống đọc hoặc ghi dữ liệu thì tất cả các thành viên của bộ đĩa đều đọc và ghi do vậy thời gian đọc ghi sẽ được cải thiện.

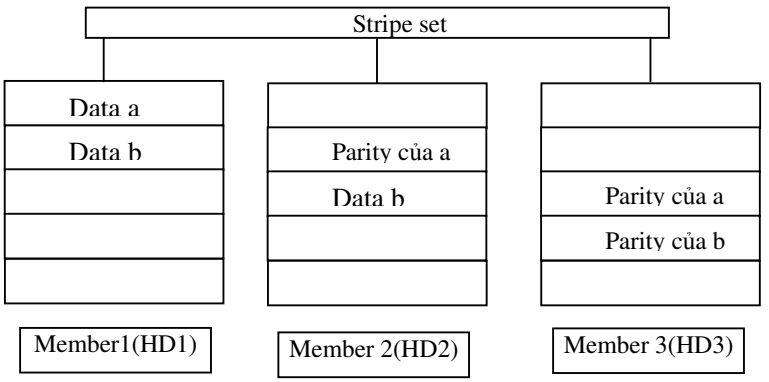
- *Raid 1: phản chiếu đĩa (disk mirroring)*

Kỹ thuật phản chiếu đĩa là kỹ thuật ghi dữ liệu lên hai đĩa vật lý riêng biệt nhưng có chung một bộ điều khiển đĩa (disk controller). Trong trường hợp một đĩa bị lỗi thì vẫn còn dữ liệu ở đĩa còn lại. Disk mirroring khác với trường hợp nhân đôi đĩa (disk duplexing), kỹ thuật disk duplexing sử dụng hai đĩa cứng với hai bộ điều khiển đĩa.



- *Raid 5: chia dải có chẵn lẻ (striping with parity)*

Kỹ thuật chia dải chẵn lẻ là kỹ thuật liên kết các chỗ trống trên các đĩa vật lý thành một bộ đĩa chia dải (stripe set). Theo kỹ thuật này khi hệ thống ghi dữ liệu lên đĩa, dữ liệu đượoc ghi rải ra khắp các đĩa trong dải. Thông tin chẵn lẻ ứng với dữ liệu cũng được ghi vào đĩa, bằng cách này nếu có sự cố sảy ra với một trong các đĩa, phần dữ liệu trên đĩa này có thể được tái tạo từ những thông tin chẵn lẻ trên các đĩa còn lại.



b. Basic disk và dynamic disk

Đĩa cơ bản (Basic disk) là loại kỹ thuật lưu trữ thông thường theo chuẩn dos , các đĩa cơ bản không hỗ trợ các phân vùng chịu lỗi. Đĩa động (dynamic disk) là kỹ thuật lưu trữ đĩa cho phép tạo ra, mở rộng và xóa bỏ các phân vùng trên nhiều đĩa vật lý. Đĩa động không có tính tương thích, mọi hệ điều hành không phải Window 2000 đều không đọc được chúng.

c. Phân khu đĩa (partition)

Một phân khu đĩa (partition) theo kiểu đĩa cơ bản, là một phần của đĩa vật lý được chia làm hai loại phân khu chính (primary partition) và phân khu mở rộng (extended partition).

- Phân khu chính: là phân khu mà hệ điều hành có thể khởi động được, không thể chia phân khu chính ra làm phân khu con. một đĩa cứng có thể chia tối đa 4 phân khu chính.
- Phân khu mở rộng: để lưu trữ được dữ liệu, thì trên phân khu này cần tạo thành một hoặc nhiều ổ đĩa logic, các hệ điều hành không thể khởi động được từ phân khu này.

d. (Volume set)

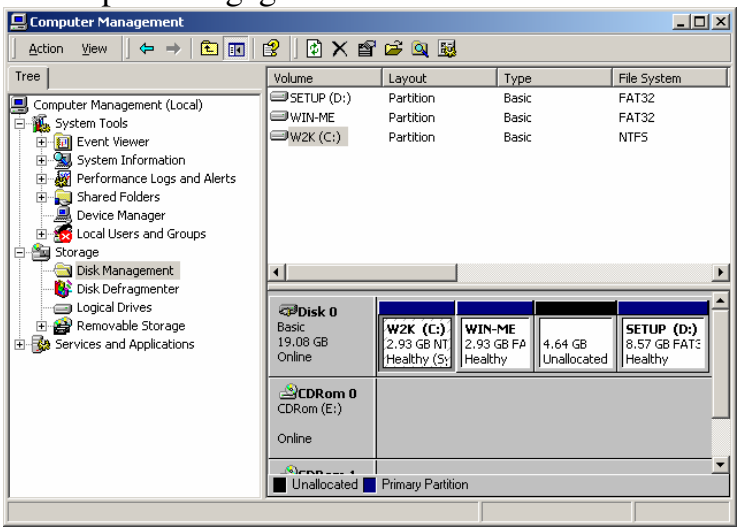
Volume set là hình thức liên kết các phân vùng của một hoặc nhiều đĩa cứng vật lý thành ổ đĩa logic, Windows 2000 hỗ trợ hai loại volume set. Volume set đơn giản (simple volume set) được tạo từ một đĩa cứng, volume set trải dàn (spanned volume set) được tạo từ nhiều đĩa cứng. Các volume set không trợ giúp tính năng chịu lỗi, chúng chỉ cho phép sử dụng đĩa cứng thành đĩa logic với dung lượng lớn. Một trong các đĩa cứng dùng volume set bị lỗi thì volume set cũng lỗi.

6.2. Quản lý đĩa cứng

Để quản lý đĩa windows 2000 cung cấp công cụ disk management trong thực đơn computer management, công cụ này hỗ trợ việc khởi tạo một đĩa cứng mới

thành đĩa làm việc, kiểm tra việc hoạt động của đĩa, xem và có thể hiệu chỉnh các thông tin về đĩa.

Chọn thực đơn computer magagement



Khởi tạo một đĩa cứng mới

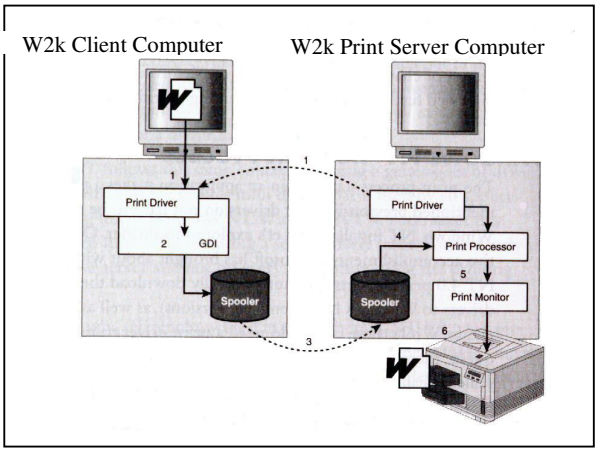
- Tạo đĩa sử dụng theo kiểu đĩa cơ bản (basic disk)
- Tạo đĩa sử dụng theo kiểu đĩa động (dynamic disk)

Giải phân mảnh đĩa

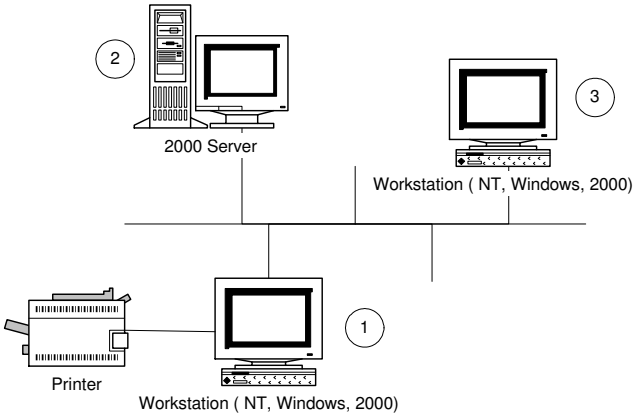
§ 7. QUẢN TRỊ DỊCH VỤ IN ẤN TRÊN MẠNG

7.1. Nguyên lý in ấn trên windows 2000 :

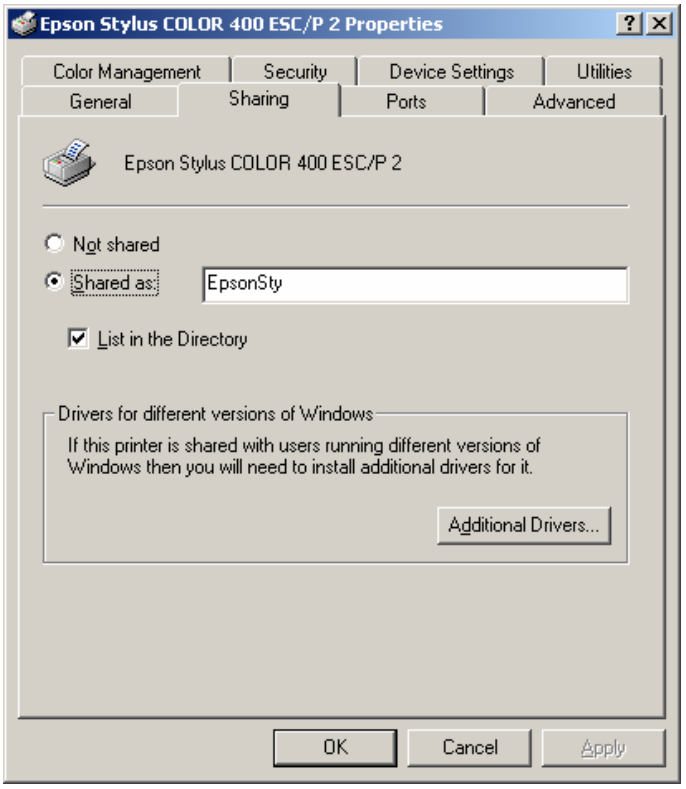
1- Client (workstation) gửi yêu cầu in
2- Client tạo tác vụ in, ngầm định GDI (Graphics Device Interface)
3- Spooler trên máy client gửi đến Spooler của máy Server
4- Print processor: xử lý
5- Máy in



7.2. Một kiểu cấu hình in trên mạng:

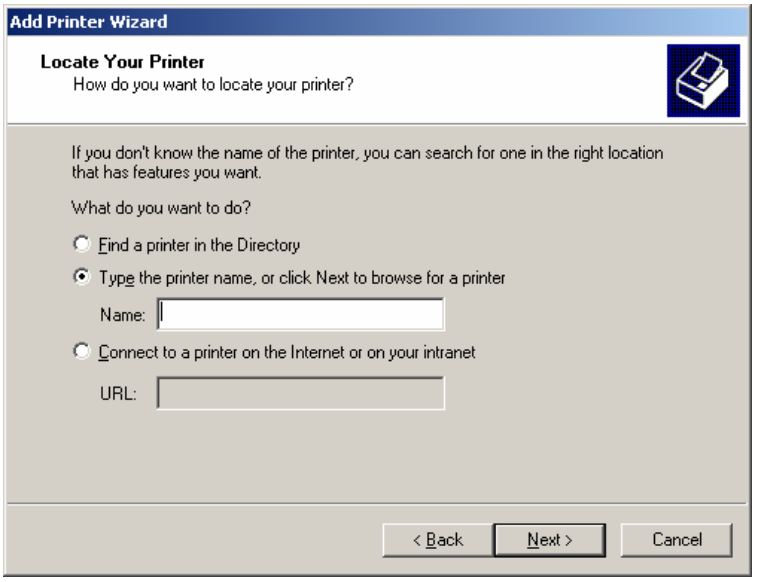


- *Máy tính cắm máy in (W2k): (1)*
  - Phải là thành viên của domain Windows 2000 server
  - Login vào mạng (domain) với đủ thẩm quyền
  - Cài đặt máy in: nối máy in vào máy tính, cài đặt driver
  - Share máy in

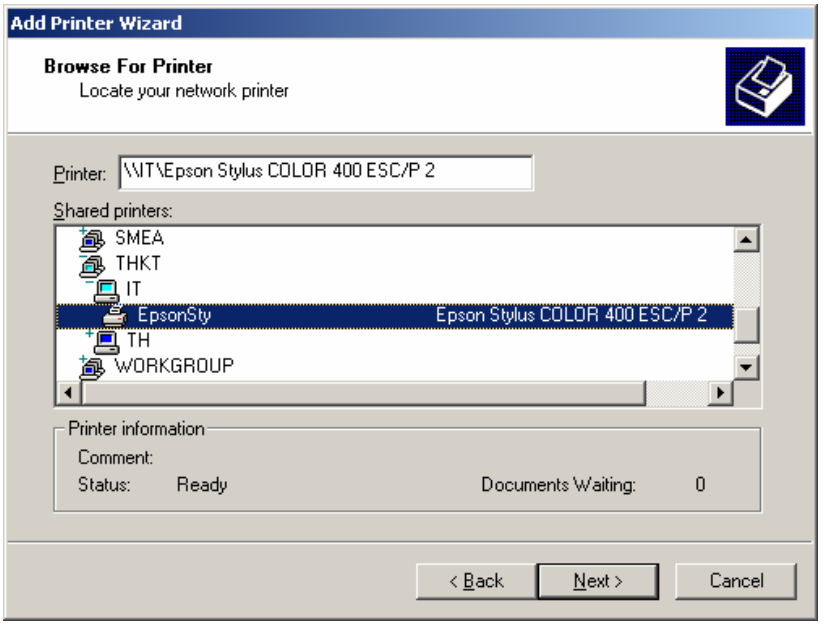


- *Printer Server (domain) (2)*

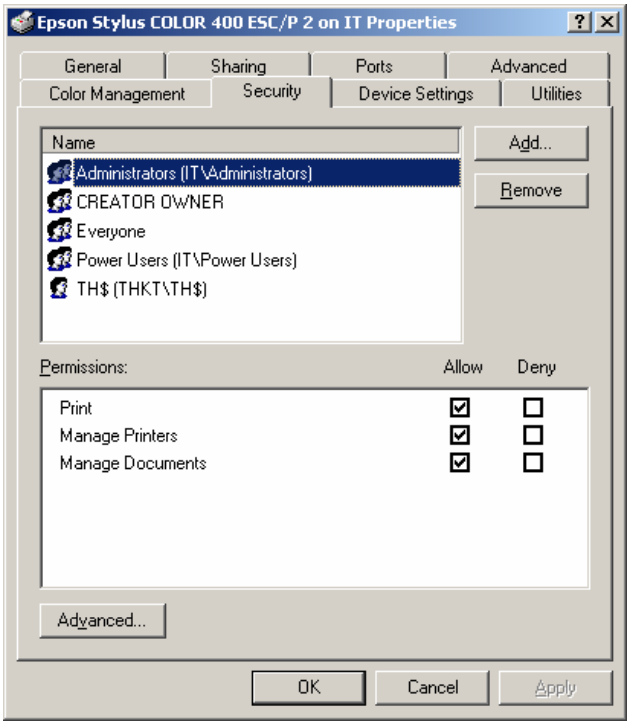
  - Add máy in (chọn máy in trên mạng):



- Có thể dùng chức năng brow để tìm kiếm máy in trên mạng



Sau đó hiệu chỉnh các tham số về máy in (nếu cần), gán quyền in điều khiển và sử dụng máy in:



Việc sử dụng máy in có thể chọn các Permission sau:

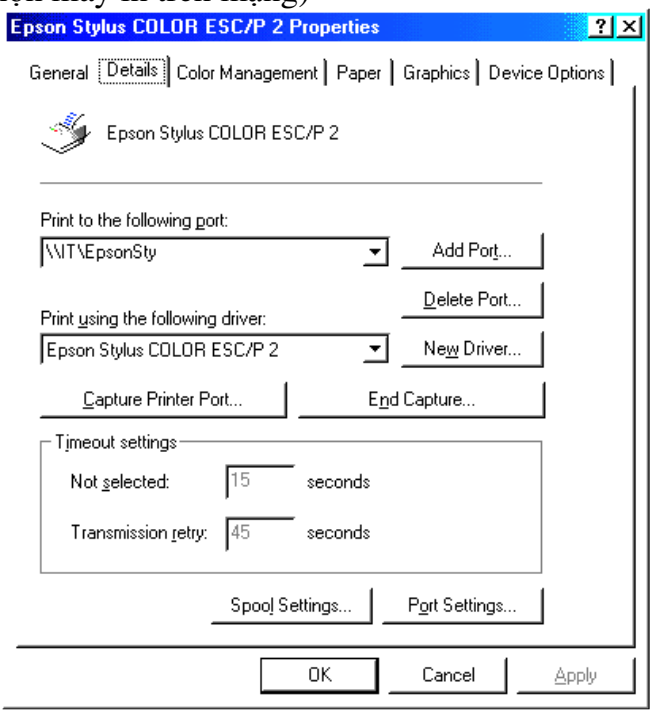
+ Print: chỉ được quyền in

+ Manage Printer: Quyền quản lý máy in

+ Manage Document: Quyền quản lý tài liệu in ấn như tạm dừng in, xóa các tác vụ in.

 Máy cần in Windows 9x (3)

- Login vào mạng (domain) với đủ thẩm quyền in
Add máy in (chọn máy in trên mạng)



Trong cửa sổ PRINT TO THE FOLLWING PORT:
\\ Server name \Printer Share name (\\IT\EpsonSty)
+ Server name: Tên máy tính cắm máy in đã được share
+ Printer Share name: tên share của máy in

§ 8. QUẢN TRỊ SAO LƯU DỮ LIỆU

8.1. Khái niệm

Trong quá trình hoạt động máy tính có thể gặp các sự cố như mất điện đột ngột, hoặc các thiết bị như đĩa cứng có thể gặp lỗi, và nhiều nguyên nhân khác, khi đó có thể xảy ra việc mất một phần hoặc toàn bộ dữ liệu trên Server. Sao lưu dữ liệu là một công việc quan trọng của quản trị mạng, có thể phân chia các công việc sao lưu dữ liệu như sau:

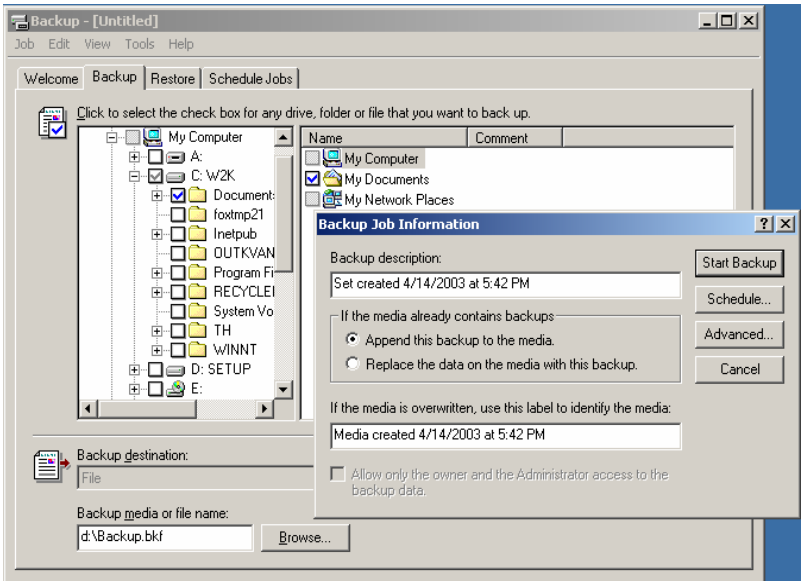
Sao lưu hệ thống: sao lưu Windows 2000 Server, cơ sở dữ liệu của hệ thống như registry và các phần mềm chạy trên nền Windows 2000 Server như SQL, Lotus notes ... Sự thay đổi của những files bao giờ cũng gắn liền với hệ thống, việc sao lưu những files này đòi hỏi can thiệp như: ngừng hoạt động chương trình, tắt các dịch vụ ... .

Sao lưu dữ liệu của người dùng: Những dữ liệu do người dùng ghi lên mạng, những files này không làm ảnh hưởng đến hệ thống.

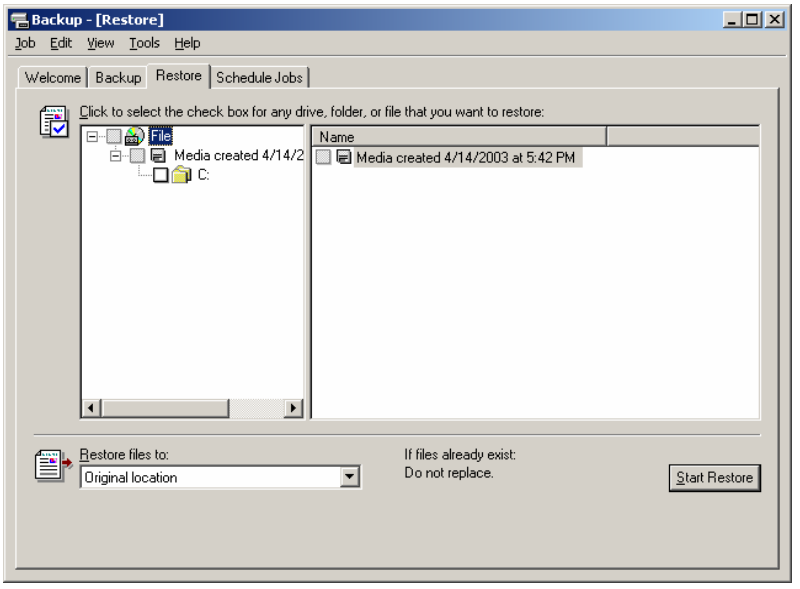
8.2. Sao lưu hệ thống

 Sử dụng trình backup của windows 2000 trong thực đơn hệ thống

- Sau lưu:



- Khôi phục:



 Sử dụng các giải pháp khác

- Hệ thống server song hành
- Giải pháp đồng bộ về sao lưu dữ liệu của HP
- Giải pháp sao lưu dữ liệu của các hãng khác như Seagate Technology
- Các phần mềm sao lưu dữ liệu

8.3. Sao lưu dữ liệu

Có thể dùng băng từ, đặt lịch sao lưu dữ liệu tự động.

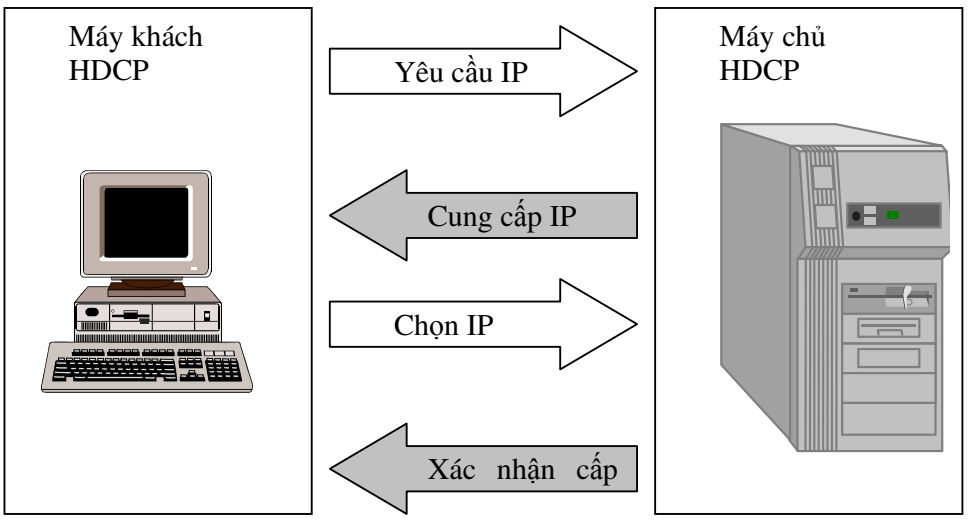
Nếu trên Server có ổ đĩa cứng dự phòng có thể sử dụng các chương trình lập lịch để tự động Copy dữ liệu.

Nếu trên Workstation dung lượng ổ đĩa cứng còn dư có thể sử dụng các chương trình lập lịch để tự động Copy dữ liệu.

§ 9. MỘT SỐ DỊCH VỤ MẠNG CỦA WINDOWS 2000

9.1. Giao thức cấu hình máy động (Dynamic Host Configuration Protocol - DHCP)

a. Khái niệm

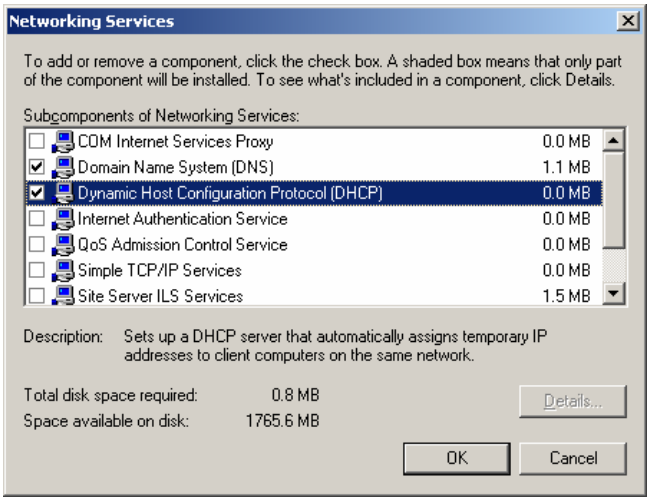


Giao thức cấu hình máy động (DHCP) được thiết kế để tập trung hóa cấu hình và quản lý thông tin cấu hình TCP/IP bằng cách gán tự động các địa chỉ IP cho các máy được cấu hình theo DHCP. Phần mềm DHCP được cài đặt trên cả máy chủ và máy khách. Trên mạng cần ít nhất một máy server được cài đặt DHCP server. Microsoft cung cấp bộ giao thức TCP/IP với các DHCP client có sẵn trên windows 9x, windows 2000.

b. Cấu hình DHCP Server

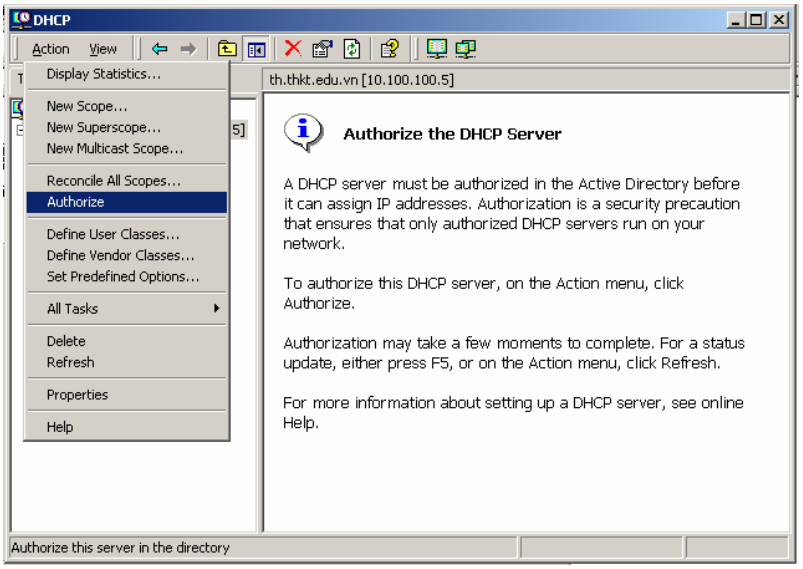
- Cài đặt:

+ Add dịch vụ DHCP trong Control Panel – Add/Remove program
+ Chọn Add/Remove Windows Components
+ Chọn Networking Services, chọn Details
+ Chọn DHCP



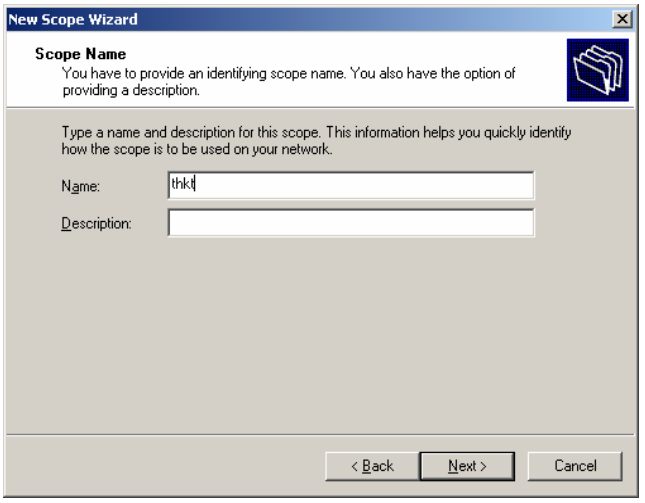
- Cấu hình DHCP được thực hiện thông qua trình DHCP Manager của Administrative Tools

+ Trao quyền cho DHCP

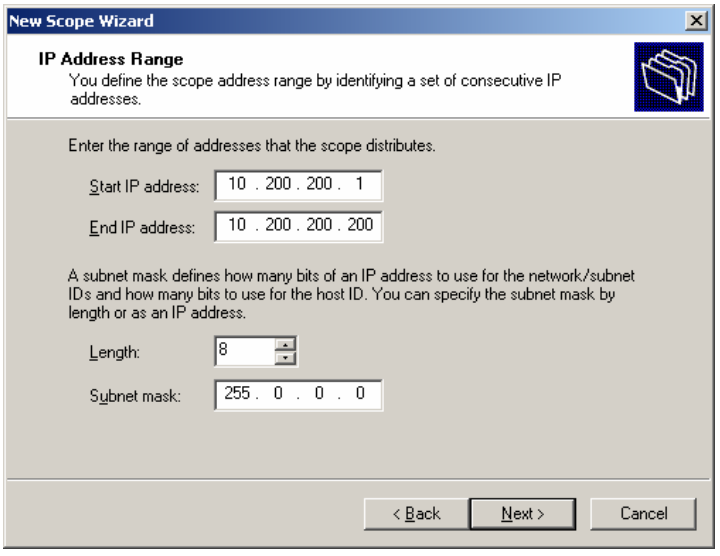


+ Tạo khu vực (Scope DHCP)

+ Đặt tên cho khu vực:



+ Chỉ định phạm vi



+ Start IP Address: Bắt đầu địa chỉ IP có thể cấp cho máy khách DHCP

+ End IP Address: Kết thúc địa chỉ IP có thể cấp cho máy khách DHCP

+ Subnet Mask: Để gán cho mọi máy khách DHCP

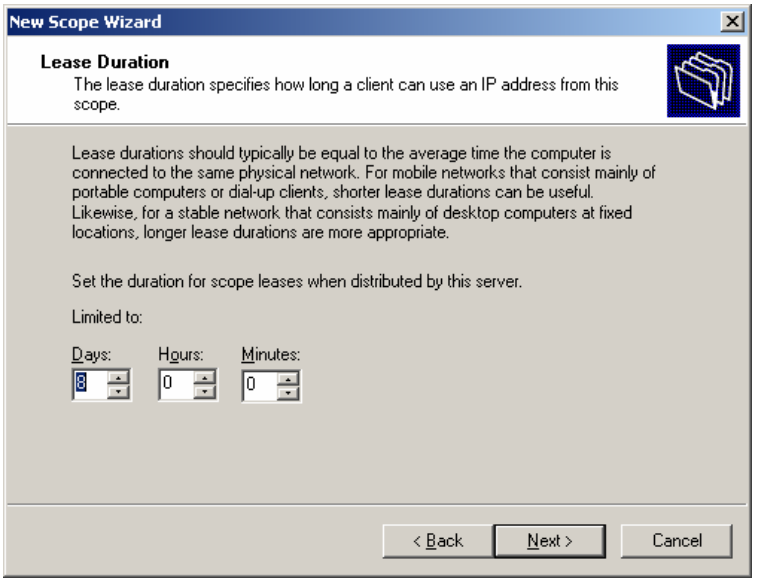
Theo ví dụ trên dải địa chỉ từ 10.200.200.1 đến 10.200.200.200 sẽ được cấp lần lượy cho các máy trạm khi truy cập mạng.

+ Có thể chọn tiếp tham số để loại ra một vài địa chỉ IP đã được dùng cho mạng

Exclusion Range – Start Address: bắt đầu địa chỉ IP cần loại bỏ

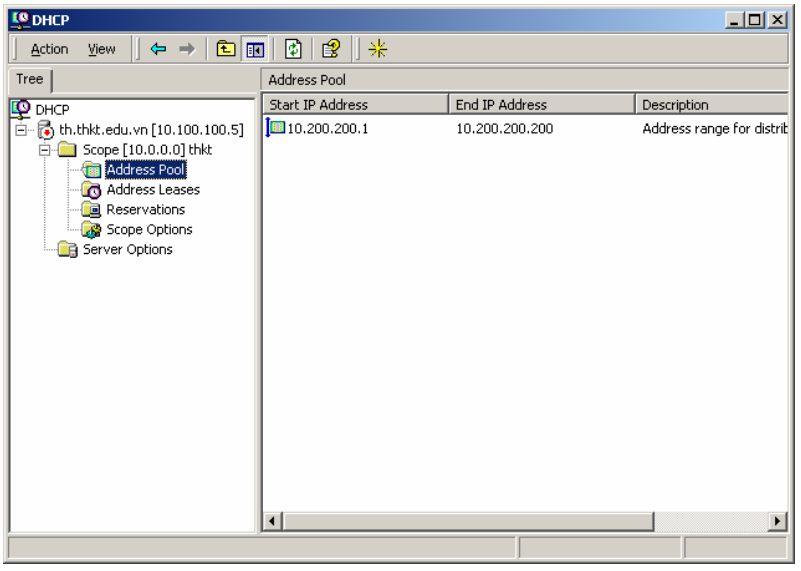
Exclusion Range – End Address: kết thúc địa chỉ IP cần loại bỏ

+ Chỉ định khoảng thời gian thuê bao (lease duration): thời gian mà DHCP cấp cho máy trạm



Lease Duration – Unlimited: không giới hạn thời gian cấp IP
Lease Duration – Limited:Thời gian IP có hiệu lực

+ Chỉ định các tham số khác
+ Xem và hiệu chỉnh các tham số



9.2. Domain Name System (DNS)

a. Chức năng

DNS là một dịch vụ mạng của Windows 2000, DNS chứa cơ sở dữ liệu phân tán nhằm cung cấp hệ thống tên có thứ bậc để nhận dạng vị trí máy trên mạng, trên Internet, DNS được dùng để quản lý quản trị vùng và tên của các máy tính. DNS xử dụng mô hình máy khách – chủ.

DNS là hệ thống phân giải tên miền được phát minh vào năm 1984, ngày nay nó được coi là một dịch vụ chuẩn của mạng dùng bộ giao thức TCP/IP và Internet. Nhiệm vụ cơ bản của DNS là đổi tên miền thành địa chỉ IP và ngược lại, hỗ trợ hoạt động của các mail server.

 Cấu trúc tên miền:

Bao gồm các ký tự viết cách nhau bởi dấu chấm (.), phần cuối cùng của tên miền thể hiện quốc gia quản lý tên miền này. Hiện nay tồn tại hai dạng tên miền, dạng thứ nhất, phần cuối cùng của tên miền có thể là:

Tên	Mô tả
Com	Tổ chức thương mại
Gov	Tổ chức chính phủ
Mil	Tổ chức quân sự
Net	Các nhà cung cấp mạng và dịch vụ internet
org	Tổ chức phi thương mại và phi lợi nhuận
int	Tổ chức quốc tế
edu	Tổ chức giáo dục

Ví dụ: www.cnn.com website của hãng tin CNN

Tất cả những tên miền này do Hoa kỳ quản lý, dạng thứ hai có phần trước đuôi của tên miền giống như trên, phần cuối cùng là tên viết tắt của các quốc gia. Ví dụ VN là tên viết tắt của Việt nam , JP tên viết tắt của Nhật, ... ví dụ tên miền có dạng bsneu.edu.vn.

b. Một số đặc điểm

 Kiểm soát tại chỗ, truy cập toàn thế giới

Internet không quản lý tập trung tên miền, mỗi tổ chức có mạng vận hành và duy trì DNS server của riêng mình. Cơ quan quản lý internet vùng (InterNIC) quản lý tất cả các tên miền đã được đăng ký, tên và địa chỉ của các DNS server của tên miền đã được đăng ký. InterNIC không cho biết địa chỉ tên miền cụ thể nhưng nó cho biết những DNS server có thể trả lời câu hỏi này.

 Khả năng chịu lỗi

Mỗi miền chỉ có một DNS server chính (primary) chịu trách nhiệm giải đáp tên đối với miền đó. Khi thiết lập một server mới cần khai báo là primary DNS server, hoặc nó chỉ có thể là server phụ (secondary), thông thường DNS server phụ sẽ nối và sao chép cơ sở dữ liệu từ server chính. Trong trường hợp server chính bị lỗi hoặc có quá nhiều yêu cầu thì server phụ có thể đáp ứng yêu cầu tra vấn tin.

 Các zone, các domain, và sự ủy quyền

Zone (khu vực) nó có nhiệm vụ lưu trữ phạm vi các địa chỉ IP mà DNS phải quan tâm. Trong trường hợp domain phải quản lý nhiều server của nhiều khu vực, để đơn giản trong việc tìm kiếm và xác định tên máy đó thuộc khu vực nào, có thể chia tách miền ra thành miền con. Khi chia tách thành miền con thì DNS server cấp trên cùng phải ủy quyền (delegate) cho server cấp dưới nó.

 Các zone tra cứu xuôi và ngược

Quá trình chuyển đổi một tên host thành một địa chỉ IP được gọi là forward name resolution (phân giải tên xuôi). Quá trình chuyển đổi một dịa chỉ IP ra tên host tương ứng được gọi là reverse name resolution (phân giải tên ngược).

 Những loại bản ghi DNS phổ biến

Các bản ghi A (loại host): loại bản ghi chỉ đơn giản liên hệ một tên nào đó với một địa chỉ IP. Đây là tập hợp các tệp tin ASCII có tên là zone files, sử dụng một chữ A để biểu thị bản ghi này là một bản ghi host.

Các bản ghi SOA (start of authority): Nó là bản ghi dành cho tên của DNS server chính của miền, cung cấp địa chỉ IP cho người quản trị của miền, chỉ ra khoảng thời gian đệm trữ dữ liệu.

Các bản ghi NS: dành cho DNS server, qui định các tên server trong miền.

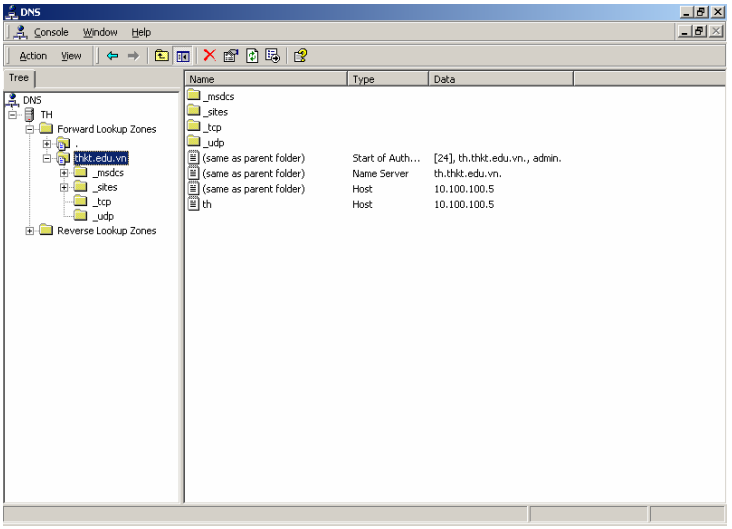
Các bản ghi CNAME: bản ghi alias hay tên kinh điển hay tên chuẩn tắc (canonical name)

Các bản ghi MX: bản ghi mail exchange

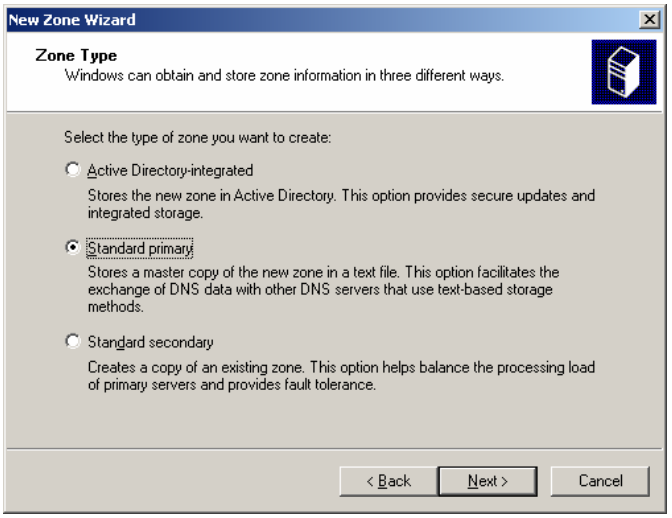
Các bản ghi PTR: bản ghi reverse host, bản ghi thông dụng trong một khu vực tra cứu xuôi.

b. Cài đặt, cấu hình

Khi cài đặt AD trên server DNS được tự động cài đặt theo, DNS cần được hiệu chỉnh các tham số.

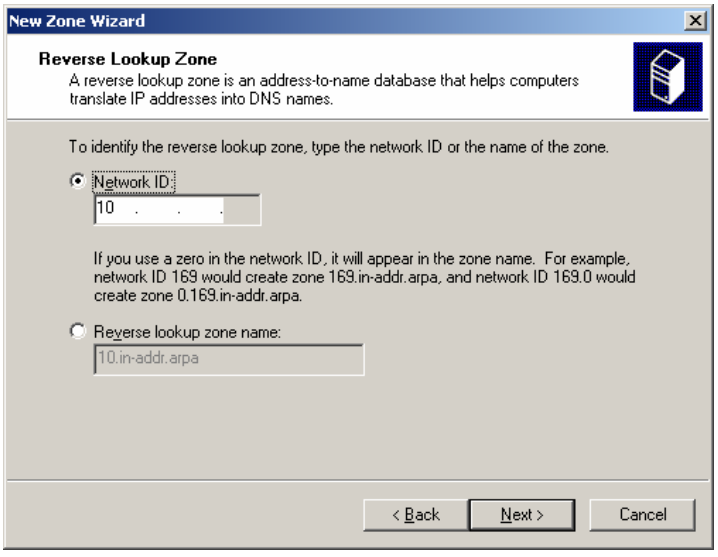


- Tạo ra Reverse lookup zone: chọn add a new zone



+ Activate Directory Integrated: tích hợp với AD
+ Standard Primary: Tạo ra một zone chính
+ Standard secondary: tạo ra một zone phụ

- Nếu chọn Activate Directory Integrated, bước tiếp theo cần điền tham số địa chỉ mạng của lô địa chỉ IP mà mạng đang sử dụng, nhập vào mục Network ID



- Khi mạng chưa có DNS hoặc muốn tạo một dịch vụ cài đặt như sau:
  + Add dịch vụ DNS trong Control Panel – Add/Remove program
  + Chọn Add/Remove Windows Components
  + Chọn Networking Services, chọn Details
  + Chọn DNS
- Điền các tham số mà trình cài đặt yêu cầu

§ 10. TÍCH HỢP NOVELL NETWARE VỚI WINDOWS 2000

X.1. Khả năng liên kết với Novell Netware

Một phần tài nguyên của mạng có thể nằm trên máy chủ Novell Netware, mạng Windows 2000 cần phải giao tiếp và chia sẻ tài nguyên với Novell Netware. Giao thức NWLink của Windows 2000 và dịch vụ Gateway Service for Netware (GSNW) hỗ trợ các máy khách của mạng Microsoft giao tiếp và sử dụng tài nguyên trên mạng Netware.

 Các đặc tính của NWLink

SPX II- NWLink hỗ trợ Windows Sockets trên giao thức Novell SPX II. SPX II được tăng cường để hỗ trợ chế độ cửa sổ và có khả năng đặt kích thước khung lớn nhất.

Da gắn kết (multiple Bindings) – NWLink có thể được gắn vào nhiều vỉ mạng với nhiều kiểu khung khác nhau (Frame)

Tự động phát hiện kiểu khung (Frame Type Auto Detect), có thể cấu hình để trong quá trình cài đặt tự động phát hiện kiểu khung nào là tốt nhất trên mạng và sử dụng nó.

 Kiểu khung và gắn kết

Kiểu khung: Cách mà vỉ mạng định dạng dữ liệu để gửi lên mạng. Các máy Novell có thể được cấu hình với nhiều kiểu khung khác nhau. Để có thể giao tiếp được chúng cần phải được cấu hình với cùng một kiểu khung, ví dụ kiểu khung 802.2, 802.3. Ngoài ra mỗi Topology (Ethernet, Token Ring, ...) đòi hỏi một định dạng kiểu khung khác nhau.

10.2. Gắn kết với NWLink

Dịch vụ gắn kết, liên kết dịch vụ với giao thức và vỉ mạng nó sẽ sử dụng, Dịch vụ này được gắn cả với NWLink NetBIOS và NWLink IPX/SPX, do vậy hỗ trợ cho liên lạc thẳng trên thành phần máy chủ. Dịch vụ máy trạm chỉ được gắn với NWLink NetBIOS không hỗ trợ cho liên lạc thẳng với thành phần máy chủ.

Để quản lý và điều phối hiệu suất NWLink có thể dùng công cụ của Microsoft (Performance Monitor). Công cụ này cho phép hiển thị thông tin về hiệu suất của mạng, theo dõi các đối tượng nào đang sử dụng.

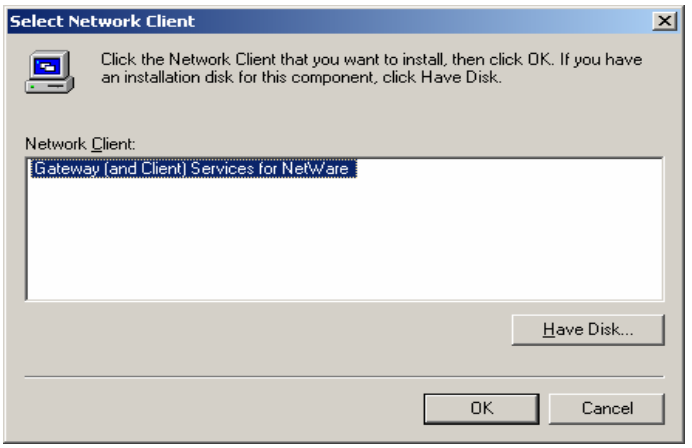
10.3. Cấu hình dịch vụ cổng nối cho Novell

 Trên máy chủ Netware

Dùng tiện ích của Novell tạo nhóm người dùng NTGATEWAY, tạo user và các tham số cần thiết, bổ sung tài khoán cổng giao tiếp vào nhóm NTGATEWAY. Gắn các quyền cần thiết để các user thuộc nhóm này có thể sử dụng tài nguyên của Novell.

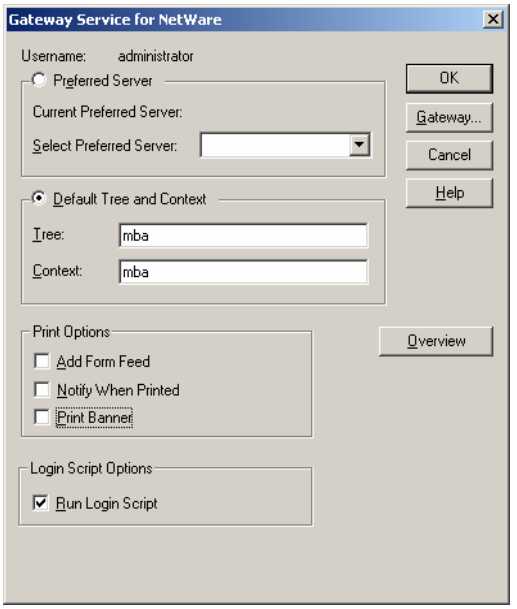
 Trên máy Windows 2000

Cài đặt dịch vụ GSNW: add dịch vụ trong Control Panel – Network



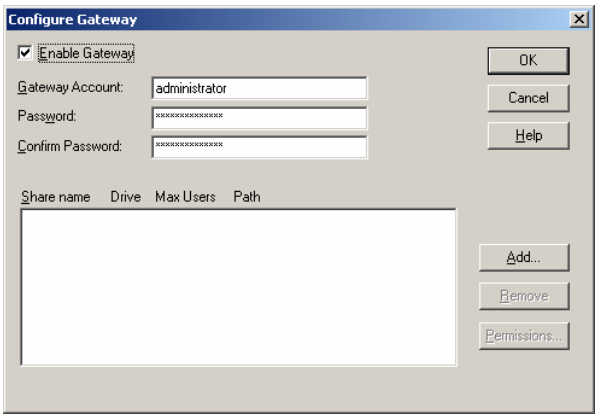
Cài đặt, hiệu chỉnh các tham số

Trong Control Panel chọn biểu tượng GSNW, hiệu chỉnh các tham số sau



Lựa chọn	Nội dung
Prerred Server	Máy chủ Novell sẽ được tự động nối đến
Default Tree	Nhập các thông số của cây thư mục NDS
Default Context	Nhập ngữ cảnh ngầm định
Run Login Script	Chạy Login script
Gateway	Cấu hình cổng nối cho các tài nguyên

Cấu hình cho Gateway



Đánh dấu vào mục Enable Gateway và điền các tham số về user

Chọn ADD để cấu hình đường dẫn đến tài nguyên trên Novell

Điền đường dẫn đầy đủ trong hộp Network Path của máy chủ Novell

§ 11. MỘT SỐ TIỆN ÍCH

a. Ipconfig

- Có trên Windows 9x, Windows 2000

Lệnh này hiển thị các giá trị cấu hình mạng TCP/IP đang hoạt động. Lệnh này đặc biệt có ý nghĩa đối với mạng có sử dụng DHCP, nó cho phép người sử dụng xác định cấu hình TCP/IP mà được cung cấp bởi DHCP

ipconfig [**/all** | **/renew** [*adapter*] | **/release** [*adapter*]]

Tham số

All : Hiển thị đấy đủ. Bỏ tham số này, **ipconfig** sẽ chỉ hiển thị địa chỉ IP, subnet mask, và default gateway.

/renew [*adapter*] : Thay mới cấu hình cung cấp bởi DHCP. Lựa chọn này chỉ tồn tại trên hệ thống chạy dịch vụ DHCP client. Để chỉ định card, đánh tên card khi sử dụng **ipconfig** mà không cần đánh tham số.

/release [*adapter*] Bỏ cấu hình DHCP hiện thời. Mục chọn này ngừng hoạt động của TCP/IP trên hệ thống và chỉ tồn tại trên các hệ thống chạy DHCP client. Để chỉ định card, đánh tên card khi sử dụng **ipconfig** mà không cần đánh tham số.

Không điền tham số, **ipconfig** sẽ hiển thị toàn bộ cấu hình TCP/IP hiện tại bao gồm địa chỉ IP và subnet mask.

b. Ping

Kiểm tra kết nối tới một hệ thống khác. Lệnh này chỉ tồn tại nếu TCP/IP được cài đặt.

ping [**-t**] [**-a**] [**-n** *count*] [**-l** *length*] [**-f**] [**-i** *ttl*] [**-v** *tos*] [**-r** *count*] [**-s** *count*] [[**-j** *computer-list*] | [**-k** *computer-list*]] [**-w** *timeout*] *destination-list*

Tham số

-t : Chỉ ngừng khi bị ngắt bởi người sử dụng.

-a : Tìm tên máy khi biết địa chỉ IP.

-n *count*

Gửi một số lần nhất định bó dữ liệu ECHO, số lần được xác định bởi count, ngầm định là 4 lần.

-l *length :* Gửi dữ liệu ECHO với độ dài được xác định bởi length, ngầm định là 32 byte, tối đa là 65527.

-f : Gửi lệnh với cờ hiệu "Do not Fragment" trong bó dữ liệu. Bó dữ liệu sẽ không bị chia tách trong khi chuyển tới đích.

-i *ttl :* Đặt trường "Time To Live" giá trị chỉ định ở tham số *ttl*.

-v *tos :* Đặt trường "Type Of Service" giá trị chỉ định bởi *tos*.

-r *count :*Ghi lại đường đi của bó dữ liệu và bó dữ liệu trả về được đặt ở trường "Record Route" . Nhỏ nhất là 1 và lớn nhất là 9 đối với tham số count

-s *count :* Chỉ rõ thời gian đối với mỗi hop bởi count.

-j *computer-list :* Chỉ rõ đường đi qua danh sách các máy chỉ định bởi computer-list. Các địa chỉ máy tiếp được ngăn cách bởi các gateway. Số lượng tối đa danh sách là 9 IP.

-k *computer-list :*Chỉ rõ đường đi qua danh sách các máy chỉ định bởi computer-list. Các địa chỉ máy tiếp không được ngăn cách bởi các gateway. Số lượng tối đa danh sách là 9 IP.

-w *timeout :* Chỉ rõ khoảng thời gian ngắt dưới dạng milliseconds.

destination-list : Chỉ rõ các máy sẽ được ping.

c. Tracert

Chương trình này xác định đường đi tới một địa chỉ bằng cách gửi bó dữ liệu kiểu Internet Control Message Protocol (ICMP) với các giá trị Time-To-Live (TTL) khác nhau. Mỗi một router trên đường đi qua của bó dữ liệu sẽ giảm biến TTL của bó dữ liệu ít nhất là 1 đơn vị trước khi chuyển tiếp bó dữ liệu, do đó, biến TTL rất hữu ích khi dùng để đếm các hop. Khi giá trị của TTL của bó dữ liệu giảm còn 0, router sẽ gửi trả hệ thống một message "ICMP Time Exceeded". **Tracert** xác định đường đi bằng cách gửi bó dữ liệu đầu tiên với giá trị của TTL là 1 và sau đó sẽ tăng TTL lên 1 sau mỗi lần gửi tiếp theo cho đến khi hệ thống đích trả lời hoặc TTL tiến tới giá trị tối đa. Đường đi được xác định bởi kiểm tra message "ICMP Time Exceeded" mà được gửi trả lại bởi các router

trung gian. Tuy nhiên, một số router hủy các bó dữ liệu mà không gửi lại message khi TTL về không sẽ làm cho **tracert** không xác định được.

tracert [**-d**] [**-h** *maximum_hops*] [**-j** *computer-list*] [**-w** *timeout*] *target_name*

Tham số

-d : Chỉ rõ không tìm tên máy.

-h *maximum_hops :* Chỉ rõ số hop tối đa trước khi tới hệ thống đích.

-w *timeout :* Chỉ rõ thời gian đợi của mỗi lần trả về thông tin trả lời theo milliseconds .

target_name : Tên của hệ thống đích.

d. Ftp

Tải file về máy hoặc tải sang một hệ thống chạy dịch vụ FTP server. **Ftp** can be used interactively. Lệnh này chỉ tồn tại nếu giao thức TCP/IP đã được cài đặt. FTP là 1 dịch vụ khi chạy tạo ra một môi trường mà bạn có thể sử dụng các lệnh con của FTP, sau đó có thể thoát ra bằng lện **quit**. Khi đang trong môi trường của FTP thì dấu nhắc **ftp** sẽ xuất hiện.

ftp [**-v**] [**-n**] [**-i**] [**-d**] [**-g**] [**-s:***filename*] [**-a**] [**-w:***windowsize*] [*computer*]

Tham số

-v : Bỏ qua các dòng trả lời từ server **ftp**

-n : Không tự động yêu cầu login sau khi chạy **ftp**.

-i : Bỏ các dòng thông báo trong khi chuyển nhiều file.

-d : Cho phép dò lỗi, hiển thị tất cả các lệnh được thực hiện giữa ứng dụng khách ftp và server ftp.

-s:*filename :* Chỉ rõ 1 file dạng text chứa các lệnh ftp; các lệnh này sẽ chạy sau khi ftp khởi động.

Computer : Chỉ rõ tên máy hoặc IP của hệ thống từ xa cần truy cập. Nếu có tham số này thì nó phải là tham số đứng cuối cùng trong dòng lệnh.

e. Finger

Hiện tất cả các thông tin về người dùng trên một hệ thống chạy dịch vụ Finger. Các thông tin hiển thị sẽ thay đổi tuỳ theo hệ thống từ xa. Lệnh này chỉ tồn tại khi có cài TCP/IP.

finger [**-l**] [*user*]@*computer* [**...**]

Tham số

-l : Hiển thị liệt kê dài.

User : Tên người sử dụng cần biết. Nếu không chỉ rõ thì thông tin của tất cả người sử dụng sẽ hiển thị

Access permission	Quyền truy cập
Account lockout	Khoá tài khoản
Account policy	Chính sách tài khoản
Address	Địa chỉ
Address class	Lớp địa chỉ
Administrative account	Tài khoản quản trị
Administrative alert	Báo động quản trị
Administrator	Nhà quản trị
Administrator privilege	Đặc quyền (ở cấp độ) nhà quản trị; đặc quyền Administrator
Archive bit	Bít lưu trữ
Associate	Phối hợp
Attribute	Thuộc tính
Auditing	Kiểm toán
Audit policy	Chính sách Audit
Backup domain controller (BDC);	Máy điều khiển vùng dự phòng
Bits per second (BPS)	Số bít/giây.
Boot partition	Phần đĩa khởi động
Bridge	Cầu nối
Broadcast message	Thông điệp phát rộng
Brouter	Kết hợp các phần tử của cầu nối và bộ định tuyến.
Browse	Duyệt xem
Built-in group	Nhóm cài sẵn
Capture	Chụp ảnh
Client	Máy khách
Client application	Ứng dụng máy khách.
Client service for Netware	Client Service for Netware
Common group	Nhóm chung
Communications setting	Xác lập truyền thông
Compact	Tiện ích dạng dòng lệnh, dùng để nén tập tin trên volume NTFS.
Computer account	Tài khoản máy tính.
Computer browser server	Dịch vụ Computer Browser.
Computer name	Tên máy tính
Configure	Lập cấu hình
Connect	Nối kết
Connected user	Người dùng được nối kết
Connection oriented protocol	Giao thức hướng nối kết
Default gateway	Cổng giao tiếp mặc định

Default network	Mạng mặc định
Default user	Người dùng mặc định
Dial-up line	Đường truyền quay số
Dial-up networking	Nối mạng qua đường quay số
Disabled user account	Tài khoản người dùng bị cấm sử dụng
DNS name server	Máy phục vụ tên DNS
DNS service	Dịch vụ DNS
Domain	Vùng
Domain controller	Máy điều khiển vùng
Domain name	Tên vùng
Domain name system (DNS)	Hệ thống tên vùng
Dynamic host configuration protocol (DHCP)	Giao thức cung cấp cấu hình động của địa chỉ IP và các thông tin liên quan
Extended partition	Phần chia mở rộng
Fault tolerance	Cơ chế dung lỗi
File allocation table (FAT)	Bảng phân phối tệp tin
File sharing	Chia sẻ tệp tin
File system	Hệ thống tệp tin
File transfer protocol (FTP)	Dịch vụ hỗ trợ việc chuyển tải tệp tin giữa những hệ thống cục bộ và ở xa nào chấp nhận giao thức này.
Gateway	Cổng giao tiếp
Geteway service for Netware	Được kèm theo Windows NT Server, W2K cho phép máy tính chạy Windows NT, W2K kết nối với máy phục vụ Netware.
Global account	Tài khoản toàn cục
Global group	Nhóm toàn cục
Gopher	Hệ thống thứ bậc để tìm và phục hồi thông tin từ Internet hoặc một Intranet nào đó.
Group	Nhóm
Group account	Tài khoản nhóm
Guest	Người dùng vãng lai, khách vãng lai
Home Directory	Thư mục cá nhân
Host	Máy chủ
Home page	Trang chủ
Host ID	Số nhận diện máy chủ
Internet Protocol (IP)	Giao thức truyền thông điệp của dãy giao thức TCP/IP

Internet Router	Bộ định tuyến Internet
IP address	Địa chỉ IP
IP Router	Bộ định tuyến IP
ISO	Viết tắt từ International Standards Organization (Tổ chức tiêu chuẩn hoá quốc tế)
Kermit	Giao thức chuyển tải các tệp tin nhị phân
Kernel	Thành phần của Windows NT Excutive W2K, chịu trách nhiệm quản lý bộ vi xử lý.
Local account	Tài khoản cục bộ
Local group	Nhóm cục bộ
Local printer	Máy in cục bộ
Localtalk	Ten do Apple Computer dùng để gọi phần cứng nối mạng Apple
Log off	Tách khỏi mạng
Log on	Truy nhập
Logon script	Script đăng nhập, kịch bản đăng nhập
Mac address	Địa chỉ MAC
Macintosh accessible volume	Volume có thể truy cập bởi macintosh
Mapping	Sự ánh xạ
Mapping file	Tệp tin ánh xạ
Master Boot Record	Vùng quan trọng trên đĩa cứng, chứa cấu trúc dữ liệu bắt đầu quy trình khởi động máy tính
Master domain	Vùng chính
Member server	Máy phục vụ thành viên
Migration Tool for Netware	Được cung cấp kèm theo Windows NT, cho phép chuyển tải các tài khoản và các thông tin liên quan đến máy tính chạy Windows NT Server, W2K
Mirror set	Một bản sao hoàn chỉnh của dữ liệu
Modem	Modulator/Demodulator, thiết bị truyền, nhận dữ liệu qua đường điện thoại chuẩn
Name mapping	Ánh xạ tên
Name Resolution Service	Dịch vụ phân giải tên
Net Logon Service	Dịch vụ Net logon
Netware Directory Services (NDS)	Một dịch vụ Netware chạy trên các

	máy phục vụ Netware, dịch vụ này cho phép định rõ địa điểm của tài nguyên trên mạng (từ Netware 4.x)
Network adapter card	Thiết bị dùng nối mạng máy tính (card mạng)
Network ID	Số nhận diện mạng
Network protocol	Giao thức mạng
Network sniffer	Công cụ giải đoán mạng
Node	Nút, điểm nối
NT file system, NTFS	Hệ thống tệp tin NT, W2K
Object	Đối tượng
Open Systems Interconnection (OSI) model	Mô hình OSI
Orphan	Thành viên mồ côi
Owner	Chủ sở hữu
Packet	Gói dữ liệu
Page fault	Lỗi trang nhớ
Paging file	Tệp tin phân trang
Parity	(thông tin) chẵn lẻ
Partition	Phần chia (đĩa)
Partition Table	Bảng quản lý đĩa
Permission	Quyền truy cập
Personal group	Nhóm cá nhân
Ping	Lệnh Ping, có thể dùng để kiểm tra các kết nối với một hay nhiều máy chủ ở xa
Point-to-Point protocol (PPP)	Tập hợp giao thức và tạo khung theo chuẩn công nghiệp, là thành phần của Windows 2000
Port ID	Số nhận diện cổng
Primary domain controller (PDC)	Máy điều khiển vùng chính
Primary Partition	Phần chia chính
Printer permission	Quyền truy cập máy in
Print sharing	Chia sẻ máy in
Print spooler	Bộ đệm in
Privilege level	Cấp độ đặc quyền
Property	Thuộc tính
Proxy	Máy được uỷ nhiệm
Queue	Hàng chờ
Raid (Redundant Array off Inexpensive Disks)	Một phương pháp dùng để chuẩn hóa và phân loại các hệ dung lỗi đĩa.

Remote Access Service (RAS)	Dịch vụ truy cập từ xa
Remote Procedure Call (RPC)	Cuộc gọi thủ tục từ xa
Repeater	Bộ đổi hướng
Resource	Tài nguyên
Roaming user profile	Hệ lưu trữ người dùng lưu động
Router	Bộ định tuyến
Security Accounts Manager (SAM)	Một cơ sở dữ liệu chứa thông tin bảo mật của Windows NT, W2K
Secure Sockets Layer (SSL)	Giao thức cung cấp truyền thông dữ liệu an toàn thông qua cơ chế mã hoá và giải mã dữ liệu
Server message block (SMB)	Khối thông điệp máy phục vụ
Server zone	Khu vực máy phục vụ
Session	Phiên, kỳ làm việc
Share	Chia sẻ
Share resource	Tài nguyên dùng chung
Simple Mail Transfer protocol (SMTP)	Giao thức chuyển thư giản đơn
Simple Network Management Protocol (SNMP)	Giao thức quản lý mạng giản đơn
Acronym for structured query language (SQL)	Ngôn ngữ hỏi có cấu trúc
System partition	Phần chia hệ thống
System police	Chính sách hệ thống
Transforms	Nguyên tắc biến đổi
Trap	Bẫy
True Type Font	Một kiểu phông chữ
Trust relationship	Quan hệ uỷ quyền
Uninterruptible power supply (UPS)	Bộ cung cấp điện năng liên tục
User account database	Cơ sở dữ liệu tài khoản người dùng
User Datagram Protocol (UDF)	Giao thức bó dữ liệu người dùng
User right policy	Chính sách về quyền người dùng
Variables	Biến
Virtual server	Máy phục vụ ảo

TÀI LIỆU THAM KHẢO

1. Mạng máy tính và các hệ thống mở, GS. TS. Nguyễn Thúc Hải, NXB Giáo dục 1999

2. Mạng máy tính, Lược dịch và biên soạn Hồ Anh Phong Nhà xuất bản Thống kê, 2002

3. Cốt tủy về mạng (Networking Essentials) , biên dịch Phạm Cao Hoàn, Phạm Đình Phước, Nguyễn Văn Khôi, NXB Đồng Nai 2000

4. Upgrading and Repairing Networks, Craig Zacker, Publisher: Que ,1/1995

5. High-Performance Networking Unleashed, Macmillan Computer Publishing

CNE Training Guide Netware 4.1 Administration, KARANJIT SIYAN, PH.D. Publisher: New Riders, Sep-1995

6. Novell Netware 4.1, Novell, Inc. December 1994

7. Làm chủ Windows 2000 Server, Phạm Hoàng Dũng, Hoàng Đức Hải, NXB Giáo dục 2000

8. Microsoft Windows 2000 Server Unleashed, Sams Publishing 2000

9. Windows 2000 Server Study Guide, SYBEC 2000

10. Administrating W2K Network Infrastructure, SYBEC 2000

11. Windows 2000 Server, Syngress 2001

12. Các trang web: www.micorosoft.com , www.novell.com, www.cisco.com,