Sổ tay

Mạng máy tính

MỤC LỤC

CHƯƠNG 1. NHẬP MÔN MẠNG MÁY TÍNH

1.1. MỞ ĐẦU

Mạng máy tính phát sinh từ nhu cầu muốn chia sẻ, dùng chung tài nguyên và cho phép giao tiếp trực tuyến (online) cũng như các ứng dụng đa phương tiện trên mạng. Tài nguyên gồm có tài nguyên phần mềm (dữ liệu, chương trình ứng dụng, ...) và tài nguyên phần cứng (máy in, máy quét, CD ROM,.). Giao tiếp trực tuyến bao gồm gửi và nhận thông điệp, thư điện tử. Các ứng dụng đa phương tiện có thể là phát thanh, truyền hình, điện thoại qua mạng, hội thảo trực tuyến, nghe nhạc, xem phim trên mạng.

Trước khi mạng máy tính được sử dụng, người ta thường phải tự trang bị máy in, máy vẽ và các thiết bị ngoại vi khác cho riêng mình. Để có thể dùng chung máy in thì mọi người phải thay phiên nhau ngồi trước máy tính được nối với máy in. Khi được nối mạng thì tất cả mọi người ngồi tại các vị trí khác nhau đều có quyền sử dụng máy in đó.

Sự kết hợp của máy tính với các hệ thống truyền thông, đặc biệt là viễn thông, đã tạo ra cuộc cách mạng trong vấn đề tổ chức khai thác và sử dụng hệ thống máy tính. Mô hình tập trung dựa trên máy tính lớn được thay thế mô hình các máy tính đơn lẻ được kết nối lại để cùng thực hiện công việc, hình thành môi trường làm việc nhiều người sử dụng phân tán, cho phép nâng cao hiệu quả khai thác tài nguyên chung từ những vị trí địa lý khác nhau. Các hệ thống như thế được gọi là *mạng máy tính*.

Mạng máy tính ngày nay đã trở thành một lĩnh vực nghiên cứu phát triển và ứng dụng cốt lõi của Công nghệ thông tin. Các lĩnh vực nghiên cứu phát triển và ứng dụng của mạng: kiến trúc mạng, nguyên lý thiết kế, cài đặt và các ứng dụng trên mạng.

1.2. CÁC KHÁI NIỆM CƠ BẢN

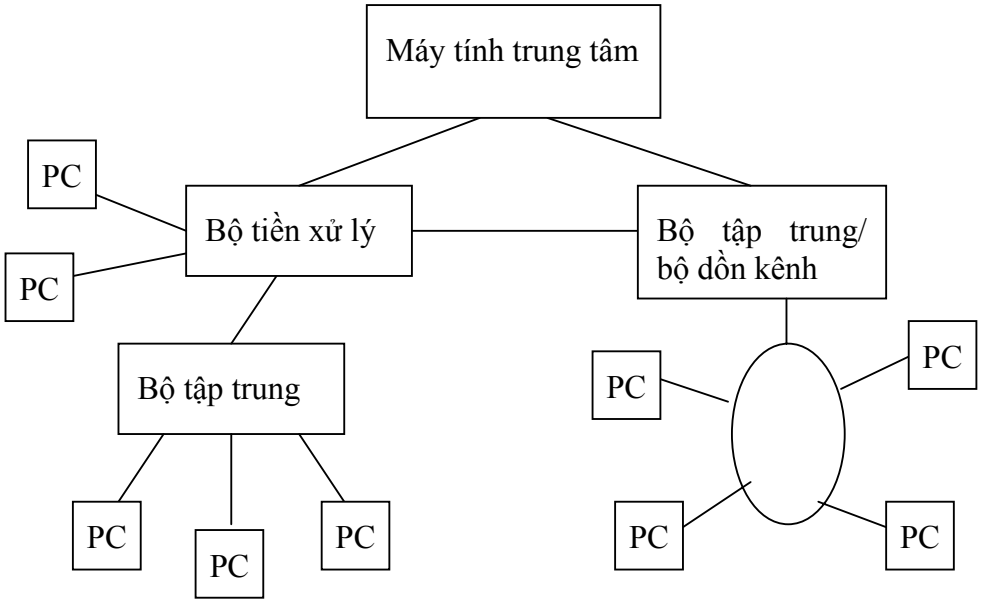
1.2.1. Lịch sử phát triển

Cuối những năm 60 đã xuất hiện các mạng xử lý gồm các *trạm cuối* (terminal) thụ động được nối vào một máy xử lý trung tâm. Máy tính trung tâm hầu như đảm nhiệm tất cả mọi việc từ xử lý thông tin, quản lý các thủ tục truyền dữ liệu, quản lý sự đồng bộ của các trạm cuối, quản lý các hàng đợi, xử lý các ngắt từ các trạm cuối,... Mô hình này bộc lộ các yếu điểm như: tốn quá nhiều vật liệu (đường truyền) để nối các trạm với trung tâm, máy tính trung tâm phải làm việc quá nhiều dẫn đến quá tải.

Để giảm nhẹ nhiệm vụ của máy tính trung tâm người ta gom các trạm cuối vào bộ gọi là *bộ tập trung* (hoặc *bộ dồn kênh*) trước khi chuyển về trung tâm. Các bộ này có chức năng tập trung các tín hiệu do trạm cuối gửi đến vào trên cùng một đường truyền. Sự khác nhau giữa hai thiết bị này thể hiện ở chỗ:

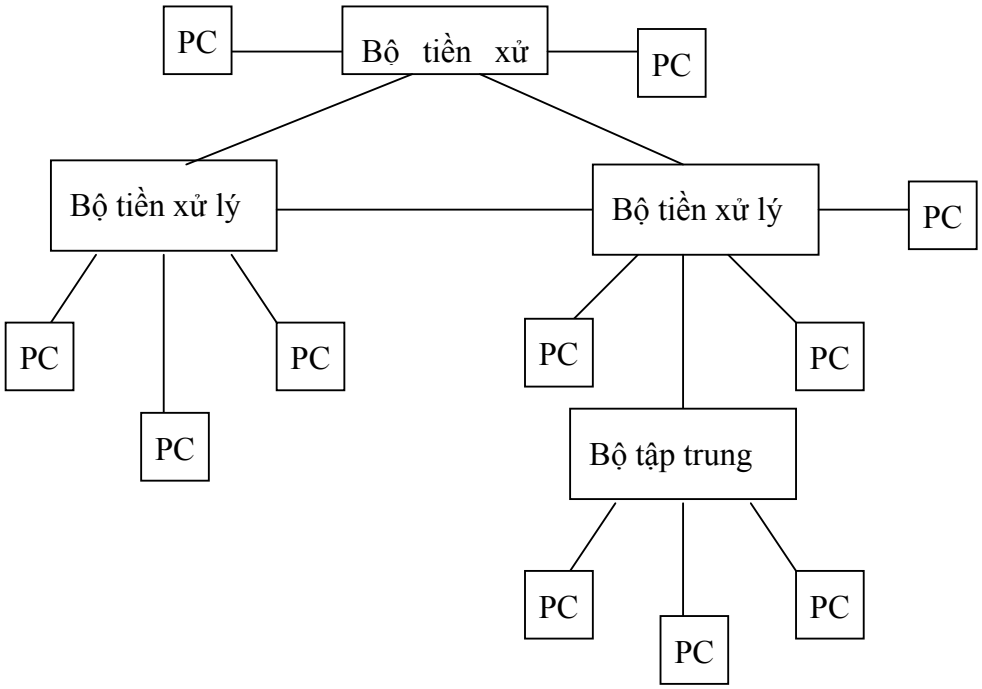
- Bộ dồn kênh (multiplexor): có khả năng truyền song song các thông tin do trạm cuối gửi về trung tâm.
- Bộ tập trung (concentrator): không có khả năng này, phải dùng bộ đệm để lưu trữ tạm thời dữ liệu

Trong hệ thống này, mọi sự liên lạc giữa các trạm cuối với nhau phải đi qua máy tính trung tâm, không được nối trực tiếp với nhau →hệ thống trên không được gọi là mạng máy tính mà chỉ được gọi là *mạng xử lý* (hình 1.1).



Hình 1.1. Mạng xử lý với các bộ tiền xử lý

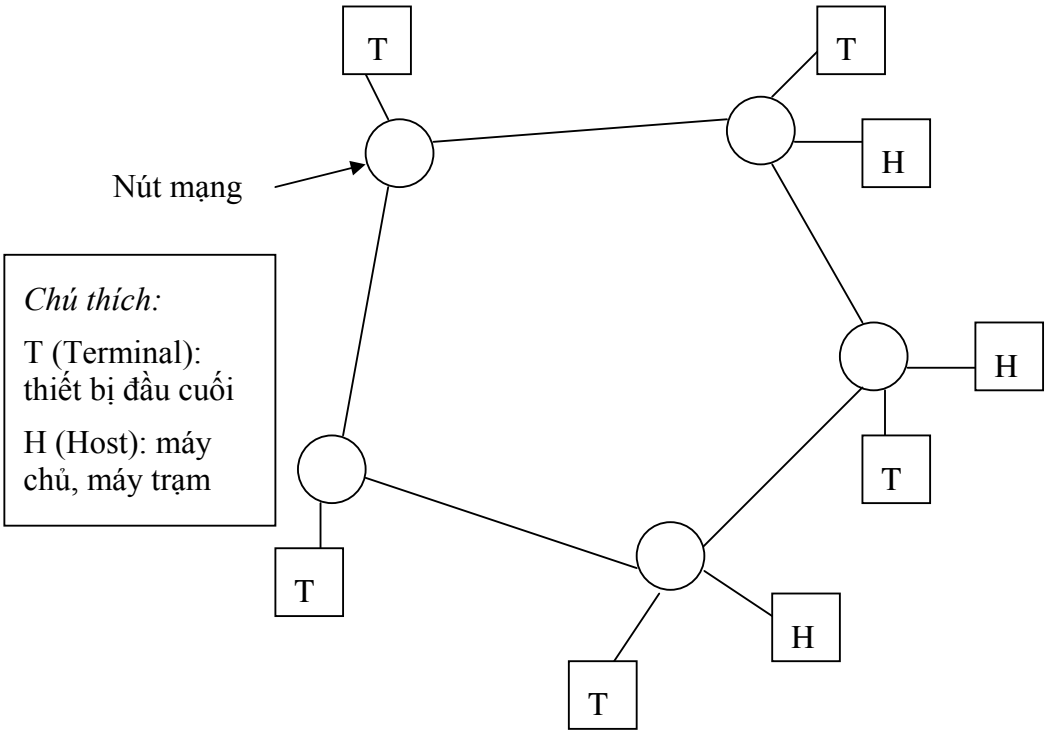
Từ cuối những năm 70, các máy tính được nối trực tiếp với nhau để tạo thành mạng máy tính nhằm phân tán tải của hệ thống và tăng độ tin cậy.



Hình 1.2. Mạng máy tính- nối trực tiếp các bộ tiền xử lý

Cũng những năm 70 xuất hiện khái niệm mạng truyền thông (communication network), trong đó các thành phần chính của nó là các nút mạng (Node), được gọi là bộ chuyển mạch (switching unit) dùng để hướng thông tin tới đích.

Các nút mạng được nối với nhau bằng đường truyền gọi là khung của mạng. Các máy tính xử lý thông tin của người sử dụng (host) hoặc các trạm cuối (terminal) được nối trực tiếp vào các nút mạng để khi cần thì trao đổi thông tin qua mạng. Bản thân các nút mạng thường cũng là máy tính nên có thể đồng thời đóng cả vai trò máy của người sử dụng. Vì vậy chúng ta không phân biệt khái niệm mạng máy tính và mạng truyền thông. (Xem hình 1.3).



Hình 1.3. Một mạng truyền thông

Các máy tính được kết nối thành mạng nhằm đạt các mục đích sau:

- Chia sẻ các tài nguyên có giá trị cao (thiết bị, chương trình, dữ liệu,..) không phụ thuộc vào khoảng cách địa lý của tài nguyên và người sử dụng.
- Tăng độ tin cậy của hệ thống: do có khả năng thay thế khi xảy ra sự cố đối với một máy tính nào đó.

1.2.2. Các yếu tố của mạng máy tính

Mạng máy tính có thể được định nghĩa: mạng máy tính là tập hợp các máy tính được nối với nhau bởi các đường truyền vật lý theo một kiến trúc nào đó. Như vậy có hai khái niệm mà chúng ta cần phải làm rõ, đó là đường truyền vật lý và kiến trúc của một mạng máy tính.

1.2.2.1. Đường truyền vật lý

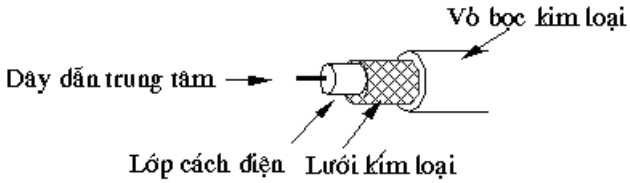
Đường truyền vật lý dùng để chuyển các tín hiệu giữa các máy tính. Các tín hiệu đó biểu thị các giá trị dữ liệu dưới dạng các xung nhị phân (on - off). Tất cả các tín hiệu đó đều thuộc dạng sóng điện từ (trải từ tần số sóng radio, sóng ngắn, tia hồng ngoại). Ứng với mỗi loại tần số của sóng điện tử có các đường truyền vật lý khác nhau để truyền tín hiệu.

Hiện nay có hai loại đường truyền:

+ Đường truyền hữu tuyến: cáp đồng trục, cáp đôi dây xoắn (có bọc kim, không bọc kim), cáp sợi quang.

+ Đường truyền vô tuyến: radio, sóng cực ngắn, tia hồng ngoại.

✧ Cáp đồng trục dùng để truyền các tín hiệu số trong mạng cục bộ hoặc làm mạng điện thoại đường dài. Cấu tạo gồm có một sợi kim loại ở trung tâm được bọc bởi một lớp cách điện và một lưới kim loại chống nhiễu. Ở ngoài cùng là vỏ bọc cách điện. Sợi kim loại trung tâm và lưới kim loại làm thành hai sợi dẫn điện đồng trục



Hình 1.4. Cáp đồng trục

Có hai loại cáp đồng trục khác nhau với những chỉ định khác nhau về kỹ thuật và thiết bị ghép nối đi kèm: cáp đồng trục mỏng (giá thành rẻ, dùng phổ biến), cáp đồng trục béo (đắt hơn, có khả năng chống nhiễu tốt hơn, thường được dung liên kết mạng trong môi trường công nghiệp).

✧ Cáp đôi dây xoắn: được sử dụng rộng rãi trong các mạng điện thoại có thể kéo dài hàng cây số mà không cần bộ khuyếch đại. Cấu tạo gồm nhiều sợi kim loại cách điện với nhau. Các sợi này từng đôi một xoắn lại với nhau nhằm hạn chế nhiễu điện từ. Có hai loại cáp xoắn đôi được sử dụng hiện nay: cáp có bọc kim loại (STP), cáp không bọc kim loại (UTP).

✧ Cáp sợi quang: là cáp truyền dẫn sóng ánh sáng, có cấu trúc tương tự như cáp đồng trục với chất liệu là thuỷ tinh. Tức là gồm một dây dẫn trung tâm (một hoặc một bó sợi thuỷ tinh hoặc plastic có thể truyền dẫn tín hiệu quang) được bọc một lớp áo có tác dụng phản xạ các tín hiệu trở lại để giảm sự mất mát tín hiệu. Có hai loại cáp sợi quang là: single-mode (chỉ có một đường dẫn quang duy nhất), multi-mode (có nhiều đường dẫn quang). Cáp sợi quang có độ suy hao tín hiệu thấp, không bị ảnh hưởng của nhiễu điện từ và các hiệu ứng điện khác, không bị phát hiện và thu trộm, an toàn thông tin trên mạng được bảo đảm. Tuy nhiên cáp sợi quang khó lắp đặt, giá thành cao.

✧ Sóng cực ngắn thường được dùng để truyền giữa các trạm mặt đất và các vệ tinh. Chúng để truyền các tín hiệu quảng bá từ một trạm phát tới nhiều trạm thu.

✧ Sóng hồng ngoại: Môi trường truyền dẫn sóng hồng ngoại là một môi trường định hướng, trong diện hẹp vì vậy nó chỉ thích hợp cho một mạng diện hẹp bán kính từ 0.5m đến 20 m, với các thiết bị ít bị di chuyển. Tốc độ truyền dữ liệu xung quanh 10Mbps

✧ Sóng radio: môi trường truyền dẫn sóng radio là một môi trường định hướng trong mạng diện rộng với bán kính 30 km. Tốc độ truyền dữ liệu hàng chục Mbps.

Liên quan đến đường truyền vật lý chúng ta có các khái niệm sau:

- *Băng thông* (còn gọi là dải thông - bandwidth): Băng thông là một khái niệm cực kỳ quan trọng trong các hệ thống truyền thông. Hai phương pháp xem xét băng thông có tầm quan trọng trong nghiên cứu các mạng là băng thông tương tự (analog) và băng thông số (digital). Băng thông tương tự là độ đo phạm vi tần số mà đường truyền có thể đáp ứng được trong một hệ thống điện tử dùng kỹ thuật tương tự. Đơn vị đo lường cho băng thông tương tự là Hz, hay số chu kỳ trên giây. Ví dụ, băng thông của cáp điện thoại là 400-4000Hz, có nghĩa là nó có thể truyền các tín hiệu với các tần số nằm trong phạm vi từ 400 đến 4000Hz. Băng thông số đo lường lượng thông tin tối đa từ nơi này đến nơi khác trong một thời gian cho trước. Đơn vị cơ bản đo lường băng thông số là bít/giây (bps) và các bội của nó là Kilôbit/giây (kbps), Megabit/giây (Mbps), Gigabit/giây (Gbps), Terabit/giây (Tbps).. Băng thông của cáp truyền phụ thuộc vào độ dài cáp. Cáp càng dài thì băng thông càng giảm. Do vậy khi thiết kế mạng phải chỉ rõ độ dài chạy cáp tối đa, bởi vì ngoài giới hạn dố thì chất lượng truyền tín hiệu không còn được bảo đảm.

- *Thông lượng* (throughput): thông lượng là lượng thông tin thực sự được truyền qua trong một đơn vị thời gian. Cũng như băng thông, đơn vị của thông lượng là bps và các bội của nó: Kbps, Mbps, Gbps, Gbps, Tbps. Trong một mạng LAN băng thông có thể cho phếp 100Mbps, nhưng điều này không có nghĩa là mỗi người dùng trên mạng đều có thể di chuyển thực sự 100 Megabit dữ liệu trong một giây. Điều này chỉ đúng trong những điều kiện vô cùng lý tưởng. Do nhiều lý do, thông lượng thường nhỏ hơn rất nhiều so với băng thông số tối đa của môi trường mạng.

- *Hiệu suất sử dụng đường truyền* (utilization): Đại lượng này đặc trưng cho hiệu suất phục vụ của đường truyền trong mạng. Nó được đo bằng tỷ lệ % giữa thông lượng và băng thông của đường truyền.

- *Độ trễ* (delay): độ trễ là thời gian cần thiết để truyền một gói tin từ nguồn đến đích. Độ trễ thường được đo bằng miligiây (ms), giây (s). Độ trễ phụ thuộc vào băng thông của mạng. Băng thông càng lớn thì độ trễ càng nhỏ.

- *Độ suy hao* là độ đo sự yếu đi của tín hiệu trên đường truyền. Nó cũng phụ thuộc vào độ dài cáp. Còn *độ nhiễu từ* gây ra bởi tiến ồn điện từ bên ngoài làm ảnh hưởng đến tín hiệu trên đường truyền.

1.2.2.2. Kiến trúc mạng máy tính

Kiến trúc mạng máy tính (network architecture) thể hiện *cách nối* các máy tính với nhau ra sao 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.

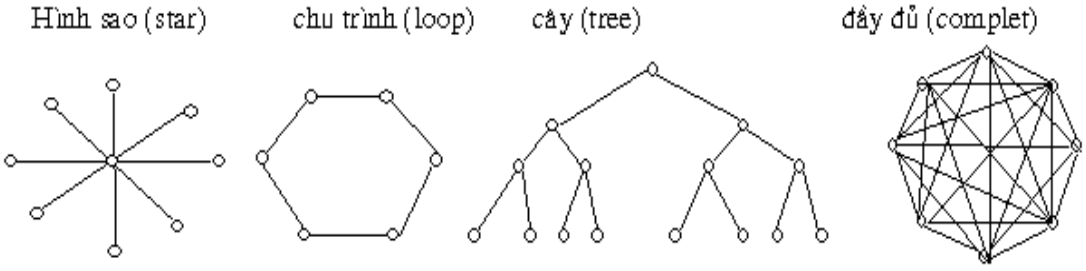
Cách nối các máy tính được gọi là hình trạng (topolopy) của mạng hay nói cho gọn là topo mạng. Còn tập hợp các quy tắc, quy ước truyền thông được gọi là giao thức (protocol) của mạng. Topo và giao thức là hai khái niệm rất cơ bản của mạng máy tính, vì thế chúng sẽ được trình bày cụ thể hơn trong những phần sau:

- **Topo mạng**

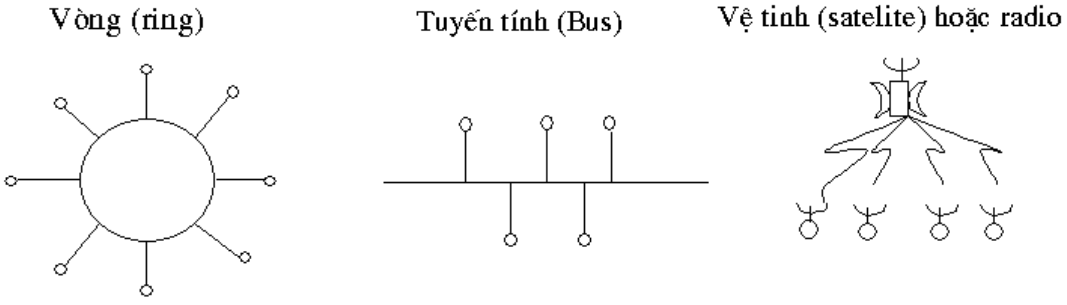
Có hai kiểu kết nối mạng chủ yếu là điểm - điểm (point-to-point) và quảng bá (broadcast hay point-to-multipoint).

Theo kiểu kết nối điểm - điểm, các đường truyền nối từng cặp nút với nhau và mỗi nút đều có trách nhiệm lưu trữ tạm thời sau đó chuyển tiếp dữ liệu đi tới đích. Do cách làm việc như thế nên mạng kiểu này còn được gọi là mạng lưu và chuyển tiếp (store and forward). Nói chung các mạng diện rộng đều sử dụng nguyên tắc này. Hình 2 cho một số dạng topo của mạng loại này.

Theo kiểu quảng bá, tất cả các nút mạng dùng chung một đường truyền vật lý. Dữ liệu gửi đi từ một nút mạng có thể được tất cả các nút mạng còn lại tiếp nhận → chỉ cần chỉ ra địa chỉ đích của dữ liệu để mỗi nút kiểm tra xem có phải là gửi cho mình hay không. Hình 3 cho một số dạng topo của mạng loại này.



Hình 1.5. Một số topo mạng điểm-điểm



Hình 1.6. Một số topo mạng quảng bá

Trong các topo dạng vòng hoặc dạng tuyến tính cần có một 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 "động" hoặc "tĩnh". Cấp phát "tĩnh" thường dung cơ chế quay vòng để phân chia đường truyền theo các khoảng thời gian định trước. Cấp phát "động" là cấp phát theo yêu cầu để hạn chế thời gian "chết" vô ích của đường truyền.

- **Giao thức mạng**

Việc trao đổi thông tin cho dù là đơn giản nhất, cũng đều phải tuân theo những quy tắc nhất định. Hai người nói chuyện với nhau muốn cho cuộc nói chuyện có kết quả thì ít nhất cả hai cũng phải ngầm định tuân theo quy 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:

+ Khuôn dạng của dữ liệu: cú pháp và ngữ nghĩa

+ Thủ tục gửi và nhận dữ liệu

+ Kiểm soát chất lượng truyền

+ Xử lý các lỗi, sự cố

Tập hợp tất cả các quy tắc, quy ước trên gọi là giao thức mạng. Yêu cầu về xử lý và trao đổi thông tin của người sử dụng ngày càng cao thì giao thức mạng càng phức tạp. Các mạng có thể có giao thức khác nhau tuỳ thuộc vào sự lựa chọn của nhà thiết kế.

1.2.3. Phân loại mạng máy tính

Có nhiều cách để phân loại mạng máy tính tuỳ thuộc vào yếu tố chính được chọn làm chỉ tiêu để phân loại: khoảng cách địa lý, kỹ thuật chuyển mạch, kiến trúc của mạng.

1.2.3.1. Theo khoảng cách địa lý

Nếu lấy khoảng cách địa lý làm yếu tố chính để phân loại thì mạng máy tính được phân thành 4 loại: mạng cục bộ, mạng đô thị, mạng diện rộng, mạng toàn cầu.

- Mạng cục bộ (Local Area Networks - LAN): cài đặt trong phạm vi tương đối hẹp (ví dụ như trong một tòa nhà, một cơ quan, một trường học,..), khoảng cách lớn nhất giữa các máy tính nối mạng là vài chục km trở lại.

- Mạng đô thị (Metropolitan Area Networks - MAN): cài đặt trong phạm vi một đô thị, một trung tâm kinh tế xã hội, có bán kính nhỏ hơn 100 km.

- Mạng diện rộng (Wide Area Networks - WAN): phạm vi của mạng có thể vượt qua biên giới quốc gia và thậm chí cả lục địa.

- Mạng toàn cầu (Global Area Networks - GAN): phạm vi rộng khắp toàn cầu. Mạng Internet là một ví dụ cho loại này.

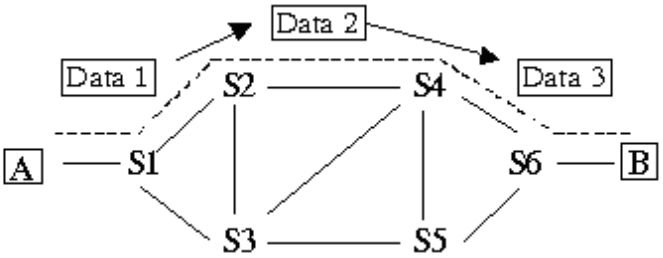
Chúng ta cũng cần lưu ý rằng: khoảng cách địa lý được dùng làm "mốc" chỉ mang tính tương đối. Cùng với sự phát triển của các công nghệ truyền dẫn và quản trị mạng thì những ranh giới đó ngày càng mờ nhạt đi.

1.2.3.2. Dựa theo kỹ thuật chuyển mạch

Nếu lấy "kỹ thuật chuyển mạch" làm yếu tố chính để phân ploại thì ta có 3 loại: mạng chuyển mạch kênh, mạng chuyển mạch thông báo và mạng chuyển mạch gói.

a. Mạng chuyển mạch kênh

Khi có hai thực thể cần trao đổi thông tin với nhau thì giữa chúng sẽ thiết lập một "kênh" cố định và được duy trì cho đến khi một trong hai bên ngắt liên lạc. Các dữ liệu chỉ được truyền theo con đường cố định đó.



Hình 1.7. Mạng chuyển mạch kênh

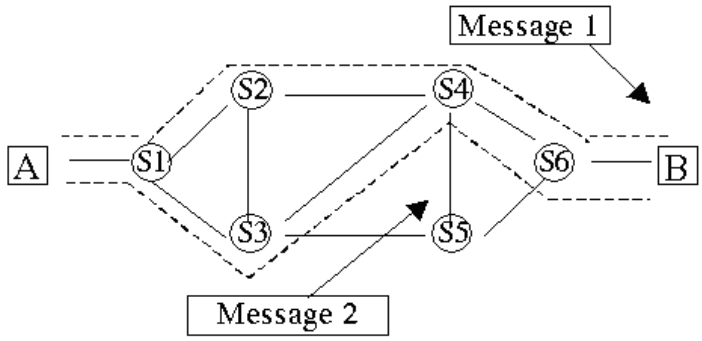
Nhược điểm:

+ Tốn thời gian để thiết lập kênh cố định giữa hai thực thể
+ Hiệu suất sử dụng đường truyền thấp vì sẽ có lúc kênh bị bỏ không do cả hai bên đều hết thông tin cần truyền trong khi các thực thể khác không được phép sử dụng kênh truyền này.

b. Mạng chuyển mạch thông báo

Thông báo (message) là một đơn vị thông tin của người sử dụng có khuôn dạng được qui định trước. Mỗi thông báo đều có chứa vùng thông tin điều khiển trong đó chỉ định rõ đích đến 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.

Mỗi nút cần phải lưu trữ tạm thời để "đọc" thông tin điều khiển trên thông báo để sau đó chuyển tiếp thông báo đi. Tuỳ thuộc vào điều kiện của mạng, các thông báo khác nhau có thể truyền theo đường truyền khác nhau.



Hình 1.8. Mạng chuyển mạch thông báo

Ưu điểm so với mạng chuyển mạch kênh:

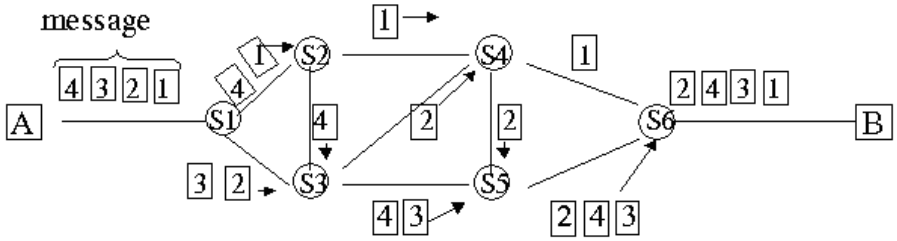
- Hiệu suất sử dụng đường truyền cao vì không bị chiếm dụng độc quyền mà được phân chia giữa nhiều thực thể.
- Mỗi nút mạng có thể lưu trữ thông báo cho tới khi kênh truyền rỗi mới gửi thông báo đi, vì vậy giảm được tình trạng tắc nghẽn mạch.
- Có thể điều khiển việc truyền tin bằng cách sắp xếp độ ưu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

Nhược điểm:

- Không hạn chế kích thước của các thông báo, dẫn đến phí tổn lưu trư tạm thời cao và ảnh hưởng tới thời gian đáp (respone time) và chất lượng truyền tin.
- Thích hợp cho các dịch vụ thư tín điện tử hơn là các áp dụng có tính thời gian thực vì tồn tại độ trễ do lưu trữ và xử lý thông tin điều khiển tại mỗi nút.

c. Mạng chuyển mạch gói

Mỗi thông báo được chia làm nhiều phần nhỏ hơn được gọi là các gói tin có khuôn dạng quy định trước. Mỗi gói tin cũng chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) và đích (người nhận) của gói tin. Các gói tin của một thông báo có thể đi qua mạng tới đích bằng nhiều con đường khác nhau. Ở bên nhận, thứ tự nhận được có thể không đúng thứ tự được gửi đi.



Hình 1.9. Mạng chuyển mạch gói

So sánh mạng chuyển mạch thông báo và mạng chuyển mạch gói:

♦ Giống nhau: phương pháp giống nhau

♦ Khác nhau: Các gói tin được giới hạn kích thước tối đa sao cho các nút mạng có thể xử lý toàn bộ gói tin trong bộ nhớ mà không cần phải lưu trữ tạm thời trên đĩa. Vì thế mạng chuyển mạch gói truyền các gói tin qua mạng nhanh chóng và hiệu quả hơn so với mạng chuyển mạch thông báo. Nhưng vấn đề khó khăn của mạng loại này là việc tập hợp các gói tin để tạo lại thông báo ban đầu của người sử dụng, đặc biệt trong trường hợp các gói được truyền theo nhiều đường khác nhau. Cần phải cài đặt cơ chế "đánh dấu" gói tin và phục hồi gói tin bị thất lạc hoặc truyền bị lỗi cho các nút mạng.

Do có ưu điểm mềm dẻo và hiệu suất cao hơn nên hiện nay mạng chuyển mạch gói được sử dụng 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- Intergrated Services Digital Networks, viết tắt là ISDN) đang là một xu hướng phát triển của mạng ngày nay.

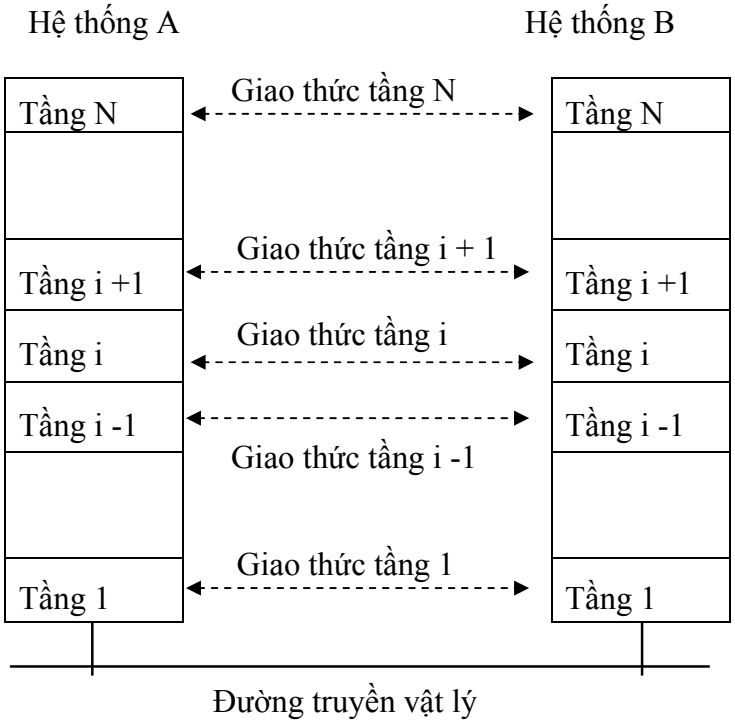
1.2.3.3. Phân loại theo kiến trúc mạng

Người ta còn phân loại mạng theo kiến trúc mạng (topo và giao thức sử dụng). Các mạng thường hay được nhắc đến như: mạng SNA của IBM, mạng ISO, mạng TCP/IP.

1.3. KIẾN TRÚC PHÂN TẦNG VÀ MÔ HÌNH OSI

1.3.1. Kiến trúc phân tầng

Để giảm độ phức tạp của việc thiết kế và cài đặt mạng, hầu hết các máy tính đều được phân tích thiết kế theo quan điểm phân tầng. Mỗi hệ thống thành phần của mạng được xem như một cấu trúc đa tầng, trong đó mỗi tầng được xây dựng trên tầng trước nó. Số lượng các tầng cũng như tên và chức năng của mỗi tầng tuỳ thuộc vào nhà thiết kế. Trong hầu hết các mạng, mục đích của mỗi tầng là để cung cấp một số dịch vụ nhất định cho tầng cao hơn. Mỗi tầng khi sử dụng không cần quan tâm đến các thao tác chi tiết mà các dịch vụ đó phải thực hiện.



Hình 1.10. Minh hoạ kiến trúc phân tầng tổng quát

Nguyên tắc của kiến trúc mạng phân tầng:

- Mỗi hệ thống trong một mạng đều có cấu trúc tầng như nhau (số lượng tầng, chức năng của mỗi tầng).

- Dữ liệu không được truyền trực tiếp từ tầng i của hệ thống này sang tầng thứ *i* của hệ thống kia (ngoại trừ đối với tầng thấp nhất). Bên gửi dữ liệu cùng với các thông tin điều khiển chuyển đến tầng ngay dưới nó và cứ thế cho đến tầng thấp nhất. Bên dưới tầng này là đường truyền vật lý, ở đấy sự truyền tin mới thực sự diễn ra. Đối với bên nhận thì các thông tin được chuyển từ tầng dưới lên trên cho tới tầng *i* của hệ thống nhận.
- Giữa hai hệ thống kết nối chỉ ở tầng thấp nhất mới có liên kết vật lý còn ở tầng cao hơn chỉ là liên kết logic hay liên kết ảo được đưa vào để hình thức hóa các hoạt động của mạng, thuận tiện cho việc thiết kế và cài đặt các phần mềm truyền thông.

Các vấn đề cần phải giải quyết khi thiết kế các tầng

- Cơ chế nối, tách: mỗi một tầng cần có một cơ chế để thiết lập kết nối, và có một cơ chế để kết thúc kết nối khi mà sự kết nối là không cần thiết nữa.
- Các quy tắc truyền dữ liệu: Trong các hệ thống khác nhau dữ liệu có thể truyền theo một số cách khác nhau:
  + Truyền một hướng (simplex)
  + Truyền hai hướng đồng thời (full-duplex)
  + Truyền theo cả hai hướng luân phiên (half-duplex)
- Kiểm soát lỗi: Đường truyền vật lý nói chung là không hoàn hảo, cần phải thoả thuận dùng một loại mã để phát hiện, kiểm tra lỗi và sửa lỗi. Phía nhận phải có khả năng thông báo cho bên gửi biết các gói tin nào đã thu đúng, gói tin nào phát lại.
- Độ dài bản tin: Không phải mọi quá trình đều chấp nhận độ dài gói tin là tuỳ ý, cần phải có cơ chế để chia bản tin thành các gói tin đủ nhỏ.
- Thứ tự các gói tin: Các kênh truyền có thể giữ không đúng thứ tự các gói tin, do đó cần có cơ chế để bên thu ghép đúng thứ tự ban đầu.
- Tốc độ phát và thu dữ liệu: Bên phát có tốc độ cao có thể làm "lụt" bên thu có tốc độ thấp. Cần phải có cơ chế để bên thu báo cho bên phát biết tình trạng đó để điều khiển lưu lượng hợp lý.

1.3.2. Một số khái niệm cơ bản

Tầng (layer)

- Mọi quá trình trao đổi thông tin giữa hai đối tượng đều thực hiện qua nhiều bước, các bước này độc lập tương đối với nhau. Thông tin được trao đổi giữa hai đối tượng A, B qua 3 bước:
- Phát tin: Thông tin chuyển từ tầng cao → tầng thấp
- Nhận tin: Thông tin chuyển từ tầng thấp → tầng cao
- Quá trình trao đổi thông tin trực tiếp qua đường truyền vật lý (thực hiện ở tầng cuối cùng)

Giao diện, dịch vụ, đơn vị dữ liệu

- Mối quan hệ giữa hai tầng kề nhau gọi là giao diện
- Mối quan hệ giữa hai tầng đồng mức của hai hệ thống khác nhau gọi là giao thức
- Thực thể (entity): là thành phần tích cực trong mỗi tầng, nó có thể là một tiến trình trong hệ đa xử lý hay là một trình con các thực thể trong cùng 1 tầng ở các hệ thống khác nhau (gọi là thực thể ngang hàng hay thực thể đồng mức). Mỗi thực thể có thể truyền thông lên tầng trên hoặc tầng dưới nó thông qua một giao diện (interface). Giao diện gồm một hoặc nhiều điểm truy nhập dịch vụ (Service Access Point - SAP). Tại các điểm truy nhập dịch vụ tầng trên chỉ có thể sử dụng dịch vụ do tầng dưới cung cấp. Thực thể được chia làm hai loại: thực thể cung cấp dịch vụ và sử dụng dịch vụ:
  + Thực thể cung cấp dịch vụ (service provide): là các thực thể ở tầng N cung cấp dịch vụ cho tầng N +1.
  + Thực thể sử dụng dịch vụ (service user): đó là các thực thể ở tầng N sử dụng dịch vụ do tầng N - 1 cung cấp.
- Đơn vị dữ liệu sử dụng giao thức (Protocol Data Unit - PDU)
- Đơn vị dữ liệu dịch vụ (Service Data Unit - SDU)
- Thông tin điều khiển (Protocol Control Information - PCI)

Một đơn vị dữ liệu mà 1 thực thể ở tầng N của hệ thống A gửi sang thực thể ở tầng N ở một hệ thống B không bằng đường truyền trực tiếp mà phải truyền xuống dưới để truyền bằng tầng thấp nhất thông qua đường truyền vật lý.

+ Dữ liệu ở tầng N-1 nhận được do tầng N truyền xuống gọi là SDU.
+ Phần thông tin điều khiển của mỗi tầng gọi là PCI.
+ Ở tầng N-1 phần thông tin điều khiển PCI thêm vào đầu của SDU tạo thành PDU. Nếu SDU quá dài thì cắt nhỏ thành nhiều đoạn, mỗi đoạn bổ sung phần PCI, tạo thành nhiều PDU.

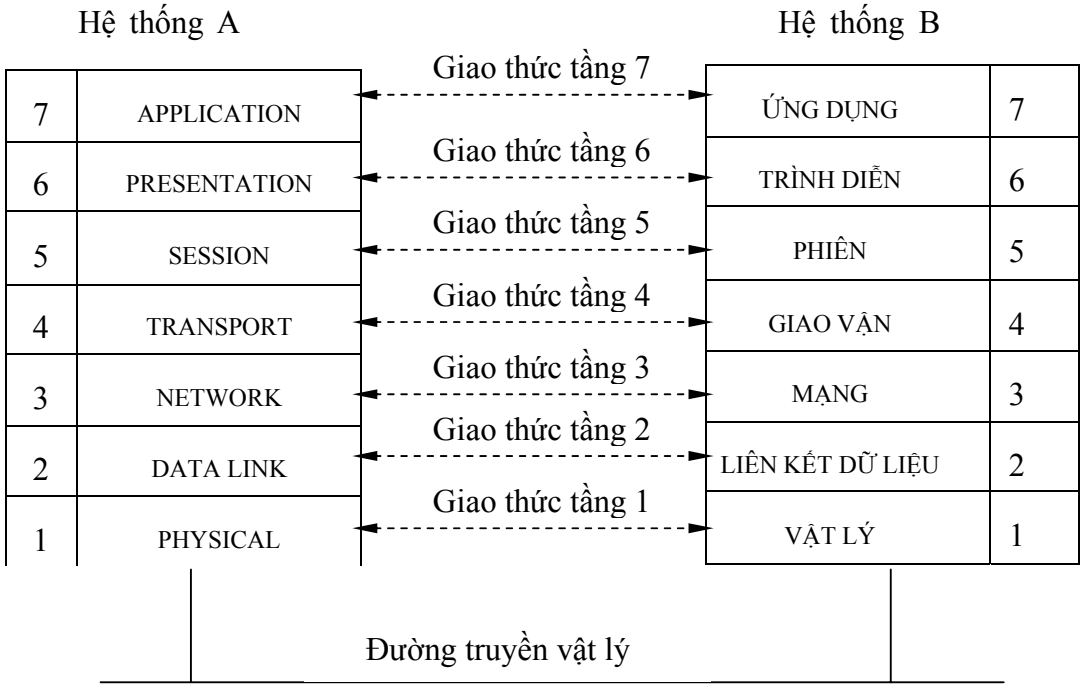
Bên hệ thống nhận trình tự diễn ra theo chiều ngược lại. Qua mỗi tầng PCI tương ứng sẽ được phân tích và cắt bỏ khỏi PDU trước khi gửi lên tầng trên.

1.3.3. Mô hình OSI

1.3.3.1. Giới thiệu

Khi thiết kế các nhà thiết kế tự do lựa chọn kiến trúc mạng riêng của mình. Từ đó dẫn đến tình trạng không tương thích giữa các mạng: phương pháp truy nhập đường truyền khác nhau, sử dụng họ giao thức khác nhau,... Sự không tương thích đó làm cho người sử dụng các mạng khác nhau không thể trao đổi thông tin với nhau được. Sự thúc bách của khách hàng khiến cho các nhà sản xuất và những nhà nghiên cứu, thông qua tổ chức chuẩn hoá quốc tế và quốc gia để tìm ra một giải pháp chung dẫn đến sự hội tụ của các sản phẩm mạng. Trên cơ sở đó những nhà thiết kế và các nghiên cứu lấy đó làm khung chuẩn cho sản phẩm của mình.

Vì lý do đó, năm 1977, Tổ chức tiêu chuẩn hoá quốc tế (International Organization for Standardization - ISO) đã lập ra một tiểu ban nhằm đưa ra một khung chuẩn như thế. Kết quả là vào năm 1984 ISO đã xây dựng mô hình 7 tầng gọi là mô hình tham chiếu cho việc nối kết các hệ thống mở (Reference Model for Open Systems Interconnection - OSI Reference Model) gọi tắt là mô hình OSI. Mô hình này được dùng làm cơ sở để nối kết các hệ thống mở phục vụ cho các ứng dụng phân tán → Mọi hệ thống tuân theo mô hình tham chiếu OSI đều có thể truyền thông tin với nhau.



Hình 1.11. Mô hình OSI 7 tầng

1.3.3.2. Chức năng các tầng trong mô hình OSI

a. Tầng vật lý (Physical)

- Tầng vật lý liên quan đến truyền dòng các bit giữa các máy với nhau bằng đường truyền vật lý. Tầng này liên kết các giao diện hàm cơ, quang và điện với cáp. Ngoài ra nó cũng chuyển tải những tín hiệu truyền dữ liệu do các tầng ở trên tạo ra.
- Việc thiết kế phải bảo đảm nếu bên phát gửi bít 1 thì bên thu cũng phải nhận bít 1 chứ không phải bít 0
- Tầng này phải quy định rõ mức điện áp biểu diễn dữ liệu 1 và 0 là bao nhiêu von trong vòng bao nhiêu giây
- Chiều truyền tin là 1 hay 2 chiều, cách thức kết nối và huỷ bỏ kết nối
- Định nghĩa cách kết nối cáp với card mạng: bộ nối có bao nhiêu chân, chức năng của mỗi chân

Tóm lại: Thiết kế tầng vật lý phải giải quyết các vấn đề ghép nối cơ, điện, tạo ra các hàm, thủ tục để truy nhập đường truyền, đường truyền các bít.

b. Tầng liên kết dữ liệu (data link)

- Cung cấp phương tiện để truyền thông tin qua liên kết vật lý đảm bảo tin cậy: gửi các khối dữ liệu với cơ chế đồng bộ hoá, kiểm soát lỗi và kiểm soát luồng dữ liệu cần thiết
- Các bước tầng liên kết dữ liệu thực hiện:
  + Chia nhỏ thành các khối dữ liệu frame (vài trăm bytes), ghi thêm vào đầu và cuối của các frame những nhóm bít đặc biệt để làm ranh giới giữa các frame
  + Trên các đường truyền vật lý luôn có lỗi nên tầng này phải giải quyết vấn đề sửa lỗi (do bản tin bị hỏng, mất và truyền lại)
  + Giữ cho sự đồng bộ tốc độ giữa bên phát và bên thu

Tóm lại: tầng liên kết dữ liệu chịu trách nhiệm chuyển khung dữ liệu không lỗi từ máy tính này sang máy tính khác thông qua tầng vật lý. Tầng này cho phép tầng mạng truyền dữ liệu gần như không phạm lỗi qua liên kết mạng

c. Tầng mạng (Network)

- Lập địa chỉ các thông điệp, diễn dịch địa chỉ và tên logic thành địa chỉ vật lý
- Kiểm soát và điều khiển đường truyền: Định rõ các bó tin được truyền đi theo con đường nào từ nguồn tới đích. Các con đường đó có thể là cố định đối với những mạng ít thay đổi, cũng có thể là động nghĩa là các con đường chỉ được xác định trước khi bắt đầu cuộc nói chuyện. Các con đường đó có thể thay đổi tuỳ theo trạng thái tải tức thời.
- Quản lý lưu lượng trên mạng: chuyển đổi gói, định tuyến, kiểm soát sự tắc nghẽn dữ liệu (nếu có nhiều gói tin cùng được gửi đi trên đường truyền thì có thể xảy ra tắc nghẽn)
- Kiểm soát luồng dữ liệu và cắt hợp dữ liệu (nếu cần)

d. Tầng giao vận (Transport)

- Thực hiện việc truyền dữ liệu giữa hai đầu nút (end - to - end).
- Thực hiện kiểm soát lỗi, kiểm soát luồng dữ liệu từ máy → máy. Đảm bảo gói tin truyền không phạm lỗi, theo đúng trình từ, không bị mất mát hay sao chép.
- Thực hiện việc ghép kênh, phân kênh cắt hợp dữ liệu (nếu cần). Đóng gói thông điệp, chia thông điệp dài thành nhiều gói tin và gộp các gói nhỏ thành một bộ.
- Tầng này tạo ra một kết nối cho mỗi yêu cầu của tầng trên nó. Khi có nhiều yêu cầu từ tầng trên với thông lượng cao thì nó có thể tạo ra nhiều kết nối và cùng một lúc có thể gửi đi nhiều bó tin trên đường truyền.

e. Tầng phiên (Session)

- Cung cấp phương tiện truyền thông giữa các ứng dụng: cho phép người sử dụng trên các máy khác nhau có thể thiết lập, duy trì, huỷ bỏ và đồng bộ hoá các phiên truyền thông giữa họ với nhau.
- Nhiệm vụ chính:

+ Quản lý thẻ bài đối với những nghi thức: hai bên kết nối để truyền thông tin không đồng thời thực hiện một số thao tác. Để giải quyết vấn đề này tầng phiên cung cấp 1 thẻ bài, thẻ bài có thể được trao đổi và chỉ bên nào giữ thẻ bài mới có thể thực hiện một số thao tác quan trọng

+ Vấn đề đồng bộ: khi cần truyền đi những tập tin dài tầng này chèn thêm các điểm kiểm tra (check point) vào luồng dữ liệu. Nếu phát hiện thấy lỗi thì chỉ có dữ liệu sau điểm kiểm tra cuối cùng mới phải truyền lại

f. Tầng trình diễn (Presentation)

- Quyết định dạng thức trao đổi dữ liệu giữa các máy tính mạng. Người ta có thể gọi đây là bộ dịch mạng. Ở bên gửi, tầng này chuyển đổi cú pháp dữ liệu từ dạng thức do tầng ứng dụng gửi xuống sang dạng thức trung gian mà ứng dụng nào cũng có thể nhận biết. Ở bên nhận, tầng này chuyển các dạng thức trung gian thành dạng thức thích hợp cho tầng ứng dụng của máy nhận.
- Tầng trình diễn chịu trách nhiệm chuyển đổi giao thức, biên dịch dữ liệu, mã hoá dữ liệu, thay đổi hay chuyển đổi ký tự và mở rộng lệnh đồ hoạ.
- Nén dữ liệu nhằm làm giảm bớt số bít cần truyền
- Ở tầng này có bộ đổi hướng hoạt đông để đổi hướng các hoạt động nhập/xuất để gửi đến các tài nguyên trên mấy phục vụ

g. Tầng ứng dụng (Application)

- 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.
- Tầng này đóng vai trò như cửa sổ dành cho hoạt động xử lý các trình ứng dụng nhằm truy nhập các dịch vụ mạng. Nó biểu diễn những dịch vụ hỗ trợ trực tiếp các ứng dụng người dùng, chẳng hạn như phần mềm chuyển tin, truy nhập cơ sở dữ liệu và email.
- Xử lý truy nhập mạng chung, kiểm soát lỗi và phục hồi lỗi.

1.3.3.3. Các dịch vụ và hàm

Dịch vụ là một dãy, một tập các thao tác sơ cấp hay là các hàm nguyên thủy mà một tầng cung cấp cho tầng trên nó. Dịch vụ liên quan đến 2 tầng kề nhau

a. Dịch vụ định hướng liên kết và dịch vụ không liên kết

Ở mỗi tầng trong mô hình OSI có hai loại dịch vụ: dịch vụ định hướng liên kết (connection - oriented service) và dịch vụ không định hướng liên kết (connectionless service)

- Dịch vụ định hướng liên kết: là dịch vụ theo mô hình điện thoại, trước khi truyền dữ liệu cần thiết lập một liên kết logic giữa các thực thể đồng mức
- Dịch vụ không liên kết: không cần phải thiết lập liên kết logic và một đơn vị dữ liệu được truyền là độc lập với các đơn vị dữ liệu trước hoặc sau nó. Loại dịch vụ này theo mô hình bưu điện: mỗi bản tin hay mỗi bức thư cần có một địa chỉ cụ thể bên nhận

Trong phương pháp liên kết quá trình truyền thông gồm có 3 giai đoạn:

- Thiết lập liên kết (logic): hai thực thể đồng mức ở hai hệ thống sẽ thương lượng với nhau về tập các tham số sẽ sử dụng trong giai đoạn truyền sau (thể hiện bằng hàm CONNECT).
- Truyền dữ liệu: dữ liệu được truyền với các cơ chế kiểm soát và quản lý kèm theo (kiểm soát lỗi, kiểm soát luồng dữ liệu, cắt/hợp dữ liệu,...) để tăng độ tin cậy và hiệu quả của việc truyền dữ liệu (hàm DATA).
- Huỷ bỏ liên kết (logic): giải phóng các tài nguyên hệ thống đã được cấp phát cho liên kết để dùng cho các liên kết khác (hàm DISCONNECT).

Trong mỗi loại dịch vụ được đặc trưng bằng chất lượng dịch vụ. Có dịch vụ đòi hỏi bên nhận tin gửi thông báo xác nhận khi đó độ tin cậy được bảo đảm.

Có những ứng dụng không chấp nhận sự chậm trễ do phải xác nhận sự truyền tin (VD hệ thống truyền tin). Nhưng có nhiều ứng dụng như thư tín điện tử người gửi chỉ cần có một dịch vụ với độ tin cậy cao, chấp nhận sự chậm trễ.

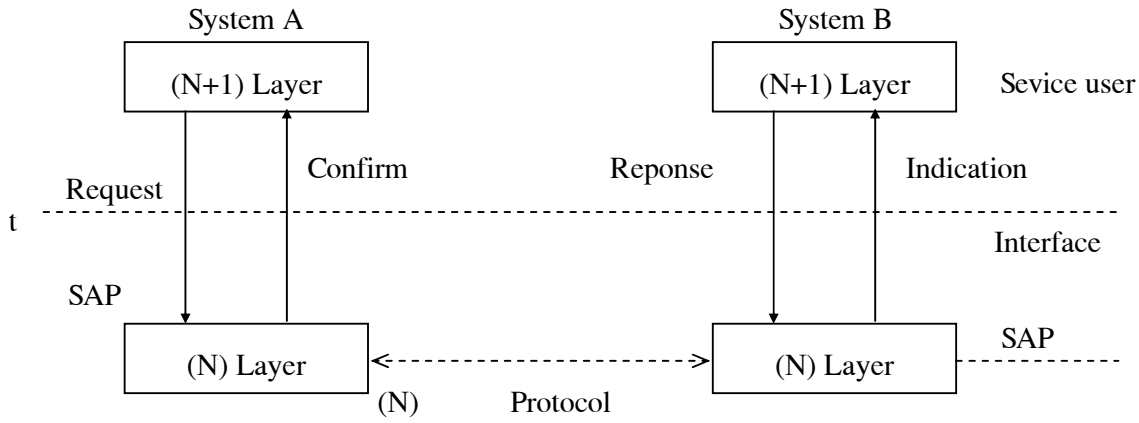
b. Các hàm nguyên thuỷ của dịch vụ

Một dịch vụ gồm 1 số thao tác sơ cấp hay các hàm nguyên thuỷ. Một thực thể cung cấp dịch vụ cho một thực thể ở tầng trên nó thông qua việc gọi các hàm nguyên thuỷ. Các hàm nguyên thuỷ chỉ rõ chức năng cần phải thực hiện và dùng để chuyển dữ liệu vào thông tin điều khiển. Có 4 hàm nguyên thuỷ được dùng để xác định tương tác giữa các tầng kề nhau (hình 1.12).

- *Request* (yêu cầu): người sử dụng dịch vụ dùng để gọi chức năng hoặc yêu cầu thực thể khác thực hiện một công việc nào đó.
- *Indication* (chỉ báo): người cung cấp dịch vụ dung để gọi một chức năng nào đó, chỉ báo một chức năng đã được gọi ở một điểm truy nhập dịch vụ.
- *Response* (trả lời): người sử dụng dịch vụ dùng để hoàn tất một chức năng đã được gọi từ trước bởi một hàm nguyên thuỷ Indication ở điểm truy nhập dịch vụ đó.
- *Confirm* (xác nhận): người cung cấp dịch vụ dùng để hoàn tất một chức năng đã được gọi từ trước bởi một hàm nguyên thuỷ Response tại điểm truy nhập dịch vụ.

Quy trình thực hiện một giao tác giữa hai hệ thống A và B :

- Tầng N+1 của A gửi xuống tầng N kề dưới nó một hàm Request
- Tầng N của A cấu tạo một đơn vị dữ liệu để gửi yêu cầu đó sang tầng N của B theo giao thức tầng N đã xác định
- Nhận được yêu cầu, tầng N của B chỉ báo lên tầng N+1 kề trên nó bằng hàm Indication
- Tầng N của B trả lời bằng hàm Response gửi xuống tầng N kề dưới nó
- Tầng N của B cấu tạo một đơn vị dữ liệu để gửi trả lời đó về tầng N của A theo giao thức tầng N đã xác định
- Nhận được trả lời, tầng N của A xác nhận với tầng N+1 kề trên nó bằng hàm Confirm, kết thúc một giao tác giữa hai hệ thống.



Hình 1.12. Sơ đồ hoạt động của các hàm nguyên thuỷ

Các thao tác sơ cấp nói chung là có tham số (VD Connect.Request). Các tham số gồm:

- Địa chỉ máy gọi
- Địa chỉ máy nhận
- Loại dịch vụ
- Kích thước cực đại của bản tin

Nếu thực thể bị gọi không chấp nhận kích thước cực đại mà bản tin đưa ra nó có thể yêu cầu kích thước mới trong thao tác của hàm Response. Các chi tiết của quá trình thoả thuận là một phần của nghi thức. Các dịch vụ có thể xác nhận hoặc không xác nhân.

- Các dịch vụ xác nhận có thể có các hàm nguyên thuỷ: Request, Indication, Response, Confirm
- Dịch vụ không xác nhận có hai hàm nguyên thuỷ: Resquest, Indication

Trong thực tế loại dịch vụ connect luôn luôn là có xác nhận, còn các loại dịch vụ DATA là không xác nhận hoặc có xác nhận.

STT	Hàm nguyên thuỷ	Ý nghĩa
1.	CONNECT.Request	Yêu cầu thiết lập liên kết
2.	CONNECT.Indication	Báo cho thực thể bị gọi
3.	CONNECT.Response	Đồng ý hay không đồng ý
4.	CONNECT.Confirm	Xác nhận với bên gọi việc kết nối có được chấp nhận hay không
5.	DATA.Request	Bên gọi yêu cầu truyền dữ liệu
6.	DATA.Indication	Báo cho bên nhận biết là dữ liệu đã đến
7.	DISCONNECT.Request	Yêu cầu huỷ bỏ liên kết
8.	DISCONNECT.Indication	Báo cho bên nhận

Ví dụ:

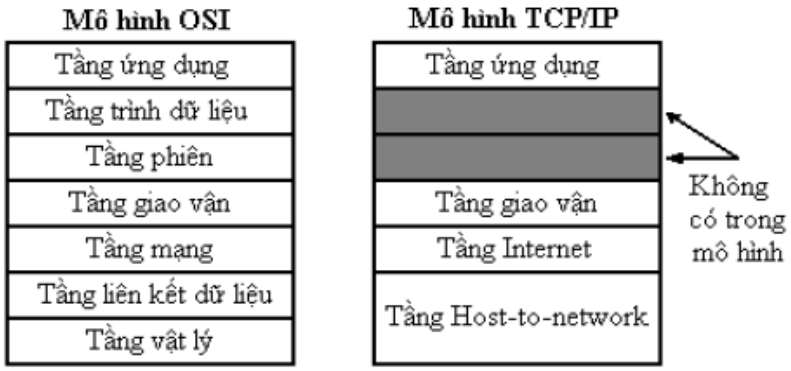
1.	CONNECT.Request	Bạn quay số điện thoại của cô Lan
2.	CONNECT.Indication	Chuông reo
3.	CONNECT.Response	Cô Lan nhấc máy
4.	CONNECT.Confirm	Chuông ngừng reo
5.	DATA.Request	Bạn nói chuyện với cô Lan
6.	DATA.Indication	Cô Lan nghe thấy bạn nói
7.	DATA.Response	Cô trả lời bạn
8.	DATA.Confirm	Bạn nghe thấy câu trả lời
9.	DISCONNECT.Request	Bạn cúp máy
10.	DISCONNECT.Indication	Cô Lan nghe thấy bạn cúp máy.

1.3.4. Các mô hình chuẩn hoá khác

1.3.4.1. Mô hình TCP/IP

Mặc dù mô hình tham chiếu OSI được chấp nhận rộng rãi khắp nơi, nhưng chuẩn mở về kỹ thuật mang tính lịch sử của Internet lại là TCP/IP (Transmission Control Protocol/Internet Protocol). Mô hình tham chiếu TCP/IP và chồng giao thức TCP/IP tạo khả năng truyền dữ liệu giữa hai máy tính từ bất kỳ nơi nào trên thế giới, với tốc độ gần bằng tốc độ ánh sáng. Mô hình TCP/IP có tầm quan trọng trong lịch sử, gần giống như các chuẩn đã cho phép điện thoại, năng lượng điện, đường sắt, truyền hình và công nghệ băng hình phát triển cực thịnh.

Hình 1.13 trình bày hai mô hình tham chiếu TCP/IP và OSI để tiện so sánh.



Hình 1.13. Tương ứng các tầng các kiến trúc TCP/IP và OSI

Các tầng của mô hình tham chiếu TCP/IP

Bộ quốc phòng Mỹ gọi tắt là DoD (Department of Defense) đã tạo ra mô hình tham chiếu TCP/IP vì muốn một mạng có thể tồn tại trong bất cứ điều kiện nào, ngay cả khi có chiến tranh hạt nhân. DoD muốn các gói dữ liệu xuyên suốt mạng vào mọi lúc, dưới bất cứ điều kiện nào, từ bất cứ một điểm đến một điểm khác. Đây là một bài toán thiết kế cực kỳ khó khăn mà từ đó làm nảy sinh ra mô hình TCP/IP, vì vậy đã trở thành chuẩn Internet để phát triển.

Tầng ứng dụng (Application)

Các nhà thiết kế TCP/IP cảm thấy rằng các giao thức mức cao nên bao gồm các tầng trình bày và tầng phiên. Để đơn giản, họ tạo ra một tầng ứng dụng kiểm soát các giao thức mức cao, các vấn đề của tầng trình bày, mã hoá và điều khiển hội thoại. TCP/IP tập hợp tất cả các vấn đề liên quan đến ứng dụng vào trong một tầng, và đảm bảo dữ liệu được đóng gói một cách thích hợp cho tầng kế tiếp.

Tầng vận chuyển (Transportation)

Tầng vận chuyển đề cập đến các vấn đề chất lượng dịch vụ như độ tin cậy, điều khiển luồng và sửa lỗi. Một trong các giao thức của nó la TCP, TCP cung cấp các phương thức linh hoạt và hiệu quả để thực hiện các hoạt động truyền dữ liệu tin cậy, hiệu xuất cao và ít lỗi. TCP là giao thức có tạo cầu nối (connection-oriented). Nó tiến hành hội thoại giữa nguồn và đích trong khi bọc thông tin tầng ứng dụng thành các đơn vị gọi là segment.Tạo cầu nối không có nghĩa là tồn tại một mạch thực sự giữa hai máy tính, thay vì vậy nó có nghĩa là các segment của tầng 4 di chuyển tới và lui giữa hai host để công nhận kết nối tồn tại một cách luận lý trong một khoảng thời gian nào đó. Điều này coi như chuyển mạch gói (packet switching).

Tầng Internet

Mục tiêu của tầng Internet là truyền các gói tin bắt nguồn từ bất kỳ mạng nào trên liên mạng và đến được đích trong điều kiện độc lập với đường dẫn và các mạng mà chúng đã trải qua. Giao thức đặc trưng khống chế tầng này được gọi là IP. Công việc xác định đường dẫn tốt nhất và hoạt động chuyển mạch gói diễn ra tại tầng này.

Tầng truy xuất mạng (Host to network)

Tên của tầng này có nghĩa khá rộng và có phần hơi rối rắm. Nó cũng được gọi là tầng host-to-network. Nó là tầng liên quan đến tất cả các vấn đề mà một gói IP yêu cầu để tạo một liên kết vật lý thực sự, và sau đó tạo một liên kết vật lý khác. Nó bao gồm các chi tiết kỹ thuật LAN và WAN, và tất cả các chi tiết trong tầng liên kết dữ liệu cũng như tầng vật lý của mô hình OSI.

Mô hình TCP/IP hướng đến tối đa độ linh hoạt tại tầng ứng dụng cho người phát triển phần mềm. Tầng vận chuyển liên quan đến hai giao thức TCP và UDP (User Datagram Protocol). Tầng cuối cùng, tầng truy xuất mạng liên kết đến các kỹ thuật LAN hay WAN đang được dùng.

Trong mô hình TCP/IP không cần quan tâm đến ứng dụng nào yêu cầu các dịch vụ mạng, và không cần quan tâm đến giao thức vận chuyển nào đang được dùng, chỉ có một giao thức mạng IP. Đây là một quyết định thiết kế có cân nhắc kỹ. IP phục vụ như một giao thức đa năng cho phép bất kỳ máy tính nào, ở bất cứ đâu, truyền dữ liệu vào bất cứ thời điểm nào.

1.3.4.2. Mô hình SNA

Tháng 9/1973, Hãng IBM giới thiệu một kiến trúc mạng máy tính SNA (System Network Architecture). Đến năm 1977 đã có 300 trạm SNA được cài đặt. Cuối năm 1978, số lượng đã tăng lên đến 1250, rồi cứ theo đà đó cho đến nay đã có 20.000 trạm SNA đang được hoạt động. Qua con số này chúng ta có thể hình dung được mức độ quan trọng và tầm ảnh hưởng của SNA trên toàn thế giới.

Cần lưu ý rằng SNA không là một chuẩn quốc tế chính thức như OSI nhưng do vai trò to lớn của hãng IBM trên thị trường CNTT nên SNA trở thành một loại chuẩn thực tế và khá phổ biến. SNA là một đặc tả gồm rất nhiều tài liệu mô tả kiến trúc của mạng xử lý dữ liệu phân tán. Nó 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.

SNA được tổ chức xung quanh khái niệm miền (domain). Một SNA 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 SNA. Nó điều khiển SNA 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 SNA.

SNA 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. Nó 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. Nó 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 SNA 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 SNA 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:

Hình 1.14. Tương ứng các tầng các kiến trúc SNA và OSI

– **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 SNA. Những dịch vụ đó như : DIA cung cấp các tài liệu phân bố giữa các hệ thống văn phòng, SNA 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). Nó 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 phiên và hợp tác giữa các phiên gởi và nhận. Nói 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 SNA và để kết nối với các mạng SNA 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.

1.4. HỆ ĐIỀU HÀNH MẠNG

1.4.1. Đặc điểm quy định chức năng của một hệ điều hành mạng.

Môi trường mạng có những đặc điểm riêng, khác với môi trường chỉ dùng máy tính cá nhân (PC), thể hiện ở các đặc trưng sau:

- Trước hết đó là môi trường nhiều người dùng. Đặc điểm này dẫn đến các nhu cầu liên lạc giữa những người sử dụng, nhu cầu bảo vệ dữ liệu và nói chung là bảo vệ tính riêng tư của người sử dụng.
- Mạng còn là môi trường đa nhiệm, có nhiều công việc thực hiện trên mạng. Đặc điểm này sẽ phát sinh các nhu nhu cầu chia sẻ tài nguyên, nhu cầu liên lạc giữa các tiến trình như trao đổi dữ liệu, đồng bộ hoá.

- Là môi trường phân tán, tài nguyên (thông tin, thiết bị) nằm ở các vị trí khác nhau, chỉ kết nối thông qua các đường truyền vật lý. Điều này phát sinh các nhu cầu chia sẻ tài nguyên trên toàn mạng nhưng sự phân tán cần được trong suốt đối để nó không gây khó khăn cho người sử dụng.
- Có nhiều quan niệm cũng như các giải pháp mạng khác nhau. Điều đó nảy sinh nhu cầu giao tiếp giữa các mạng khác nhau.
- Làm việc trên môi trường mạng chắc chắn sẽ phức tạp hơn môi trường máy đơn lẻ. Vì thế rất cần có các tiện ích giúp cho việc sử dụng và quản trị mạng dễ dàng và hiệu quả.

Tất cả các nhu cầu trên phải được tính tới trong hệ điều hành mạng.

1.4.2. Các tiếp cận thiết kế và cài đặt

Để thiết kế và cài đặt một hệ điều hành mạng có hai cách tiếp cận khác nhau:

(1) Tôn trọng tính độc lập của các hệ điều hành cục bộ đã có trên các máy tính của mạng. Khi đó hệ điều hành mạng được cài đặt như một tập các chương trình tiện ích chạy trên các máy khác nhau của mạng. Giải pháp này tuy không được "đẹp" nhưng dễ cài đặt và không vô hiệu hóa được các phần mềm đã có.

(2) Bỏ qua các hệ điều hành đã có trên các máy và cài đặt mới hoàn toàn một hệ điều hành thuần nhất trên toàn mạng, gọi là hệ điều hành phân tán. Giải pháp này đẹp hơn về phương diện hệ thống so với giải pháp trên, nhưng bù lại độ phức tạp trong công việc thì lớn hơn rất nhiều. Mặt khác, việc tôn trọng tính độc lập và chấp nhận sự tồn tại của các sản phẩm hệ thống đã có là một điểm hấp dẫn của các tiếp cận thứ nhất. Bởi vậy tùy theo điều kiện cụ thể mà ta áp dụng giải pháp nào cho phù hợp. Sau đây ta xem xét cụ thể hơn về từng giải pháp nói trên

Hệ điều hành theo giải pháp (1)

Tư tưởng chủ đạo của giải pháp này là cung cấp cho mỗi người tư tưởng chủ đạo của giải pháp này là cung cấp cho mỗi người sử dụng mọi tiến trình đồng nhất mà ta gọi là Agent làm nhiệm vụ cung cấp một giao diện đồng nhất và tất cả các hệ thống cục bộ đã có Agent quản lý một cơ sở dữ liệu chứa các thông tin về các hệ thống cục bộ và chương trình dữ liệu của người sử dụng trong trường hợp đơn giản nhất Agent chỉ hoạt động như một bộ xử lý lệnh, dịch các lệnh của người sử dụng thành ngôn ngữ lệnh của hệ thống cục bộ rồi gửi chúng để thực hiện trước khi mỗi chương trình thực hiện, Agent phải đảm bảo rằng tất cả các tệp cần thiết đề sử dụng. Việc cài đặt mạng như vậy sẽ chống lại hai công việc chính: thiết kế ngôn ngữ lệnh của mạng và cài đặt Agent.

Cách tiếp nhận này đơn giản và không gây ảnh hưởng đến hệ thống cục bộ đã có sẵn. Thậm chí các hệ thống cục bộ không cần thiết đến sự tồn tại của mạng. Nhưng giải pháp này chỉ có thể khả thi khi mà tất cả các tệp tin cần thiết đều biết trước để Agent có thể gửi chúng tới một hệ thống cục bộ trước khi chương trình bắt đầu hoạt động. Ngoài ra rất khó thực hiện các tương tác vào ra mà chương trình lại không biết tới sự tồn tại của mạng. Một giải pháp tổng quát hơn nhằm bỏ tiến trình đang chạy lại bằng cách tóm tắt tất cả các lời gọi hệ thống System Call của nó để chúng có thể thực hiện trong bối cảnh của hệ thống quản lý tệp của mạng (NetWork file System).

Hệ điều hành theo giải pháp (2)

Trong trường hợp này người ta gọi là hệ điều hành phân tán và có thể được thiết kế một trong hai mô hình: *Mô hình tiến trình* hoặc *mô hình đối tượng*.

Trong *mô hình tiến trình* mỗi tài nguyên (tệp, đĩa, thiết bị ngoại vi, …) được quản lý theo một tiến trình nào đó và hệ điều hành mạng điều khiển sự tương tác giữa các tiến trình đó. Các dịch vụ của hệ điều hành mạng tập trung truyền thông như quản lý tệp, lên lịch cho bộ xử lý, điều khiển terminal,... được quản lý bởi các Server đặc biệt có khả năng tiếp nhận các yêu cầu thực hiện dịch vụ tương ứng trong nhiều trường hợp các Server có thể chạy như tiến trình của người sử dụng thông thường.

Trong *mô hình đối tượng*, thế giới bao gồm các đối tượng khác nhau, mỗi đối tượng có một *kiểu* (type), một *biểu diễn,* và một tập các thao tác có thể thực hiện trên nó. Để thực hiện một thao tác trên một đối tượng, chẳng hạn đọc một tệp tin trên một tiến trình người sử dụng phải có "giấy phép" đối với đối tượng. Nhiệm vụ cơ bản của hệ điều hành đây là quản lý các giấy phép và cấp phát các "giấy phép" đó cho các tiến trình để thực hiện cho các thao tác cần thiết. Trong một hệ tập trung, bản thân hệ điều hành nằm giữ các "giấy phép" bên trong để ngăn ngừa những người sử dụng cố ý giả mạo chúng. Trong một hệ phân tán các "giấy phép" được luân chuyển theo một cách nào đó để mỗi tiến trình đều có cơ hội nhận được "giấy phép" và sao cho người sử dụng không thể tự tạo ra được chúng.

Việc thiết kế hệ điều hành phân tán theo môi hình đối tượng là một hướng đi rất triển vọng và tồn tại nhiều vấn đề cần giải quyết trọn vẹn hơn. Còn đối với tiến trình thì chúng ta có thể thấy rõ nhiệm vụ then chốt chính là xây dựng cơ chế liên lạc giữa các tiến trình (Interprocess Communication - IPC). Để làm điều đó người ta sử dụng một trong hai cách: dùng lời gọi hàm (Function/procedure Calls) hoặc chuyển thông báo (message passing).

Khi các lời gọi hàm hoặc thủ tục được dùng làm cơ chế IPC, hệ thống đầy đủ bao gồm tệp và các hàm (hoặc thủ tục) được viết tắt theo ngôn ngữ nào đó. Mã của các hàm nào được phân tán cho các bộ vi xử lý. Để thực hiện việc truyền thông giữa các máy, một hàm trên máy này có thể gọi một hàm trên máy khác. Ngữ nghĩa của các lời gọi hàm đây cũng giống như đối với các lời gọi hàm thông thường: *hàm gọi* bị *treo* cho đến khi *hàm gọi được* kết thúc, tham số được truyền từ *hàm gọi* cho đến *hàm được gọi*, còn kết quả được chuyển theo chiều ngược lại. Cách tiếp cận này dẫn đến một hệ điều hành được viết như một chương trình lớn, ưu điểm là chặt chẽ và nhất quán, tuy nhiên thiếu mềm dẻo.

Nếu dùng phương pháp chuyển thông báo của cơ chế IPC thì các tiến trình sẽ liên tục với nhau bằng cách chuyển thông báo. Mã của các tiến trình được tách biệt và có thể viết bằng các ngôn ngữ khác. Cách tiếp cận này đòi hỏi nhiều vấn đề hơn cách tiếp cận gọi hàm, chẳng hạn vấn đề địa chỉ hóa thiết lập các liên kết ảo, cắt, hợp thông báo, kiểm soát luồng dữ liệu truyền thông báo (broad casting).

1.4.3. Các kiểu hệ điều hàng mạng

Trên mạng cục bộ có hai kiểu hệ điều hành mạng: kiểu ngang hàng và kiểu dựa trên amý chủ:

1.4.3.1. Kiểu ngang hàng (peer-to-peer)

Mọi trạm đều có quyền bình đẳng như nhau và đều có thể cung cấp tài nguyên cho các trạm khác. Các tài nguyên cung cấp được có thể là tệp (tương ứng với thiết bị là đĩa), máy in. Nói chung trong các mạng ngang hàng không có việc biến một máy tính thành một trạm làm việc của một máy tính khác. Trong mạng ngang hàng, thông thường các máy sử dụng chung một hệ điều hành.

Win 3.1, Win 95, NT Workstation, AppleShare, Lanstic và Novell Lite là các hệ điều hành mạng ngang hàng .

Các đặc điểm của mạng ngang hàng:

- Thích hợp với các mạng cục bộ quy mô nhỏ, đơn lẻ, các giao thức riêng lẻ, mức độ thấp và giá thành rẻ.
- Các mạng ngang hàng được thiết kế chủ yếu cho các mạng nội bộ vừa và nhỏ và sẽ hỗ trợ tốt các mạng dùng một nền và một giao thức. Các mạng trên nhiều nền, nhiều giao thức sẽ thích hợp hơn với hệ điều hành có máy chủ dịch vụ.
- Yêu cầu chia sẻ file và máy in một cách hạn chế cần đến giải pháp ngang hàng.
- Người dùng được phép chia sẻ file và tài nguyên nằm trên máy của họ và truy nhập đến các tài nguyên được chia sẻ trên máy người khác, nhưng không có nguồn quản lý tập trung.
- Vì mạng ngang hàng không cần máy cụ thể làm máy chủ. Chúng thường là một phần của hệ điều hành nền hay là phần bổ sung cho hệ điều hành và thường rẻ hơn so với các hệ điều hành dựa trên máy chủ.
- Trong một mạng ngang hàng, tất cả các máy tính được coi là bình đẳng, bởi vì chúng có cùng khả năng sử dụng các tài nguyên có sẵn trên mạng.

Những thuận lợi:

- Chi phí ban đầu ít - không cần máy chủ chuyên dụng.
- Cài đặt - Một hệ điều hành có sẵn (ví dụ Win 95) có thể chỉ cần cấu hình lại để hoạt động ngang hàng.

Những bất lợi:

- Không quản lý tập trung được
- Bảo mật kém
- Có thể tốn rất nhiều thời gian để bảo trì

1.4.3.2. Kiểu hệ điều hành mạng có máy chủ (server based network)

Trong hệ điều hành kiểu này, có một số máy có vai trò cung cấp dịch vụ cho máy khác gọi là máy chủ (đúng hơn phải gọi là máy cung cấp dịch vụ – mà khi đó thì phải xem là máy "tớ").

Các dịch vụ có nhiều loại, từ dịch vụ tệp (cho phép sử dụng tệp trên máy chủ) , dịch vụ in (do một máy chủ điều khiển những máy in chung của mạng) tới các dịch vụ như thư tín, WEB, DNS ...

Trong mạng có máy chủ, hệ điều hành trên máy chủ và máy trạm có thể khác nhau. Ngay trong trường hợp máy chủ và máy trạm sử dụng cùng một hệ điều hành thì chức năng của bản trên máy chủ cũng có thể khác với chức năng cài đặt trên máy trạm.

Sau đây là một số hệ điều hành có dùng máy chủ: Novell Netware 4.1 Microsoft NT V4.0, Server, OS/2 LAN Server và Banyan Vines V6.0.

Đặc điểm của các hệ điều hành có máy chủ:

- Hệ điều hành cho các mạng an toàn, hiệu suất cao, chạy trên nhiều nền khác nhau (kể cả phần cứng, hệ điều hành và giao thức mạng)
- Một máy chủ là một máy tính trong mạng được chia sẻ bởi nhiều người dùng, như các máy dịch vụ file, máy dịch vụ in, máy dịch vụ truyền tin. Nói cách khác, nó được thiết kế để cung cấp một dịch vụ cụ thể - khác với các hệ máy tính nhiều người dùng, tập trung và đa mục đích - mặc dù máy dịch vụ file kết hợp với các hệ thống như hệ điều hành mạng Novell's NetWare 3.xx hay 4.xx thường hoạt động theo cách đó.
- Kiểm soát quyền sử dụng trên tòan mạng tại máy chủ.
- Cung cấp các dịch vụ thư mục trên tòan mạng.
- Các giải pháp dựa trên máy chủ được coi là sự quản trị mạng tập trung và thường là máy quản lý mạng nội bộ chuyên dụng.
- Bản thân máy chủ có thể chỉ là máy chủ chuyên dụng như Novell Netware 4.1, máy này không thể hoạt động như một máy trạm. Cũng có những hệ điều hành mà máy chủ NT cũng có thể được sử dụng như một máy trạm.

1.4.3.3. Mô hình khách/chủ (client/server)

Đầu thập niên 60, việc sử dụng máy tính thực hiện theo mô hình tập trung. Các trạm thực sự chỉ làm việc giao tiếp còn việc xử lý thực sự tiến hành ở một máy tính nào đó. Như vậy với mô hình này hoàn toàn không có xử lý cộng tác. Một phát triển tiếp theo là mô hình xử lý chủ tớ (master/slaver) với việc một máy xử lý và chuyển giao một số công việc cho các máy cấp thấp hơn, hoàn toàn không có việc máy cấp thấp hơn liên lạc hoặc giao việc theo chiều ngược lại. Như vậy quá trình cộng tác chỉ là một chiều.

Một bước đột phá trong mô hình tính toán cộng tác là mô hình chia sẻ thiết bị (shared device) theo đó một máy có thể cho máy khác sử dụng thiết bị của mình (chủ yếu là đĩa và máy in). Hệ điều hành mạng theo kiểu ngang hàng hay có sử dụng máy chủ dịch vụ đều có thể dùng cho mô hình này. Tuy nhiên chỉ ở mức này thôi thì chính CPU chưa bị chia sẻ nghĩa là chưa có sự phân tán trong xử lý mà chủ yếu là phân tán thông tin. Ngay cả việc sử dụng máy in từ xa cũng không mang ý nghĩa của xử lý phân tán vì thực chất chỉ là gửi nội dung in tới hàng đợi của một máy in do một máy tính nào đó quản lý mà thôi. Máy chủ cung cấp dịch vụ in không tạo ra giá trị mới cho công việc của máy uỷ thác dịch vụ in.

Trong những năm gần đây đã xuất hiện mô hình khách chủ trong đó một số máy chủ đóng vai trò cung úng dịch vụ theo yêu cầu của các máy trạm. Máy trạm trong mô hình này gọi là máy khách (client) là nơi gửi các yêu cầu xử lý về máy chủ. Máy chủ (server) xử lý và gửi kết quả về máy khách. Máy khách có thể tiếp tục xử lý các kết quả này phục vụ cho cộng việc. Như vậy máy khách chịu trách nhiệm chủ yếu về giao diện và chỉ đảm nhận một phần xử lý. Trong mô hình khách/chủ xử lý thực sự phân tán.

Ta nói đến mô hình khách chủ chứ không nói đến hệ điều hành khách chủ vì trên thực tế mô hình khách chủ yêu cầu phải có một hệ điều hành dựa trên máy chủ dù máy chủ này ở trong mạng cục bộ hay máy chủ cung cấp dịch vụ từ một mạng khác. Hầu hết các ứng dụng trên Internet là ứng dụng khách chủ sử dụng từ xa.

Lưu ý rằng các tiến trình khách và chủ đôi khi có thể thực hiện trên cùng một máy tính

- Client process và server process có thể hoạt động trên cùng một bộ xử lý, trên các bộ sử lý khác nhau ở cùng một máy (các bộ xử lý song song), hoặc trên các bộ xử lý khác nhau trên các máy khác nhau (xử lý phân tán).
- Một điều quan trọng cần nhận thấy là cả hệ điều hành ngang hàng và hệ điều hành dựa trên máy chủ đều có thể thỏa mãn mô hình khách/chủ. Trên thực tế, hầu hết các hệ điều hành hiện đại đều cung cấp ít nhất một vài chức năng khách-chủ.

Hệ điều hành khách/chủ

Các hệ điều hành cho cấu trúc khách/chủ bao gồm: Sun Solaris NFS, UnixWare NFS, Novell Netware và Windows NT Server.

- Hệ điều hành khách/chủ cho phép mạng tập trung các chức năng và các ứng dụng tại một hay nhiều máy dịch vụ file chuyên dụng. Theo cách này, chúng có thể hoạt động như trường hợp đặc biệt của hệ điều hành dựa trên máy chủ.
- Các máy dịch vụ file trở thành trung tâm củ hệ thống, cung cấp sự truy cập tới các tài nguyên và cung cấp sự bảo mật. Các máy trạm riêng lẻ (máy khách) được truy nhập tới các tài nguyên có sẵn trên máy dịch vụ file.
- OS cung cấp cơ chế tích hợp tất cả các bộ phận của mạng và cho phép nhiều người dùng đồng thời chia sẻ cùng một tài nguyên bất kể vị trí vật lý
- Các hệ điều hành ngang hàng cũng có thể hoạt động như hệ điều hành khách/chủ như với Unix/NFS và Windows 95.

Các điểm thuận lợi của một mạng khách/chủ:

- Cho phép cả điều khiển tập trung và không tập trung: Các tài nguyên và bảo mật dữ liệu có thể được điều khiển qua một máy chủ chuyên dụng hay rải rác trên tòan mạng.
- Chống quá tải mạng
- Cho phép sử dụng các máy, các mạng chạy trên các nền khác nhau
- Đảm bảo toàn vẹn dữ liệu
- Giảm chi phí phát triển hệ thống

1.4.4. Các chức năng của một hệ điều hành mạng

Sau đây là các chức năng cụ thể mà một hệ điều hành mạng.

– Cung cấp phương tiện liên lạc giữa các tiến trình, giữa những người sử dụng và giữa các tài nguyên nói chung của toàn mạng. Có thể kể đến các khía cạnh sau:

  + Chuyển dữ liệu giữa các tiến trình

  + Đồng bộ hoá các tiến trình

  + Cung cấp phương tiện liên lạc giữa người sử dụng. Ở mức thấp có thể là tạo, lưu chuyển và hiển thị các thông báo nóng trực tuyến, ở mức độ cao có thể là nhắn tin (paging) hoặc thư tín điện tử (Email)

– Hỗ trợ cho các hệ điều hành của máy trạm - cho phép truy nhập tới máy chủ từ các máy trạm. Các hệ điều hành mạng hiện đại đều cung cấp các hỗ trợ cho các hệ điều hành khác nhau chạy trên các máy trạm khách. Sau đây là một số ví dụ minh hoạ vấn đề này:

  Các hệ điều hành UNIX cung cấp các chương trình chạy trên DOS có tên là NFS (Network File System) khởi động trên DOS để các máy PC có thể sử dụng hệ thống tệp của các máy chủ UNIX.

  Một số hệ điều hành như Windows NT và Windows 95 cung cấp hỗ trợ cho các dịch vụ thư mục Novell (NDS) cho phép chúng truy nhập trực tiếp tới tài nguyên trên máy chủ Novell Netware.

– Dịch vụ định tuyến và cổng nối - cho phép truyền thông giữa các giao thức mạng khác nhau. Ví dụ một máy chạy trên Novell NetWare với giao thức IPX/SPX không thể chạy trực tiếp các ứng dụng trên TCP/IP như một số các ứng dụng Internet. Tuy vậy nếu có các modun chuyển đổi giao thức biến các gói tin IPX/SPX thành gói tin TCP/IP khi cần gửi từ mạng Netware ra ngoài và ngược lại thì một máy chạy Netware có thể giao tiếp được với Internet. Kiến trúc của Netware có ODI (Open Datalink Interface) là phần để chuyển đổi và chồng (bao gói) các giao thức khác nhau.

– Dịch vụ danh mục và tên. (Name /Directory Services)

  + Để có thể khai thác tốt tài nguyên trên mạng, NSD cần “nhìn thấy” một cách dễ dàng các tên tài nguyên (thiết bị, tệp) của toàn mạng một cách tổng thể. Vì thế một dịch vụ cung cấp danh mục tài nguyên là vô cùng quan trọng.

  + Đương nhiên việc NSD nhìn thấy các tài nguyên nào còn phụ thuộc vào thẩm quyền của người đó. Mỗi khi vào mạng, khi NSD đã được mạng nhận diện, họ có thể nhìn thấy những tài nguyên được phép sử dụng.

  + Trong NOVELL dịch vụ đó chính là NDS (Netware Directory Services). Trong Windows NT hay Windows95 đó chính là chức năng browser mà ta thấy được cài đặt trong explorer. Trong UNIX với lệnh mount ta có thể kết nối tên tài nguyên của một hệ thống con vào hệ thống tài nguyên chung.

– Bảo mật – Chức năng này đảm bảo việc kiểm soát các quyền truy cập mạng, quyền sử dụng tài nguyên của mạng. Các phương pháp được áp dụng bao gồm :

  Dùng các dịch vụ đĩa để điều khiển bảo mật:

+ Chia ổ đĩa cứng của máy chủ thành các phần được gọi là *volume* hay *partition* sau đó gán *volume* được phép cho người dùng

+ Định các thẩm quyền trên tệp và thư mục. Có nhiều loại thẩm quyền. It nhất thì các thẩm quyền được đọc, được ghi và được thực hiện được áp dụng cho đa số các hệ điều hành mạng. Một số hệ điều hành quy định thẩm quyền khá chi tiết như quyền được xoá, quyền được sao chép, quyền xem thư mục, quyền tạo thư mục. Các quyền này lại được xem xét cho đến từng nhóm đối tượng như cá nhân, nhóm là việc hay tất cả mọi người.

+ Thẩm quyền vào mạng hay thực hiện một số dịch vụ được nhận diện qua tên người sử dụng và mật khẩu.

+ Mã hoá các gói tin trên mạng.

+ Một số hệ điều hành còn cho phép mã hoá phần cứng để kiểm soát việc sử dụng thiết bị.

– Cung cấp phương tiện chia sẻ tài nguyên. Những tài nguyên trên mạng có thể cho phép nhiều người được sử dụng. Đáng kể nhất là đĩa (thực chất là tệp và thư mục) và máy in (thực chất là máy tính quản lý hàng đợi của máy in). hệ điều hành M phải có các công cụ cho phép tạo ra các tài nguyên có thể chia sẻ được. Các tài nguyên chia sẻ được phải là các tài nguyên độc lập với mọi ứng dụng. Chính vì vậy nó phải được cung cấp các trình điều khiển (driver) phù hợp với mạng. Máy in, modem .... là các tài nguyên như vậy. Trên mạng cũng cần có các công cụ can thiệp vào hoạt động của các tài nguyên mạng ví dụ: đình chỉ một tiến trình truy nhập mạng từ xa, thay đổi thứ tự hàng đợi trên máy in mạng...

– Tạo tính trong suốt để người sử dụng không nhìn thấy khó khăn trong khi sử dụng các tài nguyên mạng cũng như tài nguyên tại chỗ. Chính dịch vụ thư mục và tên nói trên là một ví dụ về chức năng này. Trong Windows 95/NT người ta có thể duyệt thư mục trên toàn mạng không có gì khác với việc duyệt thư mục trong đĩa cục bộ

– Sao lưu dự phòng - Đối với bất kỳ hệ thống nào, chạy trên môi trường nào, vấn đề sao lưu dự phòng cũng quan trọng để có thể hồi phục thông tin của hệ thống sau một sự cố gây mất dữ liệu. Tuy nhiên trong môi trường mạng thì việc sao lưu có thể thực hiện được việc sao lưu một cách tự động qua mạng. Chính vì thế các các hệ điều hành mạng đều cung cấp công cụ sao lưu như một chức năng cơ bản. Có nhiều phương pháp sao lưu. Trên Novell cho phép soi gương (mirroring) các ổ đĩa mà ta có thể đặt trong khi cài đặt hệ thống. Novell có cả một dịch vụ tên là SMS (Storage Management Services) cung cấp các công cụ sao chép, hồi phục không chỉ dữ liệu của NSD mà cả dữ liệu của hệ thống ví dụ NDS. NT có chức năng replicate không những đối với đĩa mà còn ở mức thư mục và định kỳ. Điều đó rất cần thiết không chỉ trên mạng cục bộ mà ngay cả trên mạng rộng.

1.5. KẾT NỐI LIÊN MẠNG

1.5.1. Các tiếp cận

Liên mạng (Internetwork) là một tập hợp của nhiều mạng riêng lẻ được nối kết lại bởi các thiết bị nối mạng trung gian và chúng vận hành như chỉ là một mạng lớn. Để kết

nối các mạng đang tồn tại lại với nhau, người ta thường xuất phát từ một trong hai quan điểm sau:

1) Xem mỗi nút của mạng con như là một hệ thống mở
2) Xem mỗi mạng con như là một hệ thống mở.

Quan điểm 1 cho phép mỗi nút của mạng con có truyền thông trực tiếp với một nút của một mạng con bất kỳ khác. Như vậy toàn bộ các mạng con cũng sẽ là nút một mạng lớn hơn và tuân thủ một kiến trúc chung.

Trong khi đó với quan điểm 2, hai nút thuộc hai mạng con khác nhau không thể "bắt tay" trực tiếp với nhau được mà phải thông qua một phần tử trung gia, gọi là giao diện kết nối đặt giữa hai mạng con đó. Có nghĩa là cũng hình thành một mạng mạng lớn hơn gồm các giao diện kết nối và các máy tính (host) được nối với nhau bởi các mạng con đó.

Tương ứng với hai quan điểm đó có hai chiến lược kết nối mạng khác nhau. Một chiến lược (tương ứng với quan điểm 1) tìm cách xây dựng các chuẩn chung cho các mạng (các chuẩn của ISO, CCITT theo quan điểm này). Một chiến lược khác (tươngứn với quan điểm 2) cố gắng xây dựng các giao diện kết nối để tôn trọng tính độc lập của các các mạng hiện có. Rõ ràng sự hội tụ về một chuẩn chung là một điều lý tưởng, nhưng rõ ràng là không thể ngay tức khắc loại bỏ hàng vạn mạng đang tồn tại trên thế giới được, mà phải tìm cách tận dụng chúng. Trên thị trường hiện nay có rất nhiều các sản phẩm giao diện cho phép chuyển đổi giữa các mạng khác nhau, đó là một minh chứng sống động cho sức sống của quan điểm 2.

1.5.2. Giao diện kết nối

Kết nối liên mạng có thể được thực hiện ở những tầng khác nhau, tùy thuộc vào mục đích mà ta dùng các thiết bị kết nối khác nhau. Bảng dưới đây liệt kê một số thiết bị kết nối tương ứng với các tầng khác nhau:

Tầng nối kết	**Mục đích**	**Thiết bị sử dụng**
Tầng vật lý	Tăng số lượng và phạm vi mạng LAN	HUB / Repeater
Tầng liên kết dữ liệu	Nối kết các mạng LAN có tầng vật lý khác nhau Phân chia vùng đụng độ để cải thiện hiệu suất mạng	Cầu nối (Bridge) Bộ hoán chuyển (Switch)
Tầng mạng	Mở rộng kích thước và số lượng máy tính trong mạng, hình thành mạng WAN	Router
Các tầng còn lại	Nối kết các ứng dụng lại với nhau	Gateway

1.6. CÂU HỎI VÀ BÀI TẬP

(đang tiếp tục bổ sung)

CHƯƠNG 2. KIẾN TRÚC PHÂN TẦNG OSI

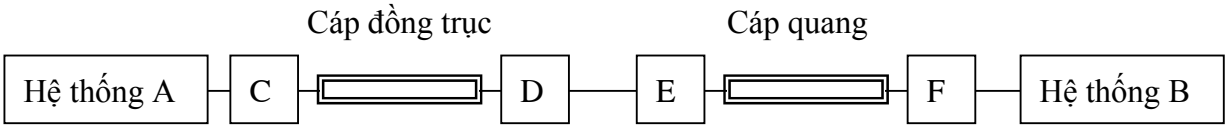
2.1. TẦNG VẬT LÝ (PHYSICAL)

2.1.1. Vai trò và chức năng của tầng vật lý.

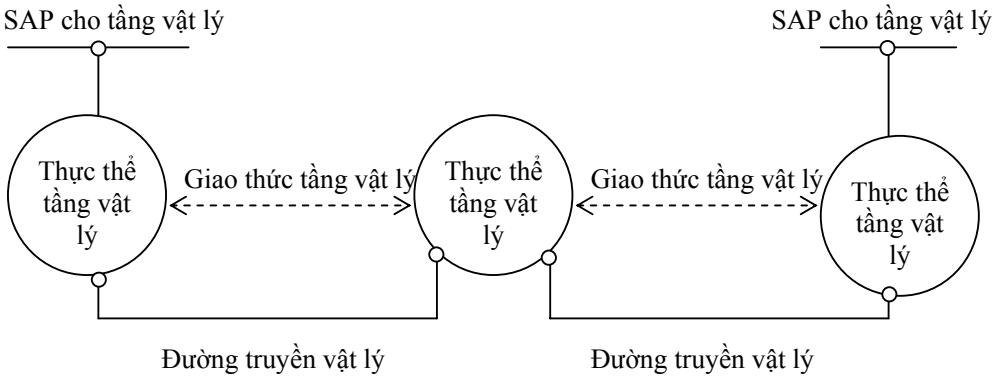
Theo định nghĩa của ISO, tầng vật lý cung cấp các phương tiện điện, cơ, chức năng, thủ tục để kích hoạt, duy trì và đình chỉ các liên kết vật lý giữa các hệ thống.

Trong đó, thuộc tính điện liên quan đến sự biểu diễn các bít (các mức tín hiệu) và tốc độ truyền các bit, thuộc tính cơ liên quan đến các tính chất vật lý của giao diện vật lý với một đường truyền (kích thước, cấu hình). Thuộc tính chức năng chỉ ra các chức năng được thực hiện bởi các phần tử của giao diện vật lý giã một hệ thống và đường truyền vật lý, và thuộc tính thủ tục liên quan đến các giao thức điều khiển việc truyền các xâu boit qua đường truyền vật lý.

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.



a) Môi trường thực



b) Môi trường logic

Hình 2.1 Quan hệ của tầng vật lý với môi trường thực

Trong hình 2.1 a), A và B là hai hệ thống mở được nối với nhau bằng một đoạn cáp đồng trục và một đoạn cáp quang. Modem C để chuyển đổi tín hiệu từ tín hiệu số sang tín hiệu tương tự để truyền trên cáp đồng, và modem D lại chuyển đổi tín hiệu từ tín hiệu tương tự sang tín hiệu số. Transducer E chuyển đổi từ xung điện thành xung ánh sáng để truyền qua các quang. Cuối cùng Transducer F chuyển đổi thành xung điện để đi vào B.

Hình 2.1 b) là môi trường logic của tầng vật lý. Ở đây, một thực thể vật lý là một cấu trúc logic giao diện với đường truyền vật lý. Các thực thể đó có mặt trong các hệ

thống A, B và cũng có thể có thực thể vạtt lý ở giao diện giữa D và E. Thực thể trung gian này là một bộ chuyển tiếp hoạt đọng ở tầng vật lý giao diện với các đường truyền vậy lý khác nhau.

Một giao thức tầng vật lý giữa các thực thể vật lý để quy định pgương thức truyền (đồng bộ, dị bộ) và tốc độ truyền,.. Điều mong muốn là giao thức đó pải độc lập tối đa với đường truyền vật lý để cho một hệ thống có thể giao diện với nhiều đường truyền khác nhau. Do vậy, các chuẩn vật lý sẽ phải bao gồm không chỉ các thực thể mà còn cả đặc tả của giao diệ với đườn truyền. Dưới đây ta sẽ xem sét các chuẩn đó.

2.1.2. Các chuẩn cho giao diện vật lý

Trước khi vào phần này chúng ta hãy làm quen với hai thuật ngữ mới, đó là *thiết bị cuối dữ liệu* (Data Terminal Equipment – DTE) và *thiết bị cuối kênh dữ liệu* (Data Circuit Terminal Equipment – DCE).

DTE là một thuật ngữ chung để chỉ các máy của người sử dụng cuối (end-user), có thể là máy tính hoặc một trạm cuối (terrminal). Tất cả các ứng dụng của người dùng đều nằm ở DTE. Mục đích của việc nối mạng chính là để nối các DTE lại với nhau để chia sẻ tài nguyên, lưu trữ thông tin chung và trao đổi dữ liệu.

DCE là thuật ngữ chung chỉ các thiết bị làm nhiệm vụ kết nối các DTE với đường truyền. Nó có thể là một Modem, Transducer, Multiplexing. DCE có thể được cài đặt ngay bên trong DTE hoặc đứng riêng như một thiết bị độc lập. Chức năng chủ yếu của nó là chuyển đổi tín hiệu biểu diễn dữ liệu của người dùng thành tín hiệu chấp nhận được bởi đường truyền và ngược lại.

Trong hình 2.1 ở trên, các hệ thống mở A, B chính là các DTE, còn các Modem C, D và Tranducer E, F đóng vai trò là các DCE.

Đa số các trường hợp kết nối mạng máy tính sử dụng cùng một kiểu giao diện vật lý để thuận tiện cho việc truyền thông trực tiếp giữa các sản phẩm khác loại, khỏi phải thực hiện việc chuyển đổi rắc rối. Các đặc tả về hoạt động của các DTE và DCE được đưa ra bởi nhiều tổ chức chuẩn hóa như CCITT, EIA và IEEE. ISO cũng đã công bố các đặc tả về các đầu nối cơ học kết nối giữa các DCE và DTE.

Việc truyền dữ liệu chủ yếu được thực hiện thông qua mạng điện thoại, bởi thế các tổ chức trên đã đưa ra nhiều khuyến nghị về vấn đề này. Các khuyến nghị loại V và loại X của CCITT là một ví dụ điển hình. Chúng là các đặc tả ở tầng vật lý được sử dụng phổ biến nhất trên thế giới, đặc biệt là ở Tây Âu. Bên cạnh đó các chuẩn thuộc họ RS- (nay đã đổi thành EIA-) của EIA cũng đã được sử dụng rất phổ biến, đặc biệt là ở Bắc Mỹ. Dưới đây ta sẽ xem xét một số chuẩn thông dụng nhất.

- V24/RS-232-C:

Là hai họ chuẩn tương ứng của CCITT và EIA nhằm định nghĩa giao diện vật lý giữa DTE và DCE (giữa máy tính và Modem chẳng hạn). Về phương diện cơ, các sản phẩm này sử dụng các đầu nối 25 chân (25- pin connector). Về điện, các chuẩn này quy định các tín hiệu số nhị phân 0 và 1 tương ứng với các thế hiệu nhỏ hơn -3V và lớn hơn +3V. Tốc độ tín hiệu không vượt quá 20 Kbps với khoảng cách tối đa là 15m.

Trong trường hợp đặc biệt, khi khoảng cách giữa các thiết bị quá gần đến mức cho phép hai DTE có thể truyền trực tiếp tín hiệu với nhau, lúc đó các mạch RS-232-C vãn có thể được dùng nhưng không cần có mặt DCE nữa.

Từ năm 1987, RS-232-C đã được sửa đổ và dặt tên lại là EIA-232-D.

- RS-449/422-A/423-A:

Nhược điểm chính của V24/RS-232-C là sự hạn chế về tốc đội và khoảng cách. Để cải thiện yếu điểm đó, EIA đã đưa ra mootj tập các chuẩn mới để thay thế, đó là RS-449, RS-422-A và RS-423-A. Mặc dù chuẩn RS-232-C vẫn được sử dụng nhiều nhất cho giao diện DET/DCE, nhưng các chuẩn mới nói trên cũng đang ngày càng được sử dụng nhiều hơn. RS-449 định nghĩa các dặc trưng cơ, chức năng, còn RS-422-A và RS-4232-A định nghĩa các đặc trưng về điện của chuẩn mới.

RS-449 tương tự như RS-232-C và có thể liên tác với chuẩn cũ,. Về phương diện chức năng, RS-449 giữ lại toàn bộ các mạch của RS-232-C (trừ mạch AA), và thêm vào 10 mạch mới, trong đó có các mạch quan trọng là: IS, NS, SF, LL, RL, TM.

Về phương diện cơ, RS-449 dùng đầu nối 37-chân cho giao diện cơ bản và dùng một đầu nối 9 chân riêng biệt cho kênh phụ. Song trong nhiều trường hợp, chỉ có một số chân được sử dụng.

Về phương diện thủ tục, RS-449 tương tự như RS-232-C. Mỗi mạch có chức năng riêng và việc truyền tin dựa trên các cặp "tác động-phản ứng". Ví dụ DTE thực hiện Request to Send thì sau đó nó sẽ đợik DCE trả lời với Clear to Send.

Cải tiến chủ yếu của RS-449 so với RS-232-C là ở các dặc trưng về điện, và các chuẩn RS-422-A, RS-423-A định nghĩa các đặc trưng đó. Trog khi RS-232-C được thiết kế ở thời đại của các linh kiện điện tử rời rạc thì các chuẩn mơi đã được tiếp nhận các ưu việt của công nghệ mạc tổ hợp (IC). RS-423-A sử dụng phương thức truyền thông không cân bằng, đạt tốc độ 3Kbps ở khoảng cách 1000m và 300 Kbps ở khoảng cách 10m. Ttrong khi đó, RS-422-A sử dụng phương thức truyền thông cân bằng, đạt tốc độ 100Kbps ở khoảng cách 1200m và tới 10 Mbps ở khoảng cách 12m.

Ngoài các chuẩn trên EIA còn hpát triển các chuẩn khác như EIA-530 để thay thế cho EIA-232 trong trường hợp các giao đòi hỏi tốc độ cao hơn 20Kbps, hau EIA-366 định nghĩa giao diện cho các thiết bị tự động, một modem và một DTE.

2.2. TẦNG LIÊN KẾT DỮ LIỆU (DATA LINK)

2.2.1. Vai trò và chức năng của tầng 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 sóat lỗi và kiểm soát luồng dữ liệu.

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 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.

2.2.2. Các giao thức của tầng liên kết dữ liệu

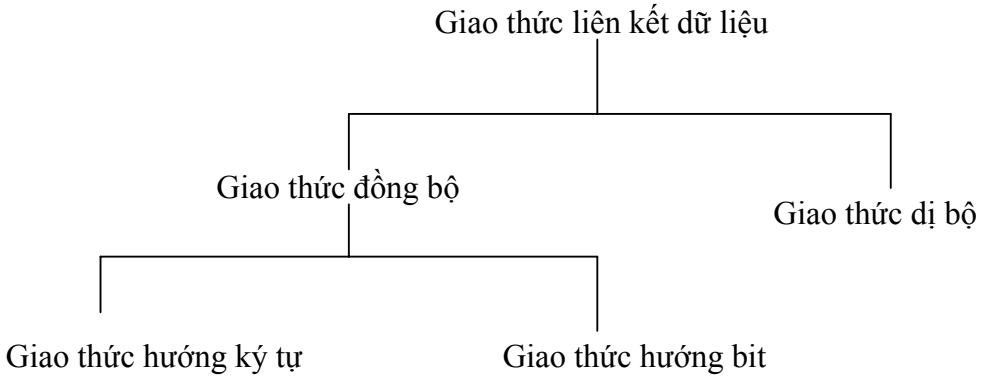
Cũng giống như tầng Vật lý, có rất nhiều giao thức được xây dựng cho tầng này, gọi chung là các giao thức liên kết dữ liệu (Data Link Protcol- DLP). Cá DLP được phân chia thành hai loại: đồng bộ và dị bộ. Trong đó, loại đồng bộ lại được chia thành 2 nhóm là hướng ký tự và hướng bit (xem sơ đồ hình 2.2).

- DLP dị bộ:

Các DLP dị bộ sử dụng phương thức truyền dị bộ, tức là không cần có sự đồng bộ liên tục giữa người gửi và người nhận, Nó cho phép một đơn vị dữ liệu được truyền đi bất kỳ lúc nào mà không cần quan tâm đến các tín hiệu đồng bộ trước đó. Ở giao thức loại này, các bít đặc biệt START và STOP được dùng để tách các xâu bit biểu diễn các ký tự trong dòng dữ liệu được truyền đi. Các giao thức loại này thường được dùng trong các máy điện báo hoặc các máy tính trạm cuối tốc độ thấp. Phần lớn các máy PC sử dụng phương thức truyền dị bộ vì tónh đơn giản của nó.

- DLP đồng bộ:

Phương thức truyền thông đồng bộ sử dụng các ký tự đặc biệt SYN, EOT hay đơn giản là các cờ (flag) giữa ác kdữ liệu củangười dùng để báo cho người nhận biết rằng dữ liệu "đang đến" hay "đã đến".



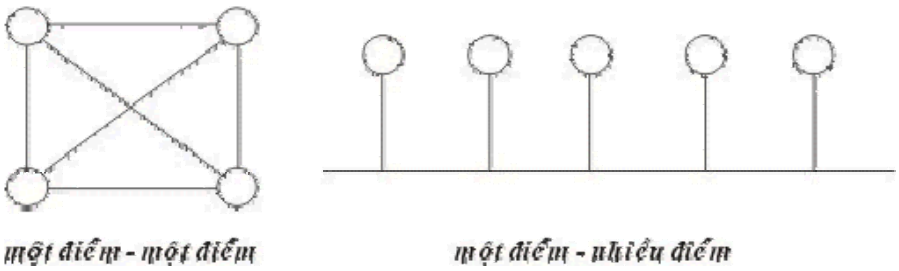
Hình 2.2. Phân loại các giao thức liên kết dữ liệu

Các giao thức tầng liên kết dữ liệu đồng bộ gồm các giao thức hướng ký tự 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.

Dưới đây chúng ta sẽ xem xét kỹ hơn hai loại giao thức này.

2.2.3. Các giao thức hướng ký tự

Các giao thức loại này xuất hiện từ những năm 60 và đến nay nó vẫn được sử dụng. Chúng được dùng cho cả hai phương thức truyền 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 2.3. Các đường truyền kết nối kiểu "một điểm - một điểm" và "một điểm - nhiều điểm".

Các giao thức loại này có thể đáp ứng cho các phương thức khai thác đường truyền khác nhau: *một chiều* (simplex), *hai chiều luân phiên* (half-duplex) và *hai chiều đồng thời* (full-duplex).

Đối với phương thức *một chiều*, giao thức hướng ký tự được dùng rộng rãi nhất là giao thức truyền tệp Kermit do Đại học Columbia (Mỹ) chế tác. Kermit có nhiều phiên bản cho phép truyền tệp giữa 2 máy PC, hoặc một PC và một máy chủ (file server) hoặc một máy tính lớn (maiframe).

Đối với phương thức *hai chiều luân phiên*, giao thức hướng ký tự nổi tiếng nhất là giao thức BSC (Binary Synchronous Control) hay còn gọi là Bisync- một sản phẩm của IBM. Giao thức này đã được lấy Iso lấy làm cơ sở để xây dựng giao thức hướng ký tự chuẩn quốc tế với tên gọi là Basic Mode. Bởi vậy ta sẽ trình bày chi tiết về giao thức này.

Có rất ít giao thức hướng ký tự cho phương thức *hai chiều đồng thời*. Ví dụ cho loại này là giao thức giữa các nút chuyển mạch (IMP – Interface Message Protcol) trong mạng ARPANET của bộ quốc phòng Mỹ.

Giao thức BSC/Basic mode

Họ giao thức này áp dụng cho trường hợp điểm-điểm, hoặc điểm-nhiều điểm và hai chiều luân phiên; sử dụng các ký tự đặc biệt của bộ mã EBCDIC (đối với BSC) và ASCII (đối với Basic Mode).

Các ký tự đặc biệt đó gồm:

SOH (Start Of Header): chỉ bắt đầu của phần header

STX (Start Of Text): chỉ phần bắt đầu của phần dữ liệu (văn bản)

ETX (End Of Text): chỉ phần kết thúc của phần dữ liệu

EOT (End Of Transmission): chỉ sự kết thúc của một hoặc nhiều đơn vị dữ liệu và giải phóng liên kết).

ETB (End Of Transmission Block): chỉ sự kết thúc của một khối dữ liệu trong trường hợp dữ liệu được chia thành nhiều khối.

ENQ (Enquiry): để yêu cầu phúc đáp từ một trạm ở xa.

DLE (Data Link Escape): để thay đổi ý nghĩa của các ký tự điều khiển khác

ACK (Acknowledge): để báo cho người gửi biết đã nhận tốt dữ liệu

NAK (Negative Acknowledge): để báo cho người gửi biết đã nhận không tốt dữ liệu

SYN (Synchronous Idle): ký tự đồng bộ, dùng để duy trì sự đồng bộ giữa người gửi và người nhận.

Đơn vị dữ liệu (frame) của nó có khuôn dạng như sau:

SOH	Header	STX	Text	ETX	BCC

Trong đó BCC(block Check Character): là 8 bit kiểm tra lỗi theo kiểu bit chẵn lẻ theo khối cho các ký tự thuộc vùng Text (trong trường hợp Basic Mode), hoặc 16 bit kiểm tra lỗi theo phương pháp CRC-16 cho vùng Text (trong trường hợp BSC). Các phương pháp kiểm tra lỗi sẽ được đề cập trong chương 4. Kích thước vùng Text được giới hạn để đảm bảo được kiểm soát lỗi khi truyền. Trong trường hợp dữ liệu lớn thì có thể chia thành nhiều khối nhỏ (block). Giả sử Text được chia làm 3 khối, khi đó khuôn dạng các khối dữ liệu như sau:

Khối 1:

SOH	Id	Header	STX	Text1	ETB	BCC

Khối 2:

SOH	Id	STX	Text2	ETB	BCC

Khối 3:

SOH	Id	STX	Text3	ETB	BCC

Các thủ tục chính của BSC/Basic Mode:

- *Mời truyền tin:*

Giả sử trạm A muốn mời trạm B truyền tin, A sẽ gửi lệnh sau tới B:

EOT	B	ENQ

Trong đó: B là địa chỉ của trạm được mời truyền tin,

EOT để chuyển liên kết sang trạng thái điều khiển.

Khi B nhận được lệnh này, có thể xảy ra hai trường hợp,:

- Nếu có tin để truyền thì trạm B sẽ cấu tạo một đơn vị dữ liệu và gửi cho A.
- Nếu không có tin để gửi, B sẽ gửi EOT để trả lời.

Ở phía A, sau khi gửi lệnh đi quá một thời gian xác định trước mà không nhận được trả lời của B, hoặc là nhận được trả lời sai thì A sẽ chuyển sang trạng thái "phục hồi". Trạng thai này sẽ được nói đến ngay sau đây.

- *Mời nhận tin:*

Giả sử trạm A muốn mời trạm B nhận tin, A sẽ gửi lệnh tương tự như trên tới B:

EOT	B	ENQ

Ở đây EOT có thể vắng mặt.

Khi B nhận được lệnh này, nếu B sẵn sàng nhận tin thì trạm B sẽ gửi ACK về A, ngược lại nó sẽ gửi NAK.

Ở phía A, sau khi gửi lệnh đi quá một thời gian xác định trước mà không nhận được trả lời của B, hoặc là nhận được trả lời sai thì A sẽ chuyển sang trạng thái "phục hồi".

- *Yêu cầu trả lời:*

Khi một trạm cận trạm khác trả lời một yêu cầu nào đó đã gửi đi trước, nó chỉ cần gửi lệnh ENQ cho trạm kia.

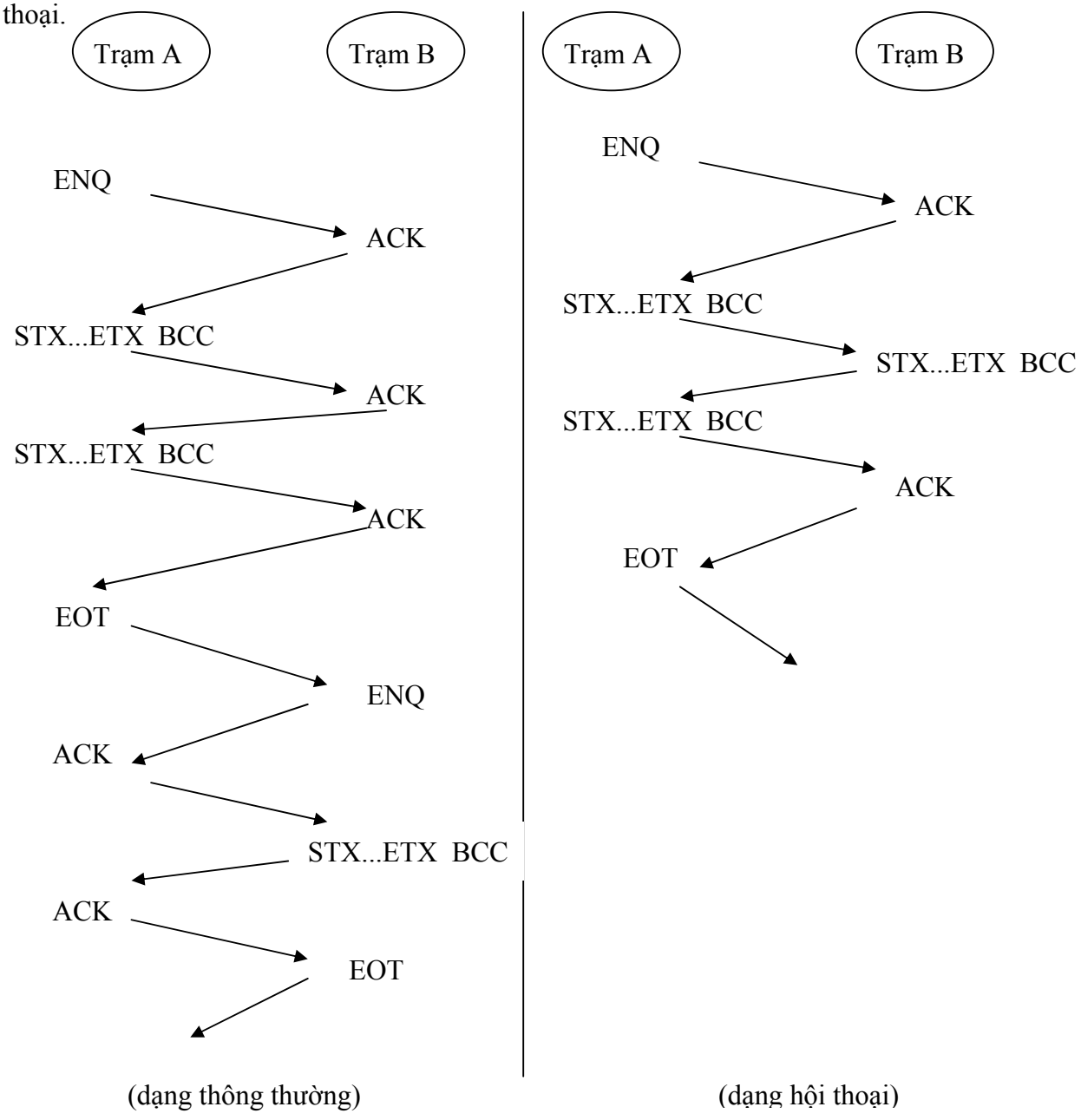
- *Ngừng truyền tin (tạm thời):* gửi lệnh EOT
- *Giải phóng liên kết:* gửi lệnh DLE EOT
- *Trạng thái phục hồi:* Khi một trạm nào đó đi vào trạm thái phục hồ nó sẽ thực iện một trong các hành đọng sau:

– Lặp lại lệnh đã gửi n lần (n là một soos nguyên chọn trước)

– Gửi "yêu cầu trả lời" n lần

– Kết thúc truyền bằng cách gửi lệnh EOT

Để thấy rõ hơn phương thức trao đổi thông tin của giao thức BSC/Basic Mode ta dùng sơ đồ minh họa ở hình ... dưới đây, trong đó có hai trường hợp: thông thường và hội thoại.



Hình 2.4. Sơ đồ minh họa hoạt động của giao thức BSC/Basic Mode

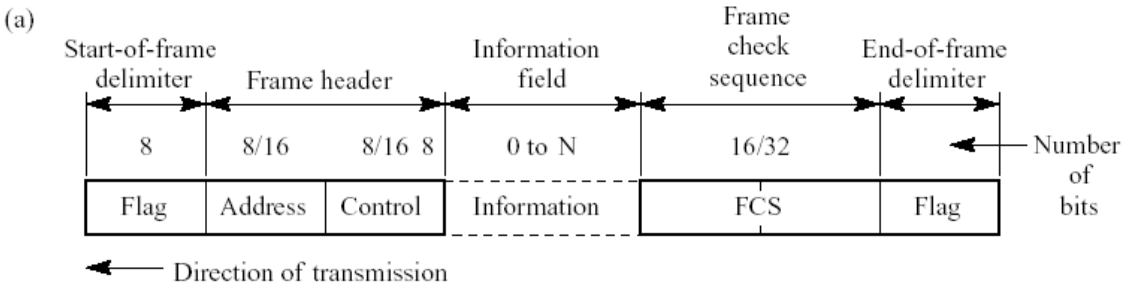
2.2.4. Các giao thức hướng bit

Giao thức HDLC

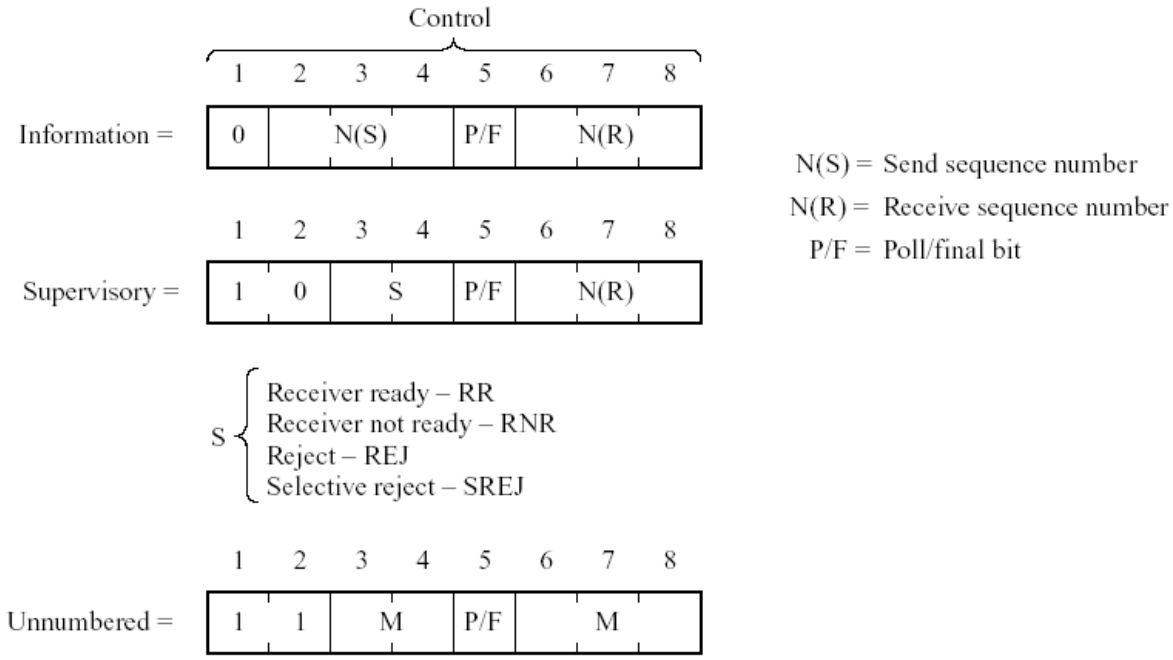
HDLC hỗ trợ 3 chế độ trao đổi số liệu

- NRM (Normal Response Mode) = chế độ trả lời bình thường: được sử dụng ở cấu hình không cân đối, S chỉ phát khi có yêu cầu của P.
- ARM (Asynchronous Response Mode) = chế độ trả lời không đồng bộ: được sử dụng ở cấu hình không cân đối, cho phép S phát không cần nhận được yêu cầu của P.
- ABM (Asynchronous Balanced Mode) = chế độ trả lời không đồng bộ ở cấu hình cân đối; hầu như chỉ được sử dụng trong mạng kết nối point-to-point + full-duplex. Hai thiết bị trao đổi với nhau là bình đẳng về chức năng (P và S)

Khuôn dạng gói số liệu (HDLC Frame Format)



- Flag - trường đồng bộ = 7EH = 0111.1110
- Address - trường địa chỉ, chứa đ/c thiết bị đích
  - + Group address
  - + Broadcast address



- Control - trường điều khiển: kết nối, truyền và kết thúc kết nối

Gói số liệu I-Frame:

- N(S), N(R) được sử dụng để điều khiển lưu lượng thu/phát. Ngoài ra N(S), N(R) còn xác định độ lớn của cửa sổ được sử dụng để trao đổi số liệu bằng HDLC.
- P/F= Poll/Final
  - P/F = 1 = P: yêu cầu S phải thực hiện lệnh và trả lời kết quả thực hiện; S báo cáo đã thực hiện lệnh
  - P/F = 0 = F: Hết thông tin cần gửi

Gói điều khiển S-Frame:

- bit P/F giống như trên
- S = 00: RR (Receive Ready) - sẵn sàng nhận, đã nhận tới gói tin thứ N(R)-1
- S = 01: REJ (Reject) - yêu cầu phát lại từ N(R)
- S = 10: RNR(Receive Not Ready) - chưa sẵn sàng, đã nhận tới N(R)-1
- S = 11: SREJ (Selative REJ) - yêu cầu phát lại có chọn lọc, chỉ riêng N(R)

Gói điều khiển U-Frame: Báo nối/tách hệ thống

- **SARM** (1 1 1 1 P 0 0 0): yêu cầu nối có phân biệt Master/Slave, tuy vậy Slave có thể hỏi.
- **SNRM** (1 1 0 0 P 0 0 1): yêu cầu nối ở mode bình thường, có Master/Slave, Slave không được hỏi, chỉ được phép trả lời.
- **SABM** (1 1 1 1 P 1 0 0): không phân biệt máy chính, máy phụ, cả hai máy coi như nhau; nếu P=1 thì yêu cầu trả lời.
- **DISC** (1 1 0 0 P 0 1 0): yêu cầu tách hệ thống, nếu trả lời UA tức là đồng ý. UA (1 1 0 0 F 1 1 0): thông báo trả lời. (Control frame cũng có thể bị mất, giống như các frame số liệu, vì thế cũng cần biên nhận (ACK). Frame đặc biệt dành cho mục đích này là UA).

Nguyên tắc hoạt động của HDLC

Quản trị thiết lập và giải phóng kết nối (V(x) = seq. #):

a) NRM – multidrop link, truyền 1 hướng

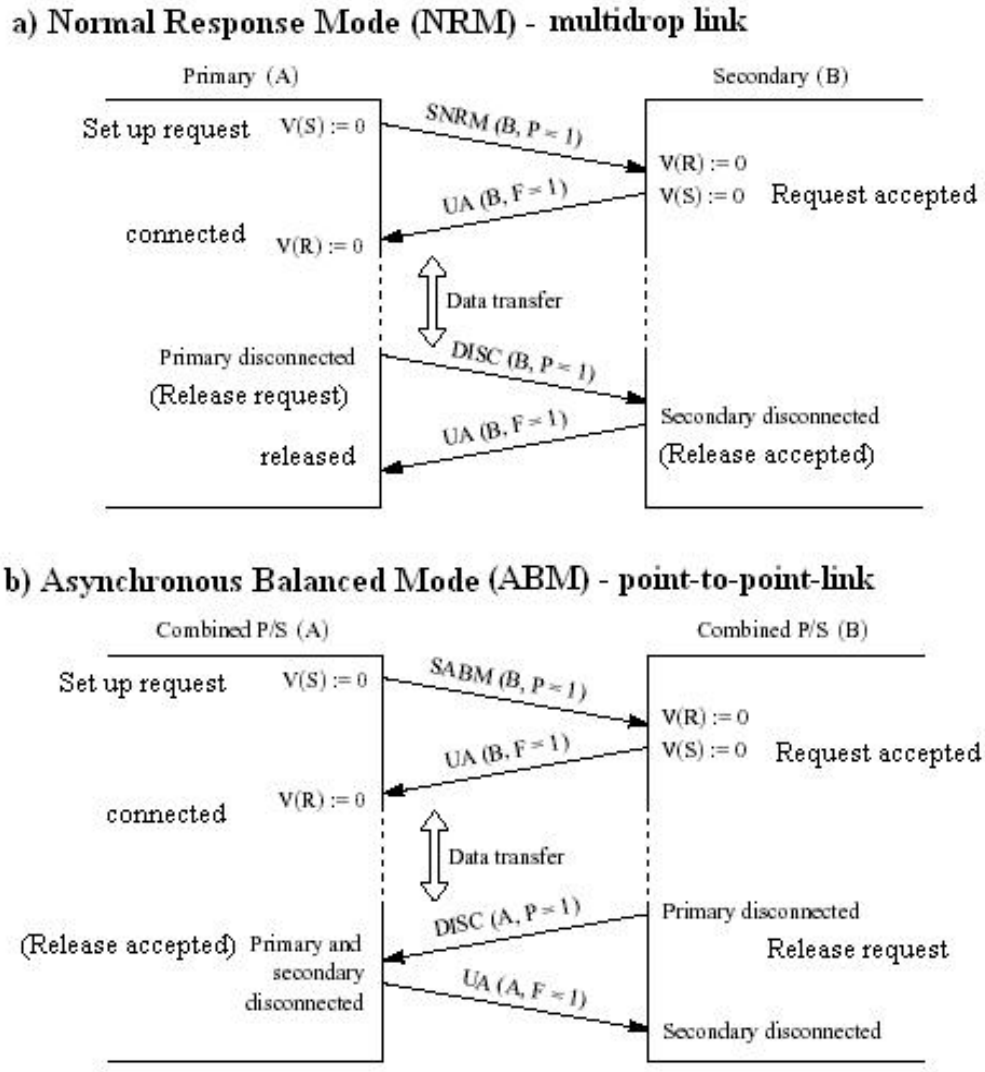
- A: SNRM(B,P=1) (Polling B station)
- B: UA(B,F=1)
- A: DISC(B,P=1)
- B: UA(B,F=1)

b) ABM – point-to-point link, truyền 2 hướng

- A: SABM(B,P=1)
- B: UA(B,F=1)

– B: DISC(A,P=1)

– A: UA(A,F=1)



Hình 2.5. Lưu đồ thời gian thực hiện giao thức HDLC

2.3. TẦNG MẠNG (NETWORK)

2.3.1. Vai trò và chức năng của tầng mạng

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 cấp 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.

Ngoài 2 chức năng quan trọng nói trên, tầng mạng cũng thực hiện một số chức năng khác, đó là: thiết lập, duy trì và giải phóng các liên kết logic (cho tầng mạng), kiểm soát lỗi, kiểm soát luồng dữ liệu, dồn/tách kênh, cắt/hợp dữ liệu,..

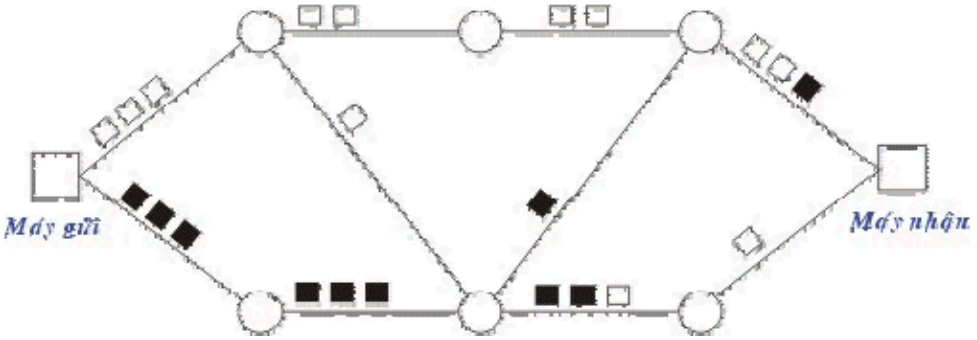
2.3.2. Các kỹ thuật chọn đường trong mạng máy tính

2.3.2.1. Tổng quan

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 2.6. Mô hình chuyển vận các gói tin trong mạng chuyển mạch

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 phân tán* 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.

2.3.2.2. Các giải thuật tìm đường tối ưu

- *Giải thuật Dijkstra cho kỹ thuật chọn đường tập trung.*

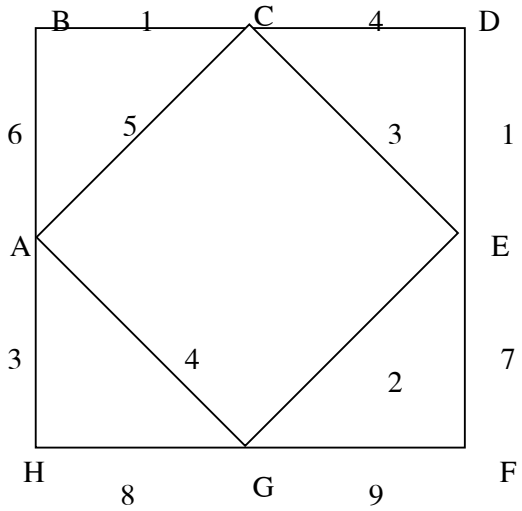
Bài toán đặt ra là: tìm đường đi có "độ dài" (một đại lượng được dùng để làm thước đo, ví dụ độ trễ, cước phí truyền tin) cực tiểu, từ một nút (nguồn) cho trước đến mỗi nuút còn lại của mạng (đích). Ở đây ta coi mạng như là một đồ thị có hướng G(V,E), V là tập đỉnh với n đỉnh tương ứng với n nút mạng, E là tập cung của đồ thị. Ma trận trọng số là a[u,v], u,v ∈ V.

Thuật toán được xây dựng dựa trên cơ sở gán cho các đỉnh các nhãn tạm thời. Nhãn của mỗi đỉnh cho biết cận của độ dài đường đi ngắn nhất từ s đến nó. Các nhãn này sẽ được biến đổi theo một thủ tục lặp, mà ở mỗi bước lặp có một nhãn tạm thời trở thành nhãn cố định. Nếu nhãn của một đỉnh nào đó trở thành một nhãn cố định thì nó sẽ cho ta không phải là cận trên mà là độ dài của đường đi ngắn nhất từ đỉnh s đến nó. Thuật toán được mô tả cụ thể như sau.

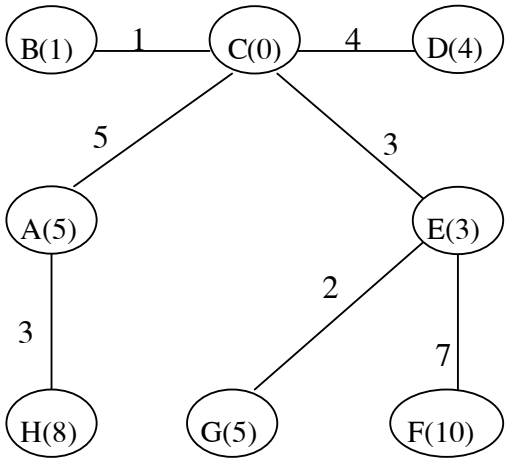
```
Procedure Dijstra;
    (* Đầu vào:
        Đồ thị có hướng G=(V,E) với n đỉnh,
        s ∈ V là đỉnh xuất phát, a[u,v], u,v ∈ V là ma trận trọng số;
    Giả thiết: a[u,v]≥0, u,v ∈ V.
    Đầu ra:
        Khoảng cách từ đỉnh s đến tất cả các đỉnh còn lại d[v], v ∈ V.
        Truoc[v], v ∈ V, ghi nhận đỉnh đi trước v trong đường đi ngắn nhất từ s
        đến v *)
    Begin
        (* Khởi tạo *)
        for v ∈ V do
        begin
            d[v]:=a[s,v];
            Truoc[v]:=s;
        end;
        d[s]:=0; T:=V\ {s} ; (* T là tập các đỉnh cá nhãn tạm thời *)
        (* Bước lặp *)
        while T <> θ do

        begin
            tìm đỉnh u ∈ T thoả mãn d[u]=min {d[z]: z ∈ T} ;
            T:=T\ {u} ; (* Cố định nhãn của đỉnh u*)
            For v ∈ T do
            If d[v]>d[u]+a[u,v] then
            Begin
                d[v]:=d[u]+a[u,v];
                Truoc[v]:=u;
            End;
        end;
End;
```

Hình 2.7. Minh họa thuật toán Dijkstra

Thí dụ. Tìm đường đi ngắn nhất từ C đến các đỉnh còn lại của đồ thị ở hình dưới đây.



Từ đó ta thiết kế được "cây chọn đường" và bảng chọn đường như các hình vẽ sau:

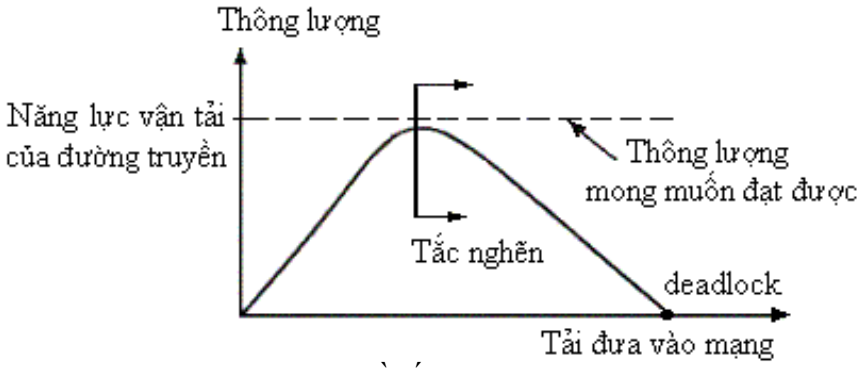


Đích	Nút kế tiếp
A	A
B	B
D	D
E	E
F	E
G	E
H	A

2.3.3. Tắc nghẽn trong mạng

Các khái niệm

- ***Hiện tượng tắc nghẽn (congestion)***: lưu lượng đến mạng tăng lên, thông lượng vận chuyển của mạng lại giảm đi.
- ***Deadlock***: tình trạng tắc nghẽn trầm trọng đến mức mạng bị nghẹt hoàn toàn, thông lượng vận chuyển của mạng tụt xuống bằng không.



Hình 2.8. Lược đồ tắc nghẽn trong mạng

- ***Nguyên nhân dẫn đến tắc nghẽn:***
  - Lưu lượng đi đến trên nhiều lối vào đều cần cùng một đường đi ra.
  - Tốc độ xử lý tại các router chậm
  - Các đường truyền có bandwidth thấp, dẫn đến hiện tượng thắt cổ chai.
- ***Biểu hiện của tắc nghẽn:*** Thời gian khứ hồi (RTT) tăng cao bất thường
- **Các biện pháp khắc phục**
  - Cung cấp đủ bộ đệm ở đầu vào và ra của các đường truyền

- Quản lý bộ đệm hợp lý, có thể loại bỏ sớm (RED)
- Hạn chế lưu lượng đến ngay ở đầu vào của toàn bộ hệ thống
- Điều khiển lưu lượng (thí dụ dùng Sliding Window)

Vấn đề này sẽ được nghiên sâu hơn trong chương 4 và chương 5.

2.3.4. Giao thức X25 PLP

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 có 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.

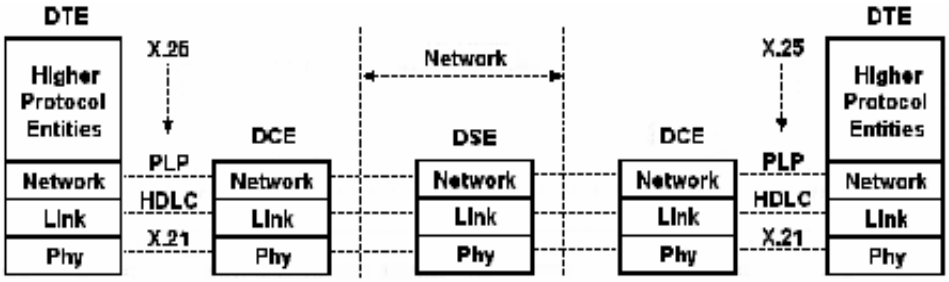
- **Đặc điểm:**
  - Là mạng truyền dữ liệu công cộng đầu tiên.
  - Vận chuyển dữ liệu hướng kết nối
  - Để sử dụng X.25, máy tính đầu tiên phải thiết lập kết nối tới một máy tính ở xa, nghĩa là phải thiết lập một cuộc gọi (telephone call)
  - Kết nối này được gán 1 *connection number* để sử dụng cho các gói (packet) số liệu vận chuyển:
    + Nhiều kết nối có thể được sử dụng đồng thời giữa 2 máy tính.
    + Kết nối trong X.25 là kết nối ảo (Virtual Circuit)
- **Nguyên tắc hoạt động**
  - X.25 là một dịch vụ truyền thông máy tính công cộng, dựa trên hệ thống viễn thông diện rộng (PSTN).
  - X.25 được CCITT và sau này là ITU chuẩn hoá (1976).
  - X.25 chỉ đặc tả giao diện giữa DTE và DCE:
    + DTE (Data Terminal Equipment)- thiết bị đầu cuối dữ liệu

+ DCE (Data Circuit-terminating Equipment) - thiết bị mạch đầu cuối dữ liệu, hay là thiết bị kết nối mạng.

– X.25 không quy định cụ thể kiến trúc và tổ chức hoạt động nội bộ của mạng.

– Tổ chức và thực hiện hệ thống mạng để cung cấp dịch vụ X.25 tại giao diện với NSD là nhiệm vụ của nhà cung cấp dịch vụ X.25 - thường là nhà cung cấp dịch vụ viễn thông công cộng.

- **Các giao thức chuẩn:** X.25 qui định sử dụng các giao thức chuẩn ở các mức như sau:
  - **Mức vật lý:**
    – X.21 cho truyền số liệu số (Digital) giữa DTE và DCE
    – X.21 bis cho truyền số liệu tương tự (Analog) giữa DTE và DCE
  - **Mức liên kết:**
    – LAPB (Link Access Protocol Balanced), là một phần của HDLC, để trao đổi số liệu tin cậy giữa DTE và DCE
  - **Mức mạng:**
    – PLP (Packet Level Protocol): giao thức chuyển mạch gói + hướng kết nối, các subscriber sử dụng để thiết lập VC và truyền thông với nhau.
    – là giao thức được đặc tả mới trong X.25
  - Ba mức trên tương ứng với 3 mức thấp nhất của mô hình ISO/OSI



Hình 2.9. Đặc tả giao diện mạng X25

Các đặc điểm quan trọng nhất của X.25:

– Các gói tin điều khiển cuộc gọi, được dùng để thiết lập và huỷ bỏ các kênh ảo, được gửi trên cùng kênh và mạch ảo như các gói in dữ liệu.

– Việc dồn kênh của các kênh ảo xảy ra ở tầng 3

– Cả tầng 2 và tầng 3 đều áp dụng cơ chế điều khiển lưu lượng và kiểm soát lỗi.

– X.25 được sử dụng rộng rãi trong khoảng 10 năm.

– Khoảng 1980s, X.25 được thay thế bởi một mạng mới – Frame Relay.

2.3.5. Công nghệ chuyển mạch nhanh

2.3.5.1. Mạng chuyển mạch khung – Frame Relay (FR)

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.

Khác nhau căn bản giữa FR và X.25:

- Tín hiệu điều khiển cuộc gọi được vận chuyển trên một kết nối logic riêng; vì vậy, các node trung gian không cần phải duy trì các bảng trạng thái và xử lý các message này cho từng kết nối.
- Multiplexing và switching đối với các kết nối logic được thực hiện ở layer 2 (chứ không phải layer 3), do đó loại bỏ được chi phí xử lý ở 1 layer.
- Điều khiển lưu lượng và kiểm soát lỗi: Không áp dụng các cơ chế điều khiển theo chặng. FR cũng không cung cấp các cơ chế điều khiển End-to-end, nhiệm vụ này các tầng trên phải giải quyết

Ưu điểm của FR với X.25:

- Làm cho quá trình truyền thông hợp lý hơn
- Chức năng giao thức tại giao diện user-network được giảm bớt
- Chi phí xử lý bên trong mạng cũng giảm

  → Lower delay & Higher throughput (cỡ 1 bậc)
- Ứng dụng quan trọng nhất của Frame Relay: kết nối các mạng LAN ở các văn phòng của một công ty.
- Frame Relay đạt được mức độ thành công vừa phải, hiện vẫn đang được sử dụng.

Tóm tắt các đặc trưng công nghệ:

- FR thực hiện các chức năng cơ bản của mức Data link: tạo và xử lý frame, địa chỉ hoá, quản lý các kênh ảo.
- Sử dụng kỹ thuật dồn/tách kênh không đồng bộ ở mức Data link: → Sử dụng hiệu quả hơn đường truyền. Tốc độ trao đổi số liệu: 56 Kbps – 2,048 Mbps.

- Thiết lập và giải phóng kênh theo giao thức báo hiệu chuẩn Q.931 của mạng ISDN.
- Không có chức năng xử lý giao thức ở mức mạng.
- No Link-by-link Flow Control and Error Control; Các ES kiểm tra phát hiện lỗi và khắc phục (end-to-end).
- Hệ chuyển mạch ở giao diện giữa mạng và hệ thống cuối kiểm tra CRC và không forward các frame bị lỗi.
- Giao diện quản trị nội tại LMI (Local Management Interface) của FR hỗ trợ việc quản trị trao đổi số liệu trên các kênh ảo trong mạng.

2.3.5.2. Kỹ thuật ATM

Hiện nay kỹ thuật Cell Relay dựa trên phương thức truyền thông không đồng bộ (ATM) có thể cho phép thông lượng 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 channel) 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, hỗ 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 phần tử chuyển mạch (a matrix of binary switching elements) để vận hành lưu thông. Khả năng mở rộng (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.

2.3.6. Dịch vụ OSI cho tầng mạng

(Tham khảo phần II.3.5, tr 71, 72) của giáo trình [1]).

2.4. TẦNG GIAO VẬN (TRANSPORTATION)

2.4.1. Vai trò và chức năng của tầng Giao vận

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 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.

2.4.2. Giao thức chuẩn cho tầng Giao vận

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.

2.4.3. Dịch vụ OSI cho tầng Giao vận

(Tham khảo phần II.4.3, tr 92-93 của giáo trình [1]).

2.5. TẦNG PHIÊN (SESSION)

2.5.1. Vai trò và chức năng của tầng Phiên

Tầng giao Phiên (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.

2.5.2. Dịch vụ OSI cho tầng Phiên

(Tham khảo phần II.5.2, tr 96-103 của giáo trình [1]).

2.5.3. Giao thức chuẩn cho tầng Phiên

(Tham khảo phần II.5.3, tr 104-106 của giáo trình [1]).

2.6. TẦNG TRÌNH DIỄN (PRESENTATION)

2.6.1. Vai trò và chức năng của tầng Trình diễn

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.

2.6.2. Dịch vụ OSI cho tầng Trình diễn

(Tham khảo phần II.6.2, tr 110-114 của giáo trình [1]).

2.6.3. Giao thức chuẩn cho tầng Trình diễn

(Tham khảo phần II.6.3, tr 115-118 của giáo trình [1]).

2.7. TẦNG ỨNG DỤNG (APPLICATION)

2.7.1. Vai trò và chức năng của tầng Ứng dụng

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.

2.7.2. Chuẩn hoá tầng ứng dụng

(Tham khảo phần II.7.2, tr 121-129 của giáo trình [1]).

2.8. CÂU HỎI VÀ BÀI TẬP

(đang tiếp tục bổ sung)

CHƯƠNG 3. MẠNG CỤC BỘ – MẠNG LAN

3.1. ĐẶC TRƯNG MẠNG CỤC BỘ

Do nhu cầu thực tế của các cơ quan, trường học, doanh nghiệp, tổ chức cần kết nối các máy tính đơn lẻ thành một mạng nội bộ để tạo khả năng trao đổi thông tin, sử dụng chung tài nguyên (phần cứng, phần mềm). Ví dụ trong một văn phòng có một máy in, để tất cả mọi người có thể sử dụng chung máy in đó thì giải pháp nối mạng có thể khắc phục được hạn chế này.

Mục đích của việc sử dụng mạng ngày nay có nhiều thay đổi so với trước kia. Mặc dù mạng máy tính phát sinh từ nhu cầu chia sẻ và dùng chung tài nguyên, nhưng mục đích chủ yếu vẫn là sử dụng chung tài nguyên phần cứng. Ngày nay mục đích chính của mạng là trao đổi thông tin và CSDL dùng chung → công nghệ mạng cục bộ phát triển vô cùng nhanh chóng

Để phân biệt mạng LAN với các loại mạng khác người ta căn cứ theo các đặc trưng sau:

– *Đặc trưng địa lý:* cài đặt trong phạm vi nhỏ (toà nhà, một căn cứ quân sự,..) có đường kính từ vài chục mét đến vài chục km → có ý nghĩa tương đối.

– *Đặc trưng về tốc độ truyền*: cao hơn mạng diện rộng, khoảng 100 Mb/s, có thể đến 1000 Mbps với công nghệ Gigabit.

– *Đặc trưng độ tin cậy*: tỷ suất lỗi thấp, có thể đạt 10^{-8} đến 10^{-11}.

– *Đặc trưng quản lý*: thường là sở hữu riêng của một tổ chức → việc quản lý khai thác tập trung, thống nhất.

Tuy nhiên sự phân biệt mạng LAN theo các đặc trưng trên chỉ mang tính tương đối, cùng với công nghệ ngày càng cao thì ranh giới giữa LAN, MAN, WAN ngày càng mờ đi.

3.2. KIẾN TRÚC MẠNG CỤC BỘ

3.2.1. Topology

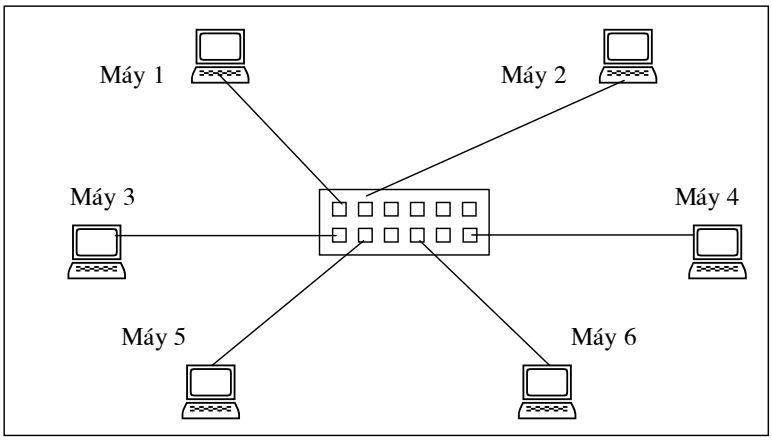
Về nguyên tắc mọi topology của mạng máy tính nói chung đều có thể dùng cho mạng cục bộ. Song do đặc thù của mạng cục bộ nên chỉ có 3 topology thường được sử dụng: hình sao (star), hình vòng (ring), tuyến tính (bus)

3.2.1.1. Hình sao (star)

- Tất cả các trạm được nối vào một thiết bị trung tâm có nhiệm vụ nhận tín hiệu từ các trạm và chuyển đến trạm đích của tín hiệu.
- Thiết bị trung tâm có thể là Hub, Switch, router

Vai trò của thiết bị trung tâm là thực hiện việc "bắt tay" giữa các trạm cần trao đổi thông tin với nhau, thiết lập các liên kết điểm - điểm giữa chúng.

SƠ ĐỒ KIỂU KẾT NỐI HÌNH SAO VỚI HUB Ở TRUNG TÂM



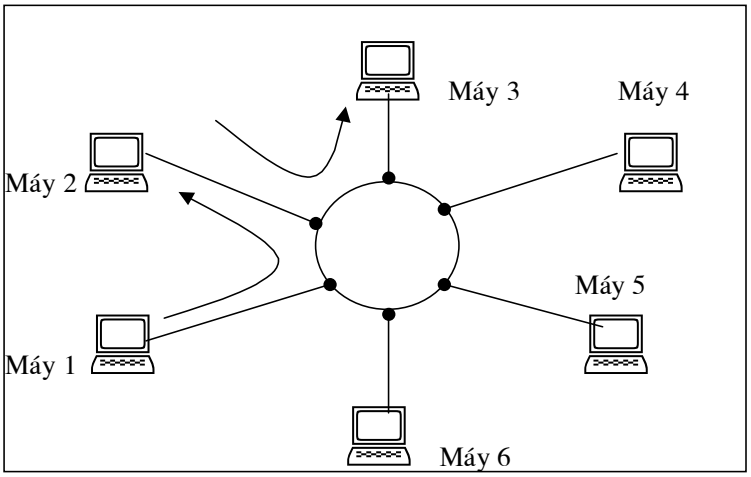
Hình 3.1. Sơ đồ kết nối hình sao

3.2.1.2. Hình vòng (ring)

- Tín hiệu được lưu chuyển theo một chiều duy nhất
- Mỗi trạm làm việc được nối với vòng qua một bộ chuyển tiếp (repeater), có nhiệm vụ nhận tín hiệu rồi chuyển đến trạm kế tiếp trên vòng

Để tăng độ tin cậy của mạng, phải lắp vòng dự phòng, khi đường truyền trên vòng chính bị sự cố thì vòng phụ được sử dụng với chiều đi của tín hiệu ngược với chiều đi của mạng chính.

SƠ ĐỒ KIỂU KẾT NỐI DẠNG VÒNG



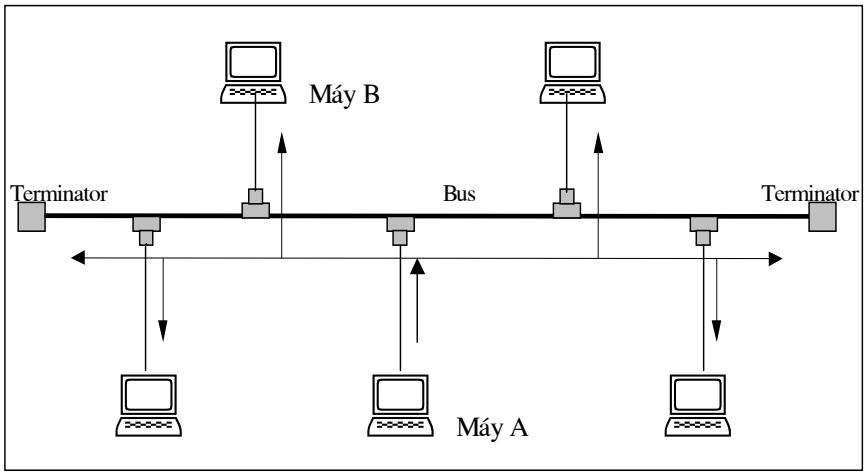
Hình 3.2. Sơ đồ kết nối vòng (ring)

3.2.1.3. Dạng đường thẳng (Bus)

- Tất cả các trạm đều dùng chung một đường truyền chính (Bus) được giới hạn bởi hai đầu nối (terminator).
- Mỗi trạm được nối vào Bus qua một đầu nối chữ T (T-connector).

- Khi một trạm truyền dữ liệu thì tín hiệu được quảng bá trên 2 chiều của Bus (tất cả các trạm khác đều có thể nhận tín hiệu)

SƠ ĐỒ KIỂU KẾT NỐI DẠNG TUYẾN TÍNH (BUS)



Hình 3.3. Sơ đồ kết nối đường thẳng (bus))

**** So sánh giữa các cách kết nối và ưu nhược điểm của chúng:***

- Khác nhau: kiểu hình sao là kết nối điểm - điểm trực tiếp giữa hai máy tính thông qua một thiết bị trung tâm. Kiểu vòng thì tín hiệu lưu chuyển trên vòng là một chuỗi các kết nối điểm - điểm. Kiểu tuyến tính thì dữ liệu truyền dựa trên điểm - nhiều điểm hoặc quảng bá.
- Ưu điểm: Cả ba cách kết nối đều đơn giản, dễ lắp đặt, dễ thay đổi cấu hình

Hình sao:

- Ưu điểm: Dễ kiểm soát. Do sử dụng liên kết điểm - điểm nên tận dụng được tối đa tốc độ của đường truyền vật lý
- 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 tại)

Dạng vòng:

– Nhược điểm: Nếu xảy ra sự cố trên đường truyền, tất cả các máy trong mạng không thể giao tiếp với nhau. Đòi hỏi giao thức truy nhập đường truyền khá phức tạp (Tuy nhiên toàn bộ công việc này được hệ phần mềm giải quyết)

Dạng đường thẳng:

- Nhược điểm: nếu xảy ra sự cố trên đường truyền, toàn bộ các máy trong mạng không thể giao tiếp với nhau được nữa. Giao thức quản lý truy nhập đường truyền phức tạp

Do ưu nhược điểm của từng loại mà trong thực tế người ta thường chọn kiểu kết nối lai - là tổ hợp của các kiểu kết nối trên.

3.3.2. Đường truyền vật lý

Mạng cục bộ thường sử dụng 3 loại đường truyền vật lý và cáp đôi xoắn, cáp đồng trục, và cáp sợi quang. Ngoài ra gần đây người ta cũng đã bắt đầu sử dụng nhiều các mạng cục bộ không dây nhờ radio hoặc viba.

Cáp đồng trục đường sử dụng nhiều trong các mạng dạng tuyến tính, hoạt động truyền dẫn theo dải cơ sở (baseband) hoặc dải rộng (broadband). Với dải cơ sở, toàn bộ khả năng của đường truyền được dành cho một kênh truyền thông duy nhất, trong khi đó với dải rộng thì hai hoặc nhiều kênh truyền thông cùng phân chia dải thông của kênh truyền

Hầu hết các mạng cục bộ đều sử dụng phương thức dải rộng. Với phương thức này tín hiệu có thể truyền đi dưới cả hai dạng: tương tự (analog)và số (digital) không cần điều chế.

Cáp đồng trục có hai loại là cáp gầy (thin cable) và cáp béo (thick cable). Cả hai loại cáp này đều có tốc độ làm việc 10Mb/s nhưng cáp gầy có độ suy hao tín hiệu lớn hơn, có độ dài cáp tối đa cho phép giữa hai repeater nhỏ hơn cáp béo→ Cáp gầy thường dùng để nối các trạm trong cùng một văn phòng, phòng thí nghiệm, còn cáp béo dùng để nối dọc theo hành lang, lên các tầng lầu,..

Phương thức truyền thông theo dải rộng có thể dùng cả cáp đôi xoắn, nhưng cáp đôi xoắn chỉ thích hợp với mạng nhỏ hiệu năng thấp và chi phí đầu tư ít.

Phương thức truyền theo dải rộng chia dải thông (tần số) của đường truyền thành nhiều dải tần con (kênh), mỗi dải tần con đó cung cấp một kênh truyền dữ liệu tách biệt nhờ sử dụng một cặp modem đặc biệt. Phương thức này vốn là một phương tiện truyền một chiều: các tín hiệu đưa vào đường truyền chỉ có thể truyền đi theo một hướng → không cài đặt được các bộ khuyếch đại để chuyển tín hiệu của một tần số theo cả hai chiều. Vì thế xảy ra tình trạng chỉ có trạm nằm dưới trạm truyền là có thể nhận được tín hiệu. Vậy làm thế nào để có hai đường dẫn dữ liệu trên mạng. Điểm gặp nhau của hai đường dẫn đó gọi là điểm đầu cuối. Ví dụ, trong topo dạng bus thì điểm đầu cuối đơn giản chính là đầu mút của bus (terminator), còn với topo dạng cây (tree) thì chính là gốc của cây (root). Các trạm khi truyền đều truyền về hướng điểm đầu cuối (gọi là đường dẫn về), sau đó các tín hiệu nhận được ở điểm đầu cuối sẽ truyền theo đường dẫn thứ hai xuất phát từ điểm đầu cuối (gọi là đường dẫn đi). Tất cả các trạm đều nhận dữ liệu trên đường dẫn đi. Để cài đặt đường dẫn về và đi, có thể sử dụng cấu hình vật lý như trên hình 3.4.

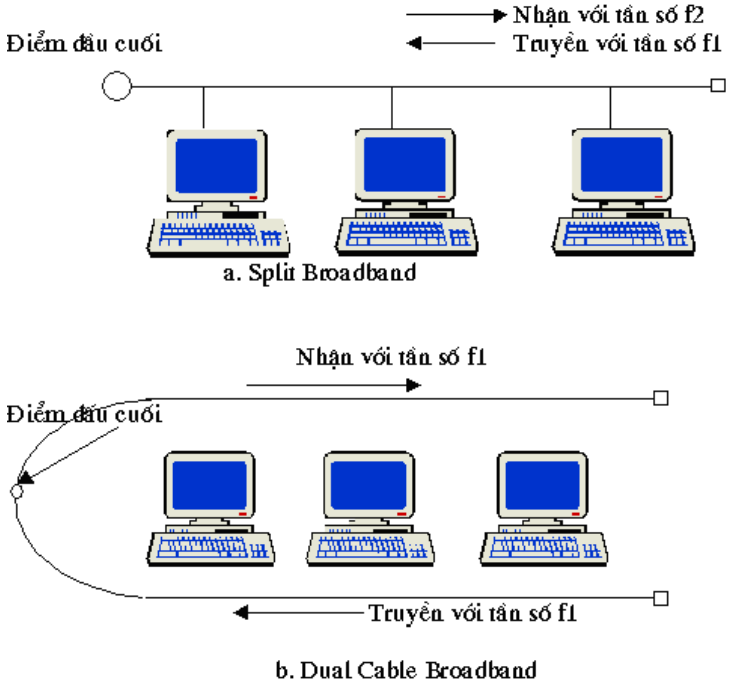
Trong cấu hình cáp đôi (dual cable), các đường dẫn về và đi chạy trên các cáp riêng biệt và điểm đầu cuối đơn giản chỉ là một đầu nối thụ động của chúng. Trạm gửi và nhận cùng một tần số

Trong cấu hình tách (split), cả hai đường dẫn đều ở trên cùng một cáp nhưng tần số khác nhau: đường dẫn về có tần số thấp và đường dẫn đi có tần số cao hơn. Điểm đầu cuối là bộ chuyển đổi tần số.

- Chú ý: việc lựa chọn đường truyền và thiết kế sơ đồ đi cáp (trong trường hợp hữu tuyến) là một trong những công việc quan trọng nhất khi thiết kế và cài đặt một mạng máy tính nói chung và mạng cục bộ nói riêng. Giải pháp lựa chọn pháp đáp ứng được nhu cầu sử dụng mạng thực tế không chỉ cho hiện tại mà cho cả tương lai.

- VD: muốn truyền dữ liệu đa phương tiện thì không thể chọn loại cáp chỉ cho phép thông lượng tối đa là vài Mb/s , mà phải nghĩ đến loại cáp cho phép thông lượng

trên 100 Mb/s. Việc lắp đặt hệ thống trong cáp trong nhiều trường hợp (toà nhà nhiều tầng) là tốn rất nhiều công của → phải lựa chọn cẩn thận, không thể để xảy ra trường hợp sau 1 -2 năm gỡ bỏ, lắp hệ thống mới.



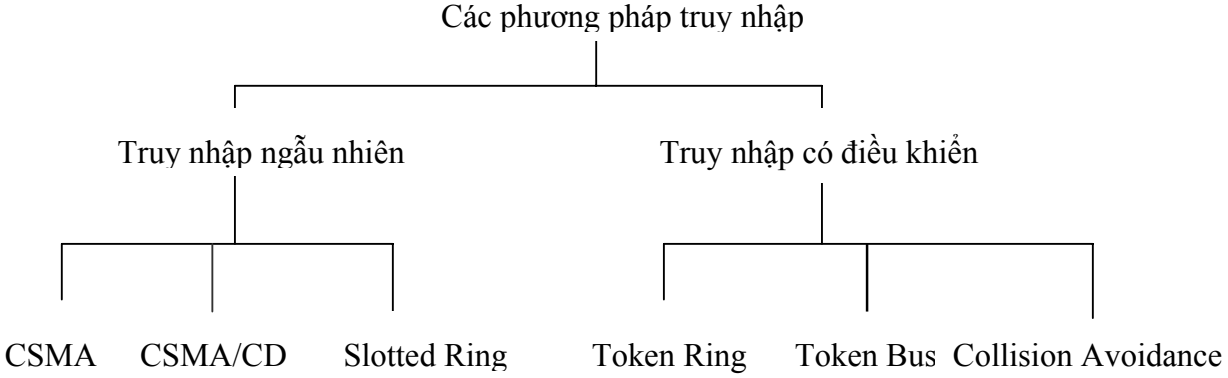
Hình 3.4. Cấu hình vật lý cho Broadband

3.3. CÁC PHƯƠNG PHÁP TRUY NHẬP ĐƯỜNG TRUYỀN VẬT LÝ

3.3.1. Giới thiệu

Đối với topo dạng hình sao, khi một liên kết được thiết lập giữa hai trạm thì thiết bị trung tâm sẽ đảm bảo đường truyền được dành riêng trong suốt cuộc truyền. Tuy nhiên đối với topo dạng vòng và tuyến tính thì chỉ có một đường truyền duy nhất nối tất cả các trạm với nhau bởi vậy cần phải có một quy tắc chung cho tất cả các trạm nối vào mạng để bảo đảm rằng đường truyền được truy nhập và sử dụng một cách tốt đẹp

Có nhiều phương pháp khác nhau để truy nhập đường truyền vật lý, được phân làm hai loại: phương pháp truy nhập ngẫu nhiên, và phương pháp truy nhập có điều kiện.



Trong đó có 3 phương pháp hay dùng nhất trong các mạng cục bộ hiện nay: phương pháp CSMA/CD, Token Bus, Token Ring

3.3.2. Phương pháp CSMA/CD

Phương pháp đa truy nhập sử dụng sóng mang có phát hiện xung đột - CSMA/CD (Carrier Sense Multiple Access with Collision Detection). Phương pháp này sử dụng cho topo dạng bus, trong đó tất cả các trạm của mạng đều 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 trên mạng theo một khuôn dạng đã định sẵn trong đó có một vùng thông tin điều khiển chứa địa chỉ trạm đích.

Phương pháp 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: một trạm cần truyền dữ liệu trước hết phải "nghe" xem đường truyền đang rỗi hay bận. Nếu rỗi thì truyền dữ liệu đi theo khuôn dạng đã quy định trước. Ngược lai, nếu bận (tức là đã có dữ liệu khác) thì trạm phải thực hiện một trong 3 giải thuật sau (gọi là giải thuật "kiên nhẫn")

1. Tạm "rút lui" chờ đợi trong một thời gian ngẫu nhiên nào đó rồi lại bắt đầu nghe đường truyền (Non- persistent)
2. Tiếp tục "nghe" đến khi đường truyền rỗi thì truyền dữ liệu đi với xác suất bằng 1 (1- persistent)
3. Tiếp tục "nghe" đến khi đường truyền rỗi thì truyền đi với xác suất p xác định trước ($0 < p < 1$) (p- persistent)

Với giải thuật 1 có hiệu quả trong việc tránh xung đột vì hai trạm cần truyền khi thấy đường truyền bận sẽ cùng "rút lui" chờ đợi trong các thời đoạn ngẫu nhiên khác.→ Nhược điểm có thể có thời gian chết sau mỗi cuộc truyền

Giải thuật 2 khắc phục nhược điểm có thời gian chết bằng cách cho phép một trạm có thể truyền ngay sau khi một cuộc truyền kết thúc. → Nhược điểm: Nếu lúc đó có hơn một trạm đang đợi thì khả năng xảy ra xung đột là rất cao

Giải thuật 3 trung hoà giữa hai giải thuật trên. Với giá trị p lựa chọn hợp lý có thể tối thiểu hoá được cả khả năng xung đột lẫn thời gian chết của đường truyền. Xảy ra xung đột là do độ trễ của đường truyền dẫn: một trạm truyền dữ liệu đi rồi nhưng do độ trễ đường truyền nên một trạm khác lúc đó đang nghe đường truyền sẽ tưởng là rỗi và cứ thể truyền dữ liệu đi → xung đột. Nguyên nhân xảy ra xung đột của phương pháp này là các trạm chỉ "nghe trước khi nói" mà không "nghe trong khi nói" do vậy trong thực tế có xảy ra xung đột mà không biết, vẫn cứ tiếp tục truyền dữ liệu đi → gây ra chiếm dụng đường truyền một cách vô ích

Để có thể phát hiện xung đột, cải tiến thành phương pháp CSMA/CD (LWT - Listen While Talk - nghe trong khi nói) tức là bổ xung thêm các quy 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 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.

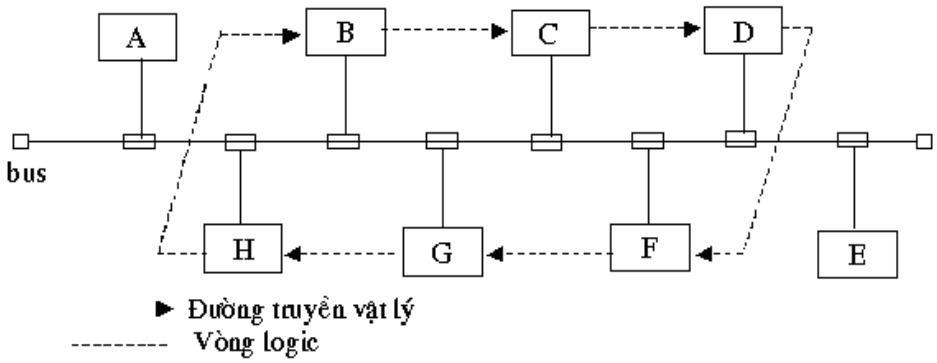
♦ Sau đó trạm chờ đợi một thời gian ngẫu nhiên nào đó rồi thử truyền lại theo các quy tắc của CSMA

→ Rõ ràng với CSMA/CD thời gian chiếm dụng đường truyền vô ích giảm xuống bằng thời gian để phát hiện xung đột. CSMA/CD cũng sử dụng một trong 3 giải thuật "kiên nhẫn" ở trên, trong đó giải thuật 2 được ưa dùng hơn cả.

3.3.3. Phương pháp Token Bus

Phương pháp truy nhập có điều khiển dùng kỹ thuật "chuyển thẻ bài" để cấp phát quyền truy nhập đường truyền. Thẻ bài (Token) là một đơn vị dữ liệu đặc biệt, có kích thước và có chứa các thông tin điều khiển trong các khuôn dạng

Nguyên lý: 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 nhận được thẻ bài thì nó có quyền sử dụng đường truyền trong một thời gian định trước. Trong thời gian đó nó có thể truyền một hoặc nhiều đơn vị dữ liệu. Khi đã hết dữ liệu hay hết thời đoạn cho phép, trạm phải chuyển thẻ bài đến trạm tiếp theo trong vòng logic. Như vậy công việc phải làm 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 được biết địa chỉ của các trạm kề trước và 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 thì không được đưa vào vòng logic và chúng chỉ có thể tiếp nhận dữ liệu.



Hình 3.5. Token Bus

Trong hình vẽ, các trạm A, E nằm ngoài vòng logic, chỉ có thể tiếp nhận dữ liệu dành cho chúng.

Vấn đề quan trọng là phải duy trì được vòng logic tuỳ theo trạng thái thực tế của mạng tại thời điểm nào đó. Cụ thể cần phải thực hiện các chức năng sau:

✧ Bổ sung một trạm vào vòng logic: các trạm nằm ngoài vòng logic cần được xem xét định kỳ để nếu có nhu cầu truyền dữ liệu thì bổ sung vào vòng logic.

✧ Loại bỏ một trạm khỏi vòng logic: Khi một trạm không còn nhu cầu truyền dữ liệu cần loại nó ra khỏi vòng logic để tối ưu hoá việc điều khiển truy nhập bằng thẻ bài

✧ Quản lý lỗi: một số lỗi có thể xảy ra, chẳng hạn trùng địa chỉ (hai trạm đều nghĩ rằng đến lượt mình) hoặc "đứt vòng" (không trạm nào nghĩ đến lượt mình)

✧ Khởi tạo vòng logic: Khi cài đặt mạng hoặc sau khi "đứt vòng", cần phải khởi tạo lại vòng.

- Các giải thuật cho các chức năng trên có thể làm như sau:

✧ Bổ sung một trạm vào vòng logic, mỗi trạm trong vòng có trách nhiệm định kỳ tạo cơ hội cho các trạm mới nhập vào vòng. Khi chuyển thẻ bài đi, trạm sẽ gửi thông báo "tìm trạm đứng sau" để mời các trạm (có địa chỉ giữa nó và trạm kế tiếp nếu có) gửi yêu cầu nhập vòng. Nếu sau một thời gian xác định trước mà không có yêu cầu nào thì trạm sẽ chuyển thẻ bài tới trạm kề sau nó như thường lệ. Nếu có yêu cầu thì trạm gửi thẻ bài sẽ ghi nhận trạm yêu cầu trở thành trạm đứng kề sau nó và chuyển thẻ bài tới trạm mới này. Nếu có hơn một trạm yêu cầu nhập vòng thì trạm giữ thẻ bài sẽ phải lựa chọn theo giải thuật nào đó.

✧ Loại một trạm khỏi vòng logic: Một trạm muốn ra khỏi vòng logic sẽ đợi đến khi nhận được thẻ bài sẽ gửi thông báo "nối trạm đứng sau" tới trạm kề trước nó yêu cầu trạm này nối trực tiếp với trạm kề sau nó

✧ Quản lý lỗi: Để giải quyết các tình huống bất ngờ. Chẳng hạn, trạm đó nhận được tín hiệu cho thấy đã có các trạm khác có thẻ bài. Lập tức nó phải chuyển sang trạng thái nghe (bị động, chờ dữ liệu hoặc thẻ bài). Hoặc sau khi kết thúc truyền dữ liệu, trạm phải chuyển thẻ bài tới trạm kề sau nó và tiếp tục nghe xem trạm kề sau đó có hoạt động hay đã bị hư hỏng. Nếu trạm kề sau bị hỏng thì phải tìm cách gửi các thông báo để vượt qua trạm hỏng đó, tìm trạm hoạt động để gửi thẻ bài.

✧ Khởi tạo vòng logic: Khi một trạm hay nhiều trạm phát hiện thấy đường truyền không hoạt động trong một khoảng thời gian vượt quá một giá trị ngưỡng (time out) cho trước - thẻ bài bị mất (có thể do mạng bị mất nguồn hoặc trạm giữ thẻ bài bị hỏng). Lúc đó trạm phát hiện sẽ gửi đi thông báo "yêu cầu thẻ bài" tới một trạm được chỉ định trước có trách nhiệm sinh thẻ bài mới và chuyển đi theo vòng logic.

3.3.4. Phương pháp Token Ring

Phương pháp này dựa trên nguyên lý dùng thẻ bài để cấp phát quyền truy nhập đường truyền. Thẻ bài lưu chuyển theo vòng vật lý chứ không cần thiết lập vòng logic như phương pháp trên

Thẻ bài là một đơn vị dữ liệu đặc biệt trong đó có một bít 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. Khi đó nó sẽ đổi bít trạng thái thành bận 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 buộc 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 xoá bỏ dữ liệu, đổi bít trạng thái thành rỗi cho lưu chuyển tiếp trên vòng để các trạm khác có thể nhận được quyền truyền dữ liệu.

Sự quay về trạm nguồn của dữ liệu và thẻ bài nhằm tạo một cơ chế nhận từ nhiên: trạm đích có thể gửi vào đơn vị dữ liệu các thông tin về kết quả tiếp nhận dữ liệu của mình.

+ Trạm đích không tồn tại hoặc không hoạt động

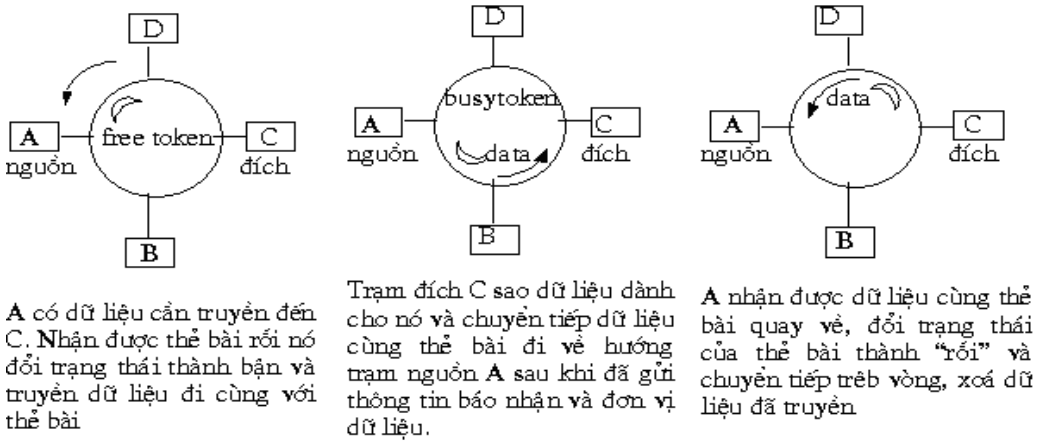
+ Trạm đích tồn tại nhưng dữ liệu không sao chép được

+ Dữ liệu đã được tiếp nhận

- Phương pháp này cần phải giải quyết hai vấn đề có thể gây phá vỡ hệ thống:

+ Mất thẻ bài: trên vòng không còn thẻ bài lưu chuyển nữa

+ Một thẻ bài bận lưu chuyển không dừng trên vòng



Hình 3.6. Sơ đồ hoạt động của phương pháp Token Ring

Giải quyết:

Đối với vấn đề mất thẻ bài, có thể quy định trước một trạm điều khiển chủ động. 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.

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 thẻ bài bận đi qua nó. Nếu nó gặp lại một thẻ bài bận với bít 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 thành rỗi và chuyển tiếp trên vòng. Các trạm còn lại trên trạm 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. Cần có một giải thuật để chọn trạm thay thế cho trạm monitor hỏng.

3.3.5. So sánh các phương pháp

- Độ phức tạp của phương pháp dùng thẻ bài đều lớn hơn nhiều so với CSMA/CD.
- Những công việc mà một trạm phải làm trong phương pháp CSMA/CD đơn giản hơn nhiều so với hai phương pháp dùng thẻ bài.

- Hiệu quả của phương pháp dùng thẻ bài không cao trong điều kiện tải nhẹ: một trạm phải đợi khá lâu mới đến lượt
- Tuy nhiên phương pháp dùng thẻ bài cùng có những ưu điểm: Khả năng điều hoà lưu thông trong mạng, hoặc bằng cách cho phép các trạm truyền số lượng đơn vị dữ liệu khác nhau khi nhận được thẻ bài, hoặc bằng cách lập chế độ ưu tiên cấp phát thẻ bài cho các trạm cho trước. Đặc biệt phương pháp dùng thẻ bài có hiệu quả cao hơn CSMA/CD trong trường hợp tải nặng.

3.4. PHẦN CỨNG VÀ CÁC THIẾT BỊ MẠNG

3.4.1. Thiết bị cấu thành mạng máy tính

Máy chủ (file server - FS), các trạm làm việc (Workstation - WS), các thiết bị ngoại vi dùng chung (máy in, ổ đĩa cứng,...), card mạng, các đầu nối, đường truyền, và một số thiết bị khác như HUB, Switch.

a. Máy chủ

- Hoạt động như một máy chính của mạng, quản lý các hoạt động của mạng (như phân chia tài nguyên chung, trao đổi thông tin giữa các trạm,..). Thông thường máy chủ còn đặt cơ sở dữ liệu dùng chung. Thường thì máy chủ có cấu hình mạnh.
- Trong dạng mạng ngang quyền (Peer to Peer) thì không có máy chủ.

b. Các trạm làm việc

- Là các máy tính cá nhân kết nối với nhau và nối với máy chủ.
- Các máy trạm có thể sử dụng tài nguyên chung của toàn bộ hệ thống mạng.

c. Card mạng (NIC)

- Là thiết bị để điều khiển việc truyền thông và chuyển đổi dữ liệu sang dạng tín hiệu điện hay quang.
- Gồm các bộ điều khiển và thu phát thông tin.
  + Bộ điều khiển thực hiện các chức năng điều khiển truyền thông, đảm bảo dữ liệu được truyền chính xác tới các nút mạng.
  + Bộ thu phát thông tin làm nhiệm vụ chuyển dữ liệu sang dạng tín hiệu điện hay quang và ngược lại.
- Được lắp vào khe cắm của mỗi máy tính của mạng.
- Tuỳ theo yêu cầu sử dụng lựa chọn card mạng cho phù hợp với máy tính, đường truyền dẫn, nhu cầu phát triển trong tương lai.

d. Đường truyền

- Là môi trường truyền dẫn, liên kết các nút mạng, truyền dẫn các tín hiệu điện hay quang.

- Mạng cục bộ sử dụng chủ yếu là các loại cáp, trong đó có hai loại cáp thường được sử dụng: cáp đồng trục, cáp đôi dây xoắn.

3.4.2. Các thiết bị ghép nối mạng

a. Bộ khuyếch đại tín hiệu - Repeater

- Làm việc với tầng thứ nhất của mô hình OSI - tầng vật lý.
- Repeater có hai cổng. Nó thực hiện việc chuyển tiếp tất cả các tín hiệu vật lý đến từ cổng này ra cổng khác sau khi đã khuyếch đại → tất cả các Lan liên kết với nhau qua repeater trở thành một LAN.
- Nó chỉ có khả năng liên kết các LAN có cùng một chuẩn công nghệ.

b. Bộ tập trung - HUB

- Là tên gọi của Repeater nhiều cổng. Nó thực hiện việc chuyển tiếp tất cả các tín hiệu vật lý đến từ một cổng tới tất cả các cổng còn lại sau khi đã khuyếch đại
- Tất cả các LAN liên kết với nhau qua HUB sẽ trở thành một LAN
- HUB không có khả năng liên kết các LAN khác nhau về giao thức truyền thông ở tầng liên kết dữ liệu.

c. Cầu nối - Bridge

- Làm việc với tầng thứ hai của mô hình OSI: tầng liên kết dữ liệu.
- Nó được thiết kế để có khả năng nhận tín hiệu vật lý, chuyển đổi về dạng dữ liệu và chuyển tiếp dữ liệu.
- Bridge có hai cổng: sau khi nhận tín hiệu vật lý và chuyển đổi về dạng dữ liệu từ một cổng, bridge kiểm tra địa chỉ đích, nếu địa chỉ này là của một node liên kết với chính cổng nhận tín hiệu, nó bỏ qua việc xử lý. Trong trường hợp ngược lại dữ liệu được chuyển tới cổng còn lại, tại cổng này dữ liệu được chuyển đổi thành tín hiệu vật lý và gửi đi. Để kiểm tra một node được liên kết với cổng nào của nó, bridge dùng một bảng địa chỉ cập nhật động → tốc độ đường truyền chậm hơn so với repeater.
- Dùng để liên kết các LAN có cung giao thức tầng liên kết dữ liệu, có thể khác nhau về môi trường truyền dẫn vật lý. Không hạn chế về số lượng bridge sử dụng. Cũng có thể được dùng để chia một LAN thành nhiều LAN con → giảm dung lượng thông tin truyền trên toàn LAN.

d. Bộ chuyển mạch - Switch

- Làm việc như một Bridge nhiều cổng. Khác với HUB nhận tín hiệu từ một cổng rồi chuyển tiếp tới tất cả các cổng còn lại, switch nhận tín hiệu vật lý, chuyển đổi thành dữ liệu, từ một cổng, kiểm tra địa chỉ đích rồi gửi tới một cổng tương ứng.
- Nhiều node mạng có thể gửi thông tin đến cùng một node khác tại cùng một thời điểm → mở rộng dải thông của LAN. Switch được thiết kế để liên kết các cổng của nó với dải thông rất lớn (vài trăm Mbps đến hàng Gbps)

- Dùng để vượt qua hạn chế về bán kính hoạt động của mạng gây ra bởi số lượng repeater được phép sử dụng giữa hai node bất kỳ của một LAN
- Là thiết bị lý tưởng dùng để chia LAN thành nhiều LAN "con" làm giảm dung lượng thông tin truyền trên toàn LAN
- Hỗ trợ công nghệ Full Duplex dùng để mở rộng băng thông của đường truyền mà không có repeater hoặc Hub nào dùng được
- Hỗ trợ mạng đa dịch vụ (âm thanh, video, dữ liệu)

d. Bộ dẫn đường -Router

- Làm việc trên tầng network của mô hình OSI.
- Thường có nhiều hơn 2 cổng. Nó tiếp nhận tín hiệu vật lý từ một cổng, chuyển đổi về dạng dữ liệu, kiểm tra địa chỉ mạng rồi chuyển dữ liệu đến cổng tương ứng.
- Dùng để liên kết các LAN có thể khác nhau về chuẩn LAN nhưng cùng giao thức mạng ở tầng network.
- Có thể liên kết hai mạng ở rất xa nhau

e. Cổng giao tiếp - Gateway

- Là thiết bị mạng hoạt động ở tầng trên cùng của mô hình OSI.
- Dùng để liên kết các mạng có kiến trúc hoàn toàn khác nhau
- Có thể hiểu và chuyển đổi giao thức ở tầng bất kỳ của mô hình OSI

3.5. CÁC CHUẨN LAN

Các chuẩn LAN là các chuẩn công nghệ cho LAN được phê chuẩn bởi các tổ chức chuẩn hoá quốc tế, nhằm hướng dẫn các nhà sản xuất thiết bị mạng đi đến sự thống nhất khả năng sử dụng chung các sản phẩm của họ vì lợi ích của người sử dụng và tạo điều kiện cho các nghiên cứu phát triển.

3.5.1. Chuẩn Ethernet

Các chuẩn Ethernet LAN hiện đạng sử dụng phổ biến nhất, đến mức đôi khi hiểu đồng nghĩa với LAN. Sự phát triển của nó trải qua các giai đoạn với tên gọi là DIX standard Ethernet và IEEE802.3 standard.

Năm 1972 công ty Xerox triển khai nghiên cứu về chuẩn LAN. 1980 chuẩn này được 3 công ty DEC (Digital), Intel, Xerox chấp nhận phát triển và gọi là chuẩn DIX Ethernet. Nó đảm bảo tốc độ truyền thông 10 Mpbs, dùng môi trường truyền dẫn là cáp đồng trucj béo, cơ chế truyền tin CSMA/CD.

IEEE (Institute of Electrical and Electrionics Engineers) - một tổ chức chuẩn hoá của Mỹ đưa ra chuẩn IEEE802.3 về giao thức LAN dựa trên DIX Ethernet với các môi trường truyền dẫn khác nhau, gọi là IEEE802.3 10BASE-5, IEEE802.3 10BASE-2 và IEEE802.3 10BASE-T. Đảm bảo tốc độ truyền thông 10Mbps.

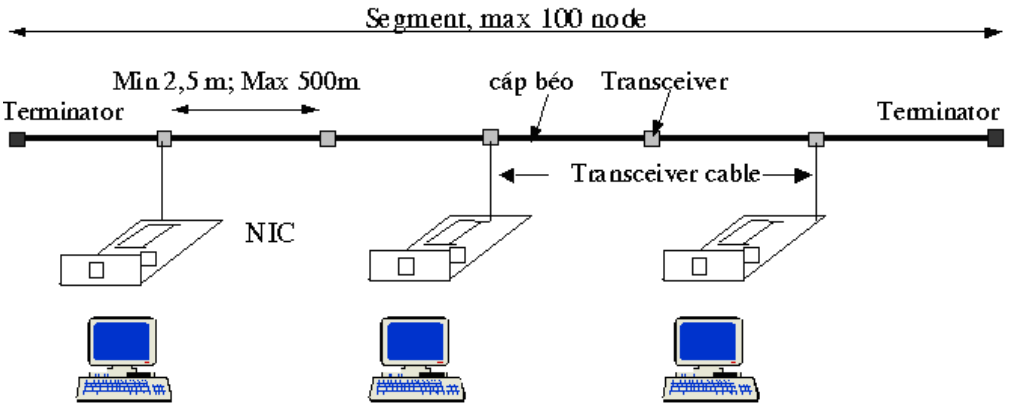
3.5.1.1. 10BASE-5

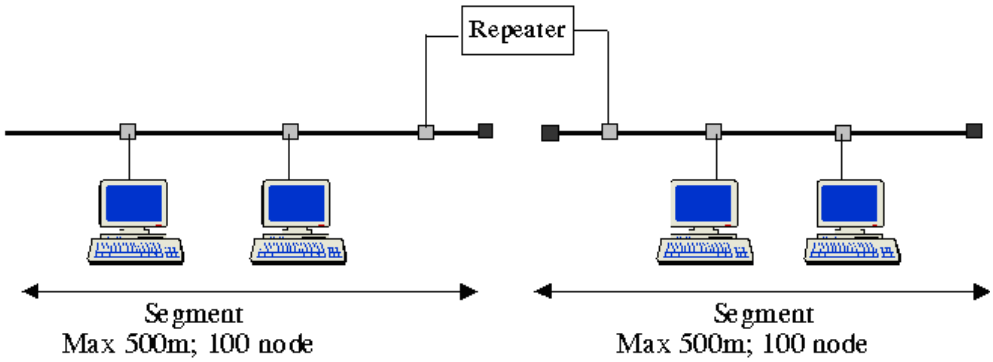
♦ Mô hình phần cứng của mạng

- Topo dạng BUS
- Dùng cáp đồng trục béo 50 Ω còn gọi là cáp vàng, AUI connector (Attachement Unit Interface)
- Hai đầu cáp có hai Terminator 50 Ω, chống phản hồi sóng mang tín hiệu. Dữ liệu truyền thông sẽ không được đảm bảo đúng đắn nếu một trong hai Terminator này bị thiếu hoặc bị lỗi.
- Trên mỗi đoạn cáp có thể liên kết tối đa 100 AUI Transceiver Connector "cái". Khoảng cách tối đa giữa hai AUI là 2,5 m, khoảng cách tối đa là 500m → trên cáp có đánh các dấu hiệu theo từng đoạn bội số của 2,5m và để đảm bảo truyền thông người ta thường chọn khoảng cách tối thiệu giữa hai AUI là 5 m.
- Việc liên kết các máy tính vào mạng được thực hiện bởi các đoạn cáp nối từ các AUI connector đến NIC trong máy tính, gọi là cáp AUI. Hai đầu cáp AUI liên kết với hai AUI connector "đực". Chiều dài tối đa của một cáp AUI là 50 m.
- Số 5 trong tên gọi 10BASE-5 là bắt nguồn từ điều kiện khoảng cách tối đa giữa hai AUI trên cáp là 500 m.

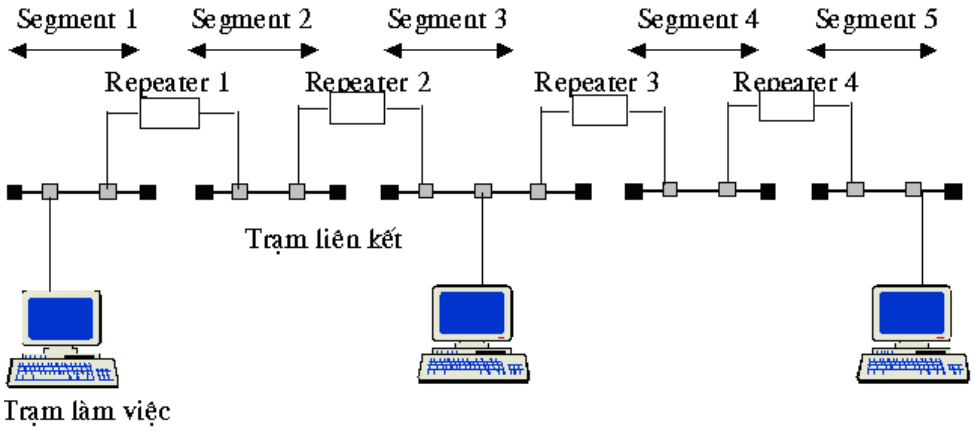
♦ Quy tắc 5- 4-3

- Repeater: Như đã trình bày ở trên, trong mỗi đoạn mạng dùng cáp đồng trục béo không được có quá 100 AUI, khoảng cách tối đa giữa hai AUI không được vượt quá 500m. Trong trường hợp muốn mở rộng mạng với nhau bằng một thiết bị chuyển tiếp tín hiệu gọi là Repeater. Repeater có hai cổng, tín hiệu được nhận vào ở cổng này thì sẽ được phát tiếp ở ra sau cổng kia sau khi đã được khuyếch đại. Tuy nhiên có những hạn chế bắt buộc về số lượng các đoạn mạng và nút mạng có thể có trên một Ethernet LAN
- Quy tắc 5-4-3 là quy tắc tiêu chuẩn của Ethernet được áp dụng trong trường hợp muốn mở rộng mạng, nghĩa là muốn xây dựng một LAN có bán kính hoạt động rộng hoặc có nhiều trạm làm việc vượt quá những hạn chế trên một đoạn cáp mạng (segment).
- Quy tắc 5-4-3 được áp dụng cho chuẩn 10BASE-5 dùng repeater như sau:
  + Không được có quá 5 đoạn mạng
  + Không được có quá 4 repeater giữa hai trạm làm việc bất kỳ
  + Không được có quá 3 đoạn mạng có trạm làm việc. Các đoạn mạng không có trạm làm việc gọi là các đoạn liên kết.





Hình 3.8. Mở rộng chuẩn 10 Base 5 bằng repeater



Hình 3.9. Luật 5-4-3

3.5.1.2. 10BASE-2

♦ Mô hình phần cứng

- Topo dạng BUS
- Dùng cáp đồng trục mỏng 50 Ω, đường kính xấp xỉ 5mm, T-connector, BNC connector
- Hai đầu cáp có hai Terminator 50 Ω, chống phản hồi sóng mang dữ liệu. Dữ liệu truyền thông sẽ không được đảm bảo đúng đắn nếu một trong hai Terminator này bị thiếu hoặc bị lỗi.
- Trên mỗi đoạn cáp có thể liên kết tối đa 30 trạm làm việc. Khoảng cách tối thiểu giữa hai trạm là 0.5 m. Khoảng cách tối đa giữa hai trạm là 185m. Để bảo đảm chất lượng truyền thông người ta thường chọn khoảng cách tối thiểu giữa hai trạm là 5 m.
- Việc liên kết các máy tính vào mạng được thực hiện bởi các T - connector và BNC connector.

- Số 2 trong tên gọi 10BASE-2 là bắt nguồn từ điều kiện khoảng cách tối đa giữa hai trạm trên đoạn cáp là 185m ≈ 200m

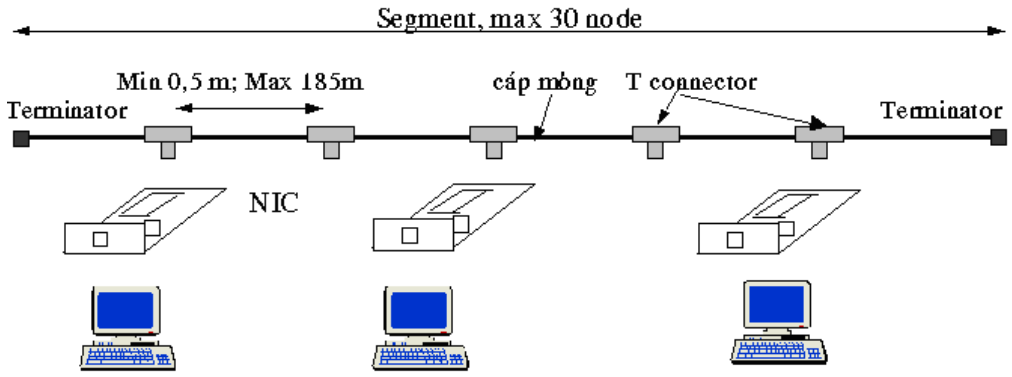
♦ Quy tắc 5 - 4 -3

- Quy tắc 5-4-3 được áp dụng cho chuẩn 10BASE-2 dùng repeater cũng tương tự như đối với trường hợp cho chuẩn 10BASE-5

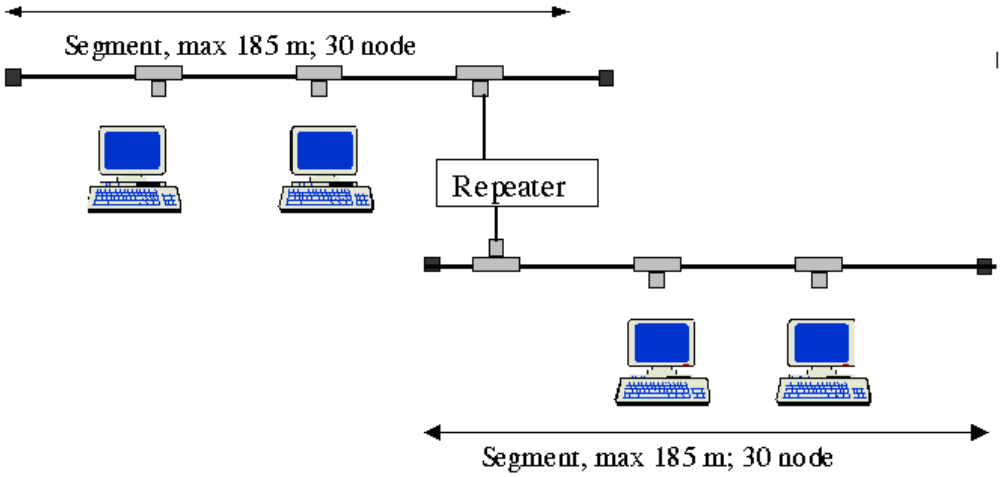
+ Không được có quá 5 đoạn mạng

+ Không được có quá 4 repeater giữa hai trạm làm việc bất kỳ

+ Không được có quá 3 đoạn mạng có trạm làm việc. Các đoạn mạng không có trạm làm việc gọi là các đoạn liên kết.



Hình 3.10. Chuẩn 10 Base-2



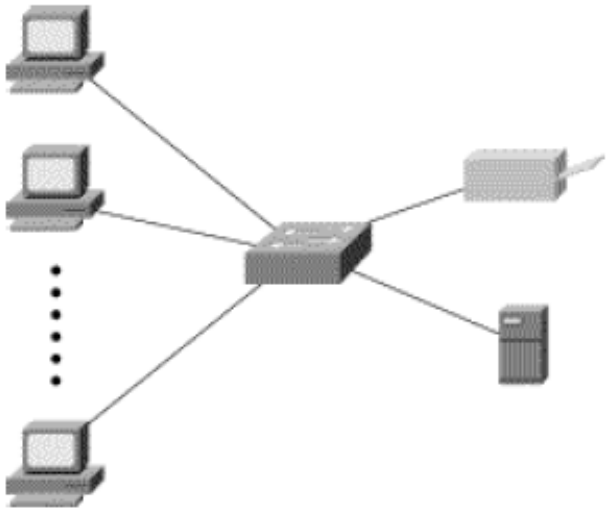
Hình 3.11. Mở rộng chuẩn 10 Base-5 bằng repeater

3.5.1.3. 10BASE-T

♦ **Mô hình phần cứng của mạng**

- Dùng cáp đôi xoắn UTP, RJ 45 connector, và một thiết bị ghép nối trung tâm gọi là HUB

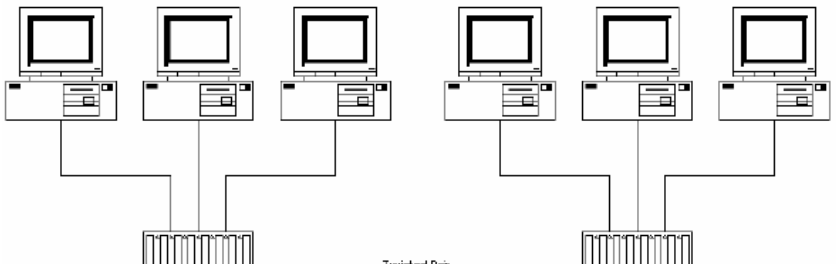
- Mỗi HUB có thể nối từ 4 tới 24 cổng RJ45, các trạm làm việc được kết nối từ NIC tới cổng HUB bằng cáp UTP với hai đầu RJ45. Khoảng cách tối đa từ HUB đến NIC là 100m
- Về mặt vật lý (hình thức) topo của mạng có dạng hình sao
- Tuy nhiên về bản chất HUB là một loại Repeater nhiều cổng vì vậy về mặt logic, mạng theo chuẩn 10BASE-T vẫn là mạng dạng BUS
- Chữ T trong tên gọi 10BASE-T bắt nguồn từ chữ Twisted pair cable (cáp đôi dây xoắn)

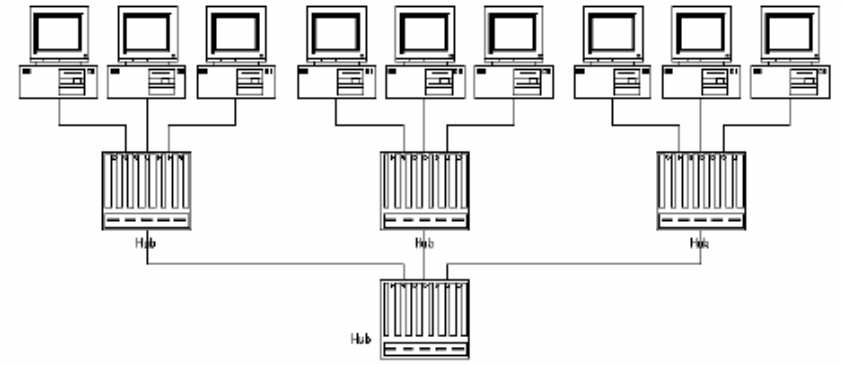


Hình 3.12. Chuẩn 10 Base- T

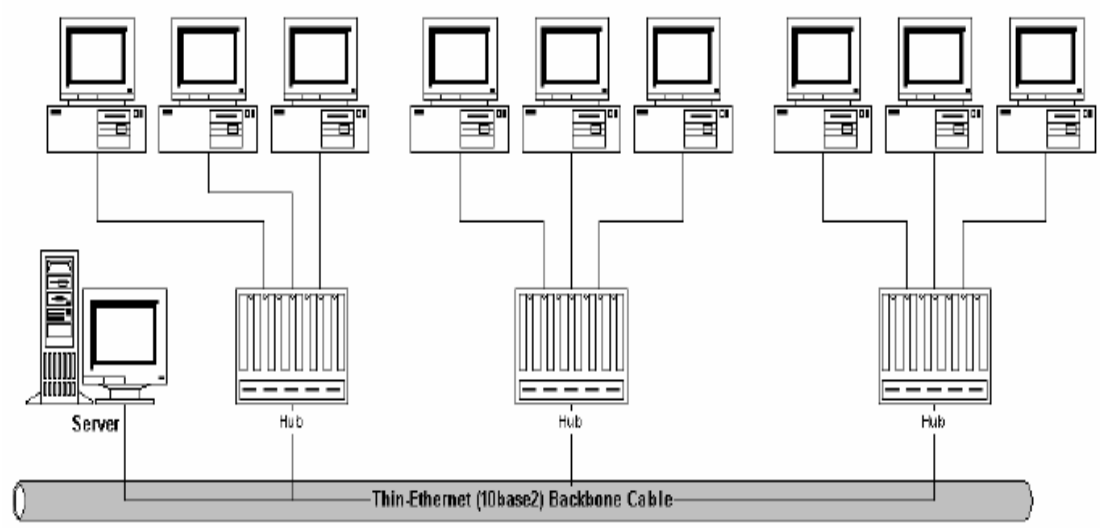
♦ **Quy tắc mở rộng mạng**

- Vì HUB là một loại Repeater nhiều cổng nên để mở rộng mạng có thể liên kết nối tiếp các HUB với nhau và cũng không được có quá 4 HUB giữa hai trạm làm việc bất kỳ của mạng (hình 3.13.)
- HUB có khả năng xếp chồng: là loại HUB có cổng riêng để liên kết các chúng lại với nhau bằng cáp riêng thành như một HUB. Như vậy dùng loại HUB này người dùng có thể dễ dàng mở rộng số cổng của HUB trong tương lai khi cần thiết. Tuy nhiên số lượng HUB có thể xếp chồng cũng có giới hạn và phụ thuộv vào từng nhà sản xuất, thông thường không vượt quá 5 HUB (hình 3.14).
- 10BASE-2 với HUB: Dù HUB có khả năng xếp chồng, người sử dụng có thể tăng số lượng máy kết nối trong mạng nhưng bán kính hoạt động của mạng vẫn không thay đổi vì khoảng cách từ cổng HUB đến NIC không thể vượt quá 100m. Một giải pháp để có thể mở rộng được bán kính hoạt động của mạng là dùng HUB có hỗ trợ một cổng AUI để liên kết các HUB bằng cáp đồng trục béo theo chuẩn 10BASE-2 (hoặc 10 Base-5). Một cáp đồng trục béo theo chuẩn 10BASE-5 có chiều dài tối đa là 500m (hình 3.15).





Hình 3.14. Mở rộng mạng 10 Base T khi số Hub nhiều hơn 4



Hình 3.15. Mở rộng mạng 10 Base T với mạng10 Base-2 làm trục chính

3.5.2. Token Ring

Chuẩn Token Ring hay còn được gọi rõ hơn là IBM Token Ring được phát triển bởi IBM, đảm bảo tốc độ truyền thông qua 4 Mbps hoặc 16 Mbps. Chuẩn này được IEEE chuẩn hoá với mã IEEE802,5 và được ISO công nhận với mã ISO 8802,5.

♦ Mô hình phần cứng

- Topo hình vòng tròn
- Dùng các MAU (multistation Access Unit) nhiều cổng MAU và cáp STP để liên kết các MAU thành một vòng tròn khép kín.
- Các trạm làm việc được liên kết vào mạng bằng các đoạn cáp STP nối từ cổng MAU tới cổng của NIC. Chiều dài đoạn cáp này được quy định dưới 100m. Số lượng

tối đa các trạm làm việc trên một Ring là 72(4Mbps)và 260(16Mbps)khoảng cách tối đa giữa hai trạm là 770m(4Mbps)và 346(16Mbps).

- Hiện tại chuẩn mạng này cũng đã hỗ trợ sử dụng cáp UTP với connector RJ45 và cáp sợi quang với connector SC.

♦ Cơ chế thâm nhập: Thâm nhập theo cơ chế phân phố lần lượt theo thẻ bài (Token)

3.5.3. FDDI (Fiber Distributed Data Interface)

Được chuẩn hoá bởi ANSI, đảm bảo tốc độ đường truyền 100Mbps.

♦ Mô hình phần cứng.

- Topo dạng vòng kép
- Dùng đôi cáp sợi quang multimode để liên kết các cáp nối DAS, SAS, DAC và SAC thành một vòng kép khép kín. Chiều dài tối đa của vòng là 100 km (200km khi vòng kép chuyển thành vòng đơn)
- DAS (Dual Attachment Station)-Bộ kết nối kép; SAS (Single Attachment Station)-Bộ kết nối đơn; DAC (Dual Attachment Concentrator)-Bộ tập trung kết nối; SAC (Single Attachment Concentrator)-Bộ tập trung kết nối đơn.
- Mỗi trạm làm việc kết nối với các bộ kết nối qua FDDI NIC bằng một hoặc hai đôi cáp sợi quang với đầu nối SC. Số trạm làm việc tối đa có thể nối vào một vòng là 500. Khoảng cách tối đa giữa hai trạm là 2 km.
- Nhờ sử dụng vòng kép nên chuẩn FDDI đã xây dựng được một cơ chế quản lý và tự khác phục sự cố trên đường truyền một cách khá hoàn hảo. Bình thường, mỗi trạm làm việc trao đổi thông tin với mạng ở chế độ dual với một đường gửi và một đường nhận thông tin đồng thời. Nếu một trong hai vòng bị sự cố, thông tin sẽ được gửi và nhận tại mỗi trạm trên cùng một đường truyền một cách luôn phiên. Nếu cả hai vòng cùng bị sự cố tại một điểm vòng kép cũng sẽ được khôi phục tự động thành một vòng đơn do tín hiệu được phản xạ tại hai bọ kết nối ở hai vị trí gần nhất hai bên điểm xảy ra sự cố.

♦ Cơ chế thâm nhập: dùng cơ chế thẻ bài

3.5. CÂU HỎI VÀ BÀI TẬP

(đang tiếp tục bổ sung)

CHƯƠNG 4. NHỮNG VẤN ĐỀ CƠ BẢN CỦA MẠNG MÁY TÍNH

4.1. KIỂM SOÁT LỖI

4.1.1. Phương pháp phát hiện lỗi với bít chẵn lẻ

Phương pháp phát hiện đơn giản nhất là VRC. Theo phương pháp này, mỗi xâu bit biểu diễn ký tự truyền đi được thêm vào một bit, gọi là (parity bit) hay là bit chẵn lẻ. Bit này có giá trị (tuỳ theo quy ước) là 0 nếu số lượng các bit 1 trong xâu là chẵn, 1 nếu số các bit 1 là lẻ. Bên nhận sẽ căn cứ vào đó để phát hiện lỗi.

Tuy nhiên phương pháp này không định vị được bit lỗi nên không tự sửa được lỗi mà chỉ yêu cầu phát lại, mặt khác nó cũng không phát hiện được lỗi kép (tức là có 2*n* bit lỗi).

Để khắc phục người ta dùng thêm phương pháp LRC. LRC kiểm tra bit chẵn lẻ theo từng khối các ký tự. Kết hợp cả hai phương pháp sẽ cho phép kiểm soát lỗi theo cả hai chiều, cho hiệu quả cao hơn so với việc dùng riêng từng phương pháp.

***Ví dụ**:* Lập bảng chèn các bit càn thiết khi truyền xâu NETWORK (cho N = 78, E= 69, T = 84, W = 87, O = 79, R = 82 và K = 75)

Đổi các giá trị mã ASCII của các ký tự trong xâu: N = 78 = 1001110, E = 69 = 1000101, T = 84 = 1010100, W = 87 = 1010111, O = 79 = 1001111, R = 82 = 1010010, K = 75 = 1001011.

Ta có bảng tính VRC, LRC như sau:

Vị trí bít trong ký tự	Khối ký tự truyền đi							LRC
	N	E	T	W	O	R	K	↓
1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	0
3	0	0	1	1	0	1	0	1
4	1	0	0	0	1	0	1	1
5	1	1	1	1	1	0	0	1
6	1	0	0	1	1	1	1	1
7	0	1	0	1	1	0	1	0
VRC →	0	1	1	1	1	1	0	1

4.1.2. Phương pháp mã sửa sai Hamming

Ta rải đều các bit chẵn lẻ lẫn vào các bit dữ liệu theo nguyên lý cân bằng chẵn lẻ để chỉ ra vị trí lỗi. Nếu ta dùng *r* bit chẵn lẻ thì sẽ kiểm tra được $2^r - 1$ bit dữ liệu. Các vị trí bit cần chèn là các luỹ thừa của cơ số 2 như: 1, 2, 4, 8,... 2^{r-1}. Tổng số bit truyền đi là $r+n$.

Phương pháp này chỉ cho phép phát hiện và sửa được đúng 1 bit lỗi, nghĩa là nếu có từ 2 bit lỗi trở lên thì phương pháp này không áp dụng được.

Thí dụ: Cho chuỗi bit gốc M = 1101100111, hãy chèn các bit cần thiết vào teho phương pháp mã sửa sai Hamming.

Chuỗi bit gốc 1101100111. Các bước thực hiện:

Chuỗi bit có 10 bit → cần 4 bit chẵn lẻ, gọi là c_8, c_4, c_2, c_1 để chèn vào các vị trí 8, 4, 2, 1.

1 1 0 1 1 0 c_8 0 1 1 c_4 1 c_2 c_1

- Cộng modulo 2 tất cả các vị trí khác không trong chuỗi vừa thu được, đó là các vị trí: 14, 13, 11,10, 6, 5, 3:

14 = 1110
13 = 1101
11 = 1011
10 = 1010 ⊕
6 = 0110
5 = 0101
3 = 0011
0010 = $c_8c_4c_2c_1$

- Kết quả: c_8 = 0, c_4 = 0, c2 = 1, c_1 = 0.

Rải các bit vừa tính được vào trong xâu gốc ta được xâu cần truyền là:

1 1 0 1 1 0 0 0 1 1 0 1 1 0

- Giả sử trên đường truyền bit thứ 7 bị lỗi, tức là giá trị của nó bị đổi từ 0 thành 1.

Tại trạm thu ta sẽ tiến hành lại thao tác cộng modul 2 tất cả các vị trí khác 0 trong chuỗi bit vừa nhận được, đó là 14, 13, 11,10, 7, 6, 5, 3

14 = 1110
13 = 1101
11 = 1011
10 = 1010 ⊕
7 = 0111
6 = 0110
5 = 0101
3 = 0011
2 = 0010
0111 = 7

Kết quả thu được bit thứ 7 là bít lỗi.

4.1.3. Phương pháp mã dư vòng (CRC)

Tư tưởng của phương pháp CRC:

- Chọn trước một đa thức (gọi là đa thức sinh) G(x) với hệ số bậc cao nhất và thấp nhấp đều bằng 1.

- Tìm tập bit kiểm tra Checksum thoả mãn điều kiện: đa thức tương ứng với xâu ghếp (xâu gốc và checksum) phải chia hết (theo modulo 2) cho G(x).
- Khi nhận tin, bên nhận kiểm tra lỗi bằng cách lấy xau bit nhận được chia (modulo 2) cho G(x). Nếu không chia hết thì có nghĩa là đã có lỗi (ngược lại thì cũng chưa thể khẳng định là không có lỗi).

Giả sử G(x) có bậc là r, xâu bit gốc tương ứng với da thức M(x) có bậc m. Các bước tính checksum như sau:

(1) Thêm r bit 0 vào cuối xâu bit cần truyền: xâu ghép sẽ gồm có m+r bit tương ứng với đa thức $x^rM(x)$.

(2) Chia (modulo 2) xâu bit tương ứng cho xâu bit tương ứng với G(x)

(3) Lấy xâu bit bị chia trừ (modulo2) cho số dư. Kết quả là xâu bit được truyền đi (xâu gốc+checksum). Ký hiệu đa thức tương ứng với nó là T(x), rõ ràng T(x) chia hết (modulo 2) cho G(x).

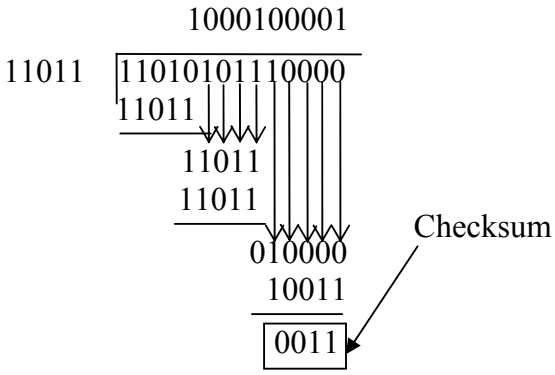
Thí dụ áp dụng: Cho xâu bit 1101010111, đa thức sinh $G(x) = x^4 + x^3 + x + 1$, hãy tính xâu bit được truyền đi trên mạng.

Xâu bit gốc 1101010111 tương ứng với đa thức $M(x) = x^9 + x^8 + x^6 + x^4 + x^2 + x + 1$.

Đa thức sinh $G(x) = x^4 + x^3 + x + 1$ tương ứng với xâu bit 11011.

$x^rM(x)$ = 11010101110000.

Ta tính checksum như sau:



Kết quả:

Checksum = 0011.

Xâu bit được truyền đi trên mạng là: T(x) = 11010101110011.

4.2. ĐIỀU KHIỂN LƯU LƯỢNG VÀ ĐIỀU KHIỂN TẮC NGHẼN

4.2.1. Các khái niệm

Điều khiển lưu lượng liên quan đến việc vận chuyển giữa một người gửi đã biết nào đó và một người nhận. Nhiệm vụ của nó là đảm bảo rằng bên gửi có tốc độ nhanh không thể tiếp tục truyền dữ liệu nhanh hơn mức mà bên nhận có thể tiếp thu được. Điều

khiển lưu lượng luôn luôn liên quan đến một sự phản hồi trực tiếp từ phía người nhận đến người gửi để báo cho bên gửi về khả năng nhận số liệu thực của bên nhận.

Điều khiển tắc nghẽn thực hiện nhiệm vụ đảm bảo cho mạng có khả năng vận chuyển lưu lượng đưa vào, đó là một vấn đề toàn cục, liên quan đến hành vi của mọi nút mạng, quá trình chứa và chuyển tiếp trong mỗi nút mạng và các yếu tố khác có khuynh hướng làm giảm thông lượng của mạng.

Điều khiển lưu lượng và điều khiển tắc nghẽn là hai khái niệm khác nhau, nhưng liên quan chặt chẽ với nhau. Điều khiển lưu lượng là để tránh tắc nghẽn, còn điều khiển tắc nghẽn là để giải quyết vấn đề tắc nghẽn khi nó xuất hiện hoặc có dấu hiệu sắp xảy ra. Trong thực tế triển khai thực hiện các thuật toán điều khiển lưu lượng và điều khiển tắc nghẽn, nhiều khi cả hai thuật toán này cùng được cài đặt trong một giao thức, thể hiện ra như là một thuật toán duy nhất, thí dụ trong giao thức TCP.

Các tầng có thể thực hiện điều khiển lưu lượng

Có thể thực hiện điều khiển lưu lượng ở một vài tầng trong mạng, thí dụ:

Điều khiển lưu lượng ở tầng Giao vận: thường được gọi là điều khiển lưu lượng đầu cuối - đầu cuối: nhằm tránh cho bộ đệm của quá trình nhận tại đích khỏi bị tràn.

Điều khiển lưu lượng trên từng chặng: nhằm tránh cho từng đường truyền khỏi bị tắc nghẽn. Tuy nhiên, việc điều khiển lưu lượng trên từng chặng sẽ có ảnh hưởng đến các chặng khác, do đó nó cũng có tác dụng tránh tắc nghẽn cho các đường truyền có nhiều chặng. Trong mô hình tham chiếu OSI, điều khiển lưu lượng theo từng chặng được thực hiện ở tầng Liên kết dữ liệu và tầng Mạng.

4.2.2. Điều khiển lưu lượng theo cơ chế cửa sổ trượt

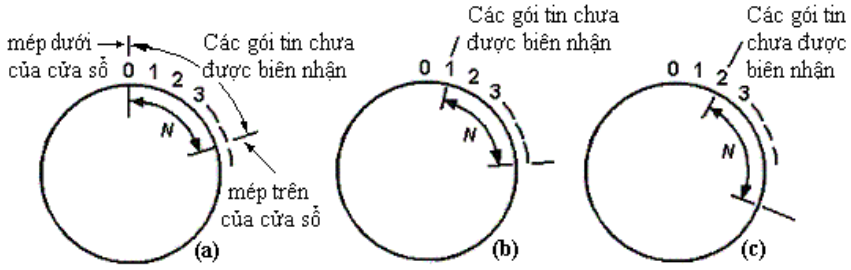
Đây là một trong các cơ chế điều khiển lưu lượng được sử dụng rộng rãi nhất, có thể áp dụng tại một hay nhiều tầng của mạng, thường là tầng Liên kết dữ liệu, tầng Mạng hay tầng Giao vận.

Cơ chế điều khiển lưu lượng bằng cửa sổ trượt cho phép bên gửi phát đi liên tiếp một số gói số liệu nhất định rồi mới phải dừng lại chờ thông báo về kết quả nhận, gọi là biên nhận, trước khi tiếp tục phát. Bên nhận điều khiển lưu lượng bằng cách kìm lại hay gửi ngay biên nhận, đó là một gói số liệu điều khiển, hoặc một gói số liệu có chứa thông tin điều khiển, mà bên nhận dùng để báo cho bên gửi biết về việc đã nhận một hay một số gói số liệu như thế nào. Tại mọi thời điểm, bên gửi phải ghi nhớ một danh sách chứa số thứ tự liên tiếp các gói số liệu mà nó được phép gửi đi, các gói số liệu này được gọi là nằm trong cửa sổ gửi. Tương tự như vậy, bên nhận cũng duy trì một danh sách gọi là cửa sổ nhận, tương ứng với các gói số liệu mà nó được phép nhận. Hai cửa sổ gửi và nhận không nhất thiết phải có độ lớn bằng nhau.

Người ta đã đề xuất và sử dụng một số phương thức quản lý cửa sổ khác nhau, thí dụ: biên nhận riêng rẽ cho mỗi gói số liệu nhận được, biên nhận ở cuối cửa sổ, biên nhận ở đầu cửa sổ v.v. Dưới đây sẽ trình bày hai cách đã được sử dụng phổ biến trong số các cách đã được nêu.

Biên nhận từng gói số liệu

Theo cách quản lý này, mỗi khi nhận được một gói số liệu, bên nhận sẽ gửi một biên nhận cho bên gửi. Cửa sổ gửi tương ứng với các gói số liệu đã gửi đi nhưng chưa được biên nhận (hình 4.1.a). Khi có một gói số liệu mới từ tầng trên chuyển xuống để gửi đi, nó sẽ được gán số thứ tự lớn nhất tiếp theo, do đó mép trên của cửa sổ gửi sẽ tăng thêm 1. Nếu cửa sổ tăng tới cực đại thì tiến trình truyền ở tầng trên bị chặn lại, không thể truyền các gói số liệu xuống nữa, cho đến khi có chỗ trống trong vùng nhớ đệm. Mỗi gói số liệu sau khi đến đích sẽ được bên nhận biên nhận một cách riêng rẽ. Khi biên nhận về đến bên gửi, mép dưới của cửa sổ gửi sẽ được tăng thêm 1, làm cho danh sách các gói số liệu đã truyền nhưng còn chờ biên nhận giảm đi một phần tử, đồng thời vùng nhớ tương ứng với phần tử đó cũng được giải phóng để cấp phát cho một gói số liệu mới (hình 4.1.b, c). Bằng cách này, cửa sổ gửi luôn ghi nhớ được danh sách các gói số liệu còn chưa được biên nhận. Vì các gói số liệu trong cửa sổ gửi có thể bị hỏng hoặc mất trên đường truyền, nên bên gửi phải giữ lại bản sao của chúng trong bộ nhớ đệm để phát lại nếu sau một khoảng thời gian nhất định vẫn không nhận được biên nhận.



Hình 4.1. Điều khiển lưu lượng bằng cửa sổ trượt, biên nhận từng gói số liệu.
a) Trạng thái ban đầu. b) Gói số 0 được biên nhận. c) Gói số 1 được biên nhận

Các gói số liệu nằm ngoài cửa sổ nhận nếu đến sẽ bị loại bỏ. Khi nhận được một gói số liệu có số thứ tự bằng mép dưới của cửa sổ, nó sẽ được truyền cho tầng trên, bên nhận sinh ra một biên nhận gửi tới người gửi và tăng cửa sổ lên một ô. Nếu kích thước cửa sổ nhận bằng 1, có nghĩa là nó chỉ chấp nhận các gói số liệu đến theo đúng thứ tự. Nếu khác 1 thì không phải như vậy, trong trường hợp này, bên nhận sẽ giữ gói số liệu đến không đúng thứ tự trong bộ đệm, chờ nhận đủ các gói số liệu trong cửa sổ rồi mới chuyển các gói số liệu lên tầng trên theo thứ tự mà bên gửi đã gửi đi. Khoảng thời gian chờ này luôn được giới hạn.

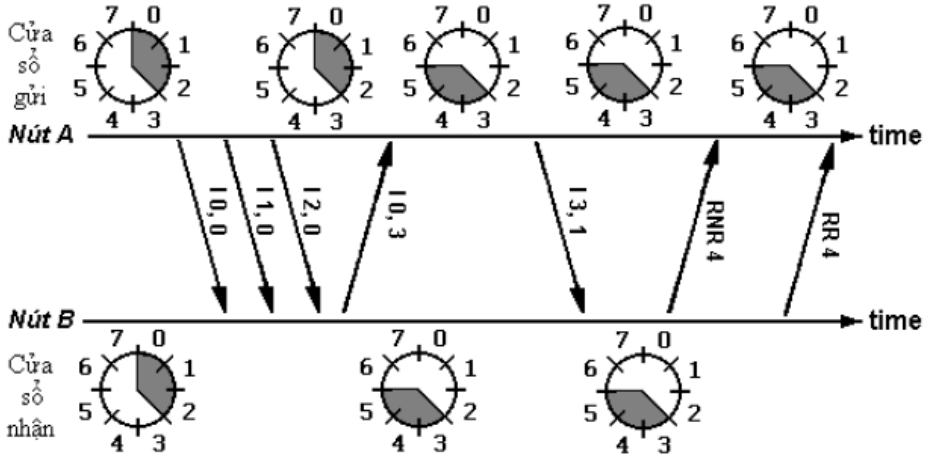
Biên nhận ở cuối cửa sổ

Đây là cách đơn giản nhất, bên nhận sẽ phát ra một biên nhận sau khi nhận được tất cả các gói số liệu trong cửa sổ nhận. Hình 4.2 minh hoạ cho phương pháp này, trong đó nút A truyền thông với nút B, sử dụng một giao thức tựa như giao thức HDLC, kích thước cửa sổ gửi và cửa sổ nhận ban đầu bằng 3. Các gói số liệu đi trên mạng được biểu diễn bằng các mũi tên, kiểu của gói số liệu được ghi bên cạnh mũi tên, ý nghĩa như sau:

- I n.m: là gói số liệu, với trường số thứ tự gói số liệu N(S) = n, trường biên nhận N(R) = m. Bên B cho gói số liệu của nó "cõng" ("piggyback") biên nhận tới bên A, việc này giúp nâng cao hiệu quả sử dụng đường truyền. Tất nhiên, bên A cũng có thể biên nhận các gói số liệu mà B gửi cho nó bằng cách trên.
- RNR 4: là gói số liệu điều khiển, B báo cho A rằng, lúc này nó không thể nhận tiếp các gói số liệu của A, đồng thời biên nhận cho các gói số liệu có số thứ tự nhỏ hơn hoặc

bằng 3. Khi nhận được tín hiệu này, A sẽ phải ngừng gửi, chờ cho đến khi nhận được tín hiệu cho phép gửi tiếp của B.

- RR 4: là gói số liệu điều khiển, B báo cho A rằng, lúc này nó sẵn sàng nhận tiếp các gói số liệu của A, bắt đầu từ gói số 4.

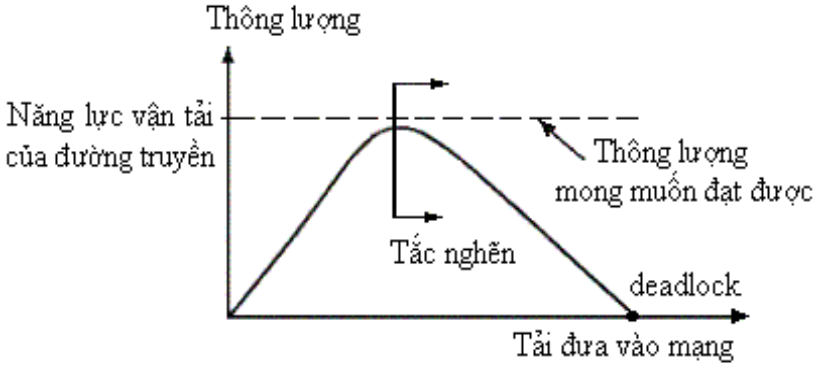


Hình 4.2. Điều khiển lưu lượng bằng cửa sổ trượt, biên nhận ở cuối cửa sổ. Kích thước cửa sổ nhận và gửi ban đầu bằng 3.

4.2.3. Điều khiển tắc nghẽn

4.2.3.1. Hiện tượng tắc nghẽn

Trong mạng máy tính, tắc nghẽn xảy ra khi số lượng gói số liệu đến nút mạng vượt quá khả năng xử lý của nó hoặc vượt quá khả năng vận tải của các đường truyền ra, điều đó dẫn đến việc thông lượng của mạng bị giảm đi khi lưu lượng đến mạng tăng lên. Hiện tượng tắc nghẽn có thể xảy ra ở một hoặc một số nút mạng, hay trên toàn mạng và được miêu tả trên hình 4.3.



Hình 4.3. Hiện tượng tắc nghẽn

Khi số lượng gói số liệu đến mạng còn tương đối nhỏ, nằm trong khả năng vận tải của nó, chúng sẽ được phân phát đi hết, số lượng gói số liệu được chuyển đi tỉ lệ thuận với số lượng gói số liệu đến mạng. Do luôn có một tỉ lệ gói số liệu phải phát lại do bị lỗi trong quá trình vận chuyển, lưu lượng mà mạng thực sự phải vận chuyển nhìn chung lớn hơn lưu lượng đi qua mạng (thông lượng).

Khi lưu lượng đến cao quá một mức nào đó, các nút mạng không còn đủ khả năng chứa và chuyển tiếp các gói số liệu, do đó các nút mạng bắt đầu phải loại bỏ các gói số liệu. Bên gửi sẽ phát lại các gói số liệu không được biên nhận sau một khoảng thời gian nhất định, gọi là thời gian hết giờ. Nếu lưu lượng đến mạng tiếp tục tăng lên nữa, tỉ lệ gói số liệu phát lại trên tổng số gói số liệu trong mạng có thể tăng đến 100%, nghĩa là không có gói số liệu nào được phân phát đi cả, thông lượng của mạng giảm xuống bằng không, mạng bị nghẹt hoàn toàn.

Một số yếu tố có thể dẫn đến tắc nghẽn, ngay cả khi lưu lượng đi vào mạng thấp hơn khả năng vận tải của mạng. Chẳng hạn, khi số lượng gói số liệu đến trên hai hoặc ba lối vào của một nút mạng đều cần đi ra trên cùng một đường truyền để đến đích, chúng sẽ phải xếp hàng đợi được truyền đi. Nếu tình trạng trên kéo dài, hàng đợi sẽ dài dần ra và đầy, không còn chỗ cho các gói số liệu mới đến, chúng bị loại bỏ và sẽ được phát lại, làm tăng tỉ lệ gói số liệu phát lại trong mạng. Biện pháp khắc phục bằng cách tăng kích thước hàng đợi (bộ nhớ) tại các nút mạng trong một chừng mực nào đó là có ích, tuy nhiên, người ta đã chứng minh được rằng, tăng kích thước hàng đợi quá một giới hạn nào đó sẽ không mang lại lợi ích gì, thậm chí còn có thể làm cho vấn đề tắc nghẽn tồi tệ hơn. Đó là vì các gói số liệu sẽ bị hết giờ ngay trong quá trình xếp hàng, bản sao của chúng đã được bên gửi phát lại rồi, làm tăng số lượng gói số liệu phát lại trong mạng.

Tốc độ xử lý chậm của các nút mạng cũng là một nguyên nhân quan trọng gây nên tắc nghẽn, bởi vì chúng có thể sẽ làm hàng đợi bị tràn ngay cả khi lưu lượng gói số liệu đến nút mạng nhỏ hơn năng lực vận tải của đường truyền đi ra. Các đường truyền dung lượng thấp cũng có thể gây ra tắc nghẽn. Việc tăng dung lượng đường truyền nhưng không nâng cấp bộ xử lý tại nút mạng, hoặc chỉ nâng cấp từng phần của mạng đôi khi cũng cải thiện được tình hình đôi chút, nhưng thường chỉ làm cái "cổ chai", nơi xảy ra tắc nghẽn, dời đi chỗ khác mà thôi. Giải quyết vấn đề tắc nghẽn nói chung, cần đến các giải pháp đồng bộ.

Tắc nghẽn có khuynh hướng tự làm cho nó trầm trọng thêm. Nếu một nút mạng nào đó bị tràn bộ đệm, gói số liệu đến sẽ bị loại bỏ, trong khi đó nút mạng bên trên, phía người gửi, vẫn phải giữ bản sao của gói số liệu đã gửi trong hàng đợi, cho đến khi hết giờ để phát lại. Việc phải giữ bản sao gói số liệu trong hàng đợi để chờ biên nhận, cộng thêm việc có thể phải phát lại gói số liệu một số lần có thể làm cho hàng đợi tại chính nút trên cũng có thể bị tràn. Sự tắc nghẽn lan truyền ngược trở lại phía nguồn phát sinh ra gói số liệu.

4.2.3.2. Các giải pháp điều khiển tắc nghẽn

Vấn đề điều khiển tắc nghẽn có thể được giải quyết theo quan điểm của Lý thuyết điều khiển. Theo cách tiếp cận này, có thể chia các giải pháp thành hai nhóm: các giải pháp Vòng lặp mở (Open loop) và các giải pháp Vòng lặp đóng (Closed loop). Theo các giải pháp vòng lặp mở, tắc nghẽn sẽ được giải quyết bằng việc thiết kế tốt, đảm bảo sao cho tắc nghẽn không xảy ra. Một hệ thống như vậy phải có khả năng quyết định khi nào thì nhận thêm các lưu lượng mới vào, khi nào thì loại bỏ các gói số liệu và loại các gói số liệu nào. Các quyết định này phải theo lịch trình và phải có ở từng nút mạng, chúng được hệ thống đưa ra mà không xem xét đến trạng thái hiện thời của mạng.

Trái lại, các giải pháp vòng lặp đóng lại dựa trên khái niệm về vòng phản hồi (feedback loop), chúng gồm có ba phần, hay ba bước như sau:

Bước một: theo dõi hệ thống để phát hiện tắc nghẽn xảy ra khi nào và ở đâu.

Việc phát hiện tắc nghẽn có thể dựa trên một số độ đo khác nhau. Các độ đo thường được sử dụng là tỉ lệ gói số liệu bị loại bỏ do thiếu bộ đệm, chiều dài trung bình của hàng đợi, số gói số liệu phải phát lại do bị hết giờ, thời gian trễ trung bình của gói số liệu khi đi qua mạng v.v. Sự tăng lên của các số đo này nói lên rằng tắc nghẽn đang tăng lên trong mạng.

Bước hai: nơi phát hiện ra tắc nghẽn cần phải chuyển thông tin về sự tắc nghẽn đến những nơi có thể phản ứng lại. Một cách thực hiện rất đơn giản là nút mạng phát hiện ra tắc nghẽn sẽ gửi gói số liệu đến các nguồn sinh lưu lượng trên mạng, báo tin về sự cố. Tất nhiên, việc này sẽ làm tăng thêm lưu lượng đưa vào mạng đúng lúc lẽ ra phải giảm đi. Người ta cũng đã đề xuất và thực hiện một số cách khác nữa. Chẳng hạn, nút mạng phát hiện ra tắc nghẽn sẽ đánh dấu vào một bit hay một trường định trước của mọi gói số liệu trước khi gói số liệu được nút mạng chuyển tiếp đi, nhằm loan báo cho các nút mạng khác về trạng thái tắc nghẽn. Có thể nêu ra một cách thực hiện khác nữa, đó là làm cho các nút mạng đều đặn gửi đi các gói số liệu thăm dò để biết tình trạng của mạng.

Bước ba: điều chỉnh lại hệ thống để sửa chữa sự cố. Các cơ chế thực hiện phản hồi đều nhằm mục đích là để các máy tính trên mạng có những phản ứng phù hợp nhằm làm giảm tắc nghẽn. Nếu phản ứng xảy ra quá nhanh, lưu lượng trong hệ thống sẽ thăng giáng mạnh và không bao giờ hội tụ. Nếu phản ứng quá chậm, việc điều khiển tắc nghẽn có thể không có ý nghĩa thực tế gì nữa. Chính vì vậy, để cơ chế phản hồi có hiệu quả, cần phải sử dụng một số cách tính trung bình.

Các thuật toán điều khiển tắc nghẽn sẽ được trình bày cụ thể trong chương 5, phần giao thức TCP.

4.3. AN TOÀN THÔNG TIN TRÊN MẠNG

4.3.1. Giới thiệu

Đặc điểm của môi trường mạng là *nhiều người sử dụng* và *phân tán về địa lý*. Số lượng users tăng lên dẫn tới mức độ quan trọng của thông tin ngày càng lớn như thông tin về tài chính, ngân hàng, chứng khoán, chính phủ,..liên quan đến an ninh quốc gia, lợi ích quốc tế, các cơ quan doanh nghiệp lớn. Các thông tin này được trao đổi từng giờ từng phút trên mạng, do vậy phải có các biện pháp bảo đảm an toàn thông tin trên mạng.

Nhóm người xâm hại mạng gồm nhiều loại với hành vi ngày một tinh vi hơn, các đối tượng đó gồm: sinh viên, doanh nhân, người môi giới chứng khoán, hacker, khủng bố,..

Mục đích của mỗi đối tượng:

- Sinh viên: tò mò, nghịch ngợm, muốn chứng tỏ khả năng.
- Doanh nhân: thăm dò chiến lược kinh doanh của đối thủ cạnh tranh để phá hoại đối phương.

- Người môi giới chứng khoán: tác động tâm lý người chơi chứng khoán, thuyết phục người chơi mua/bán cổ phiếu, làm đảo lộn thị trường.
- Hacker: tấn công vào lỗ hổng phần mềm để ăn cắp bản quyền hoặc phá huỷ hệ thống
- Khủng bố: tổ chức các vụ khủng bố nhằm mục đích kinh tế, chính trị,..

Để bảo vệ thông tin đạt hiệu quả cao chúng ta cần phải lường trước được tốt các khả năng xâm phạm, các sự cố rủi ro đối với các thiết bị và dữ liệu trên mạng. Xác định càng chính xác các nguy cơ thì việc tìm ra các giải pháp sẽ chính xác và giảm thiểu các thiệt hại. *ATTT trên mạng đề cập đến những biện pháp bảo vệ thông tin tránh khỏi những xâm phạm trái phép.*

Các loại vi phạm có thể được chia làm hai loại: *vi phạm chủ động* và *vi phạm thụ động*. Chủ động và thụ động ở đây được hiểu theo nghĩa có can thiệp vào nội dung và luồng thông tin trao đổi hay không? Vi phạm thụ động chỉ nhằm mục tiêu cuối cùng là nắm bắt thông tin, có thể không biết được nội dung, nhưng vẫn có thể dò ra được thông tin về người gửi, người nhận nhờ vào thông thông tin điều khiển nằm trong header của gói tin. Hơn nữa, kẻ phá hoại còn có thể kiểm tra được số lượng, độ dài, lưu lượng, tần suất trao đổi dữ liệu trao đổi. Tóm lại, vi phạm loại này không làm sai lệch hoặc hủy hoại thông tin. Trong khi đó, các vi phạm chủ động có thể làm biến đổi, xóa bỏ, làm trễ, sắp xếp lại thứ tự, hoặc chèn vào một số các thông tin ngoại lai vào để làm sai lệch thông tin gốc. Một hình thức vi phạm chủ động nữa là có thể làm vô hiệu các chức năng phục vụ người dùng tạm thời hoặc lâu dài.

Vi phạm thụ động thường khó phát hiện nhưng dễ ngăn chặn, ngược lại, vi phạm chủ động dễ phát hiện nhưng rất khó ngăn chặn.

Kẻ phá hoại có thể xâm nhập ở bất cứ đâu có thông tin mà họ quan tâm. Có thể là ở trên đường truyền, các máy chủ nhiều người dùng, các máy trạm, hay ở các thiết bị kết nối (bridge, router, gateway,..), các thiết bị ngoại vi, bàn phím, màn hình chính là những cửa ngõ thuật lợi cho các loại xâm nhập trái phép.

4.3.2. Các lớp bảo mật trong mạng

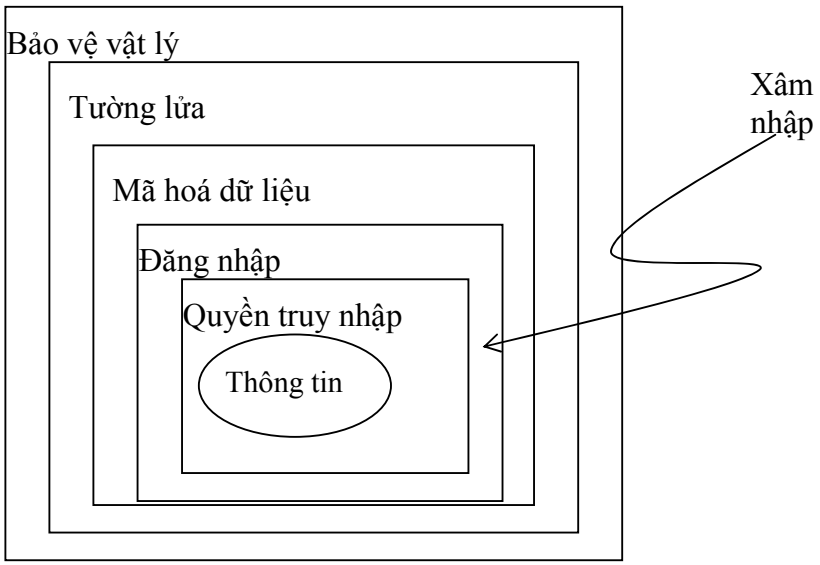
Các hệ thống an ninh mạng thường bao gồm nhiều lớp để chống lại các kiểu xâp phạm khác nhau. Sơ đồ các lớp bảo mật thông dụng hiện nay được trình bày dưới đây:

- Lớp Quyền truy nhập: Lớp bảo vệ trong cùng nhằm kiểm tra, giới hạn quyền truy cập của các đối tượng sử dụng tài nguyên mạng. Quyền truy cập quy định người sử dụng có thể truy cập vào tài nguyên gì và được phép thực hiện những thao tác gì trên đó.

- Lớp Đăng nhập: yêu cầu mỗi cá nhân khi bước vào mạng phải xuất trình Tên (User name) và Mật khẩu (Password). Đây là một cơ chế đơn giản, ít tốn kém nhưng rất hiệu quả, góp phần hạn chế ngay từ ngoài những truy xuất trái phép. Mỗi người sử dụng hợp lệ đều phải có tên và mật khẩu, dựa vào đó hệ thống nhận biết được anh ta có thể được sử dụng tài nguyên nào và thao tác gì trên những tài nguyên đó.

- Các phương pháp mã hoá chủ yếu dành cho những thông tin truyền trên mạng để tránh bị nghe trộm, nhưng chúng cũng được áp dụng cho việc bảo mật tại chỗ. Dữ liệu

được biến đổi từ dạng tự nhiên sang dạng mã hoá để gửi đi, còn tại bên nhận lại diễn ra quá trình ngược lại – giải mã. Chúng ta sẽ xem xét kỹ vấn đề này trong mục sau.



Hình 4.4. Các lớp bảo mật dữ liệu trong mạng

- Tường lửa: để bảo vệ mạng nội bộ, thông thường hiện nay các hệ thống mạng thường sử dụng các phần mềm FireWall (tường lửa). Chức năng của Tường lửa là ngăn chặn những truy nhập trái phép từ môi trường mạng bên ngoài vào hệ thống mạng nội bộ, lọc bỏ những gói tin mà ta không muốn gửi đi hoặc nhận vào, cấm những truy nhập trái phép theo một danh sách được quy định trước. Một số phần mềm Tường lửa có thể thấy như Tích hợp ngay trong Windows (Windows Firewall), Tích hợp trong các phần mềm diệt virut, Phần mềm ISA Server 2004... Phương thức này được sử dụng rộng rãi trong môi trường liên mạng Internet.

- Lớp bảo vệ vật lý nhằm ngăn chặn những thao tác sử dụng trái phép trên hệ thống. Đó là các biện pháp như: kiểm soát người ra vào phòng điều hành trung tâm, lắp ổ khoá trên máy tính, sử dụng các máy trạm không có ổ đĩa để tránh sao chép thông tin, lắp đặt các thiết bị nhận dạng (vân tay, mặt người, video) để kiểm soát ra vào...

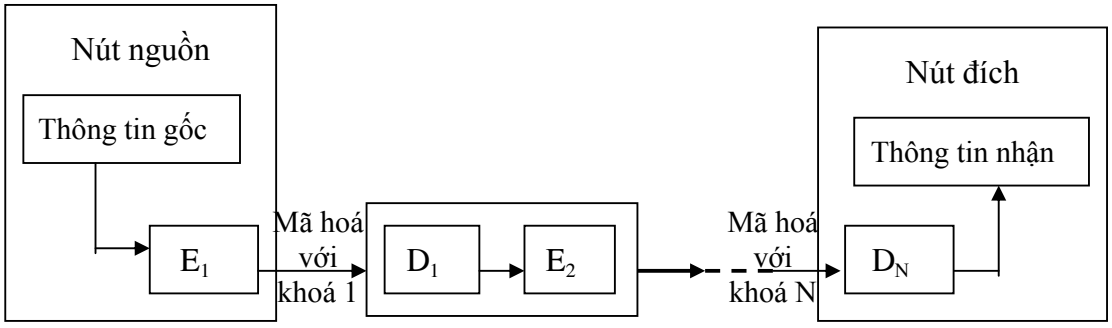
4.3.3. Bảo vệ dữ liệu bằng mật mã

Có hai cách tiếp cận để bảo vệ thông tin bằng mật mã: đso là *theo đường truyền* (link-oriented security) và từ *nút-đến-nút* (end-to-end).

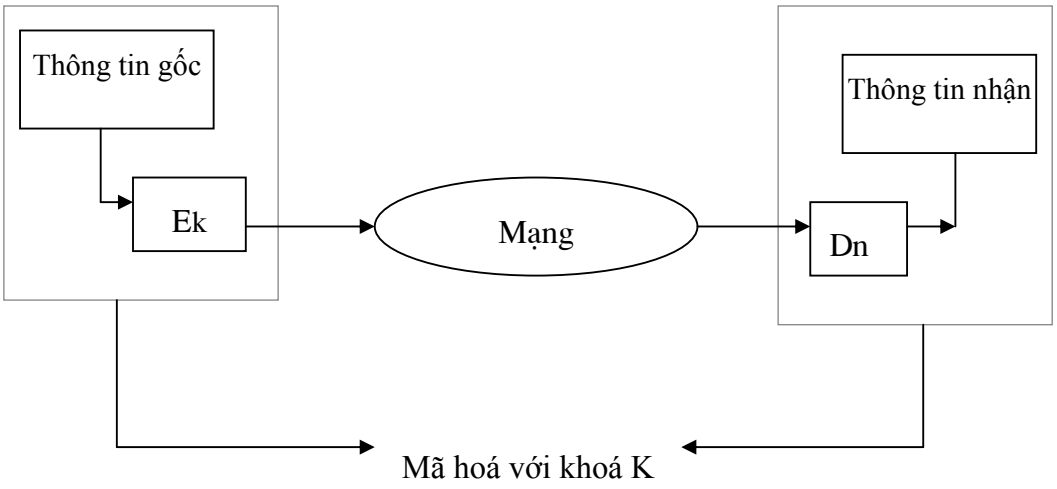
Theo cách thứ nhất, việc mã hóa chỉ thực hiện đối với thông tin trên đường truyền mà không quan tâm đến nguồn và đích của thông tin đó (hình 4.5 a). Ưu điểm: có thể bí mật được luồng thông tin giữa nguồn và đích và có thể ngăn chặn được toàn bộ các vi phạm nhằm phân tịch lưu thông trên mạng. Nhược điểm của nó là đòi hỏi các nút phải được bảo vệ tốt.

Theo cách thứ hai, thông tin được mã hóa trên toàn bộ đường đi từ nguồn đến đích (hình 4.5 b). Thông tin được mã hóa ngay khi được tạo ra và chỉ được giải mã ở trạm đích. Ưu điểm chính là: người dùng có thể sử dụng nó mà không cần quan tâm đến người

dùng khác. Nhược điểm: chỉ có dữ liệu người dùng được mã hóa, còn các thông tin điều khiển thì phải được giữ nguyên để có thể xử lý tại các nút trên đường đi.

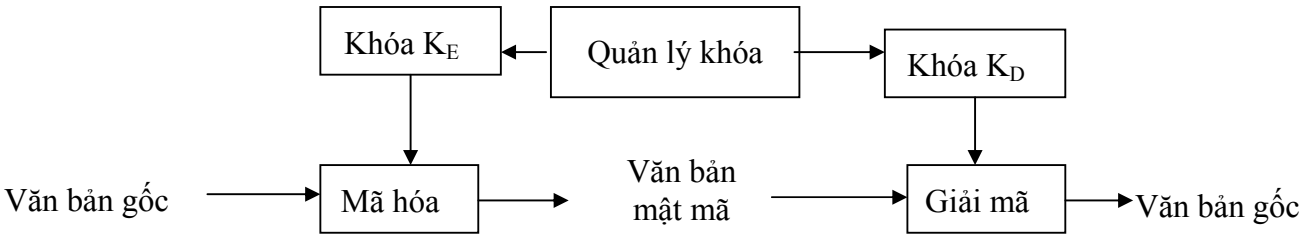


Hình 4. 5 a) Tiếp cận theo đường truyền (Link Oriented)



Hình 4.5 b) Tiếp cận nút tới nút (End - to - End)

4.3.3.1. Quy trình mật mã



Hình 4.6. Sơ đồ mật mã dữ liệu

Trong đó:

- Văn bản gốc: là văn bản chưa được mã hóa
- Khóa: gồm một xâu hữu hạn các bit thường được biểu diễn dưới dạng các xâu ký tự chữ số.

Gọi M là văn bản gốc, C là văn bản mật mã, E là hàm má hóa, D là hàm giải mã ta có: $C = E_{KE}(M)$ (đ/v mã hóa) $M = D_{KD}(C) = D_{KD}(E_{KE}(M))$ (đ/v giải mã).

Khóa K_E được dùng đê mã hóa, khóa K_D được dùng để giải mã.

Có hai phương pháp mã hóa (phân loại theo cách thức dùng khóa): phương pháp *cổ điển* (hay *khóa đối xứng*, hay *một khóa*): sử dụng một khóa duy nhất cho việc mã hóa và giải mã. Do đó, khóa phải được giữ bí mật.

Phương pháp thứ hai là sử dụng khóa *công khai*. Trong đó hệ thống sử dụng hai khóa, một để mã hóa và một để giải mã. Khóa mã hóa có thể công khai, còn khóa giải mã phải giữ bí mật.

Dưới đây chúng ta sẽ xem xét 4 phương pháp mật mã chủ yếu, đó là:

- Phương pháp đổi chỗ (Transportation Ciphers)
- Phương pháp thay thế (Subsstitution Ciphers)
- Phương pháp sử dụng chuẩn mật mã (DES)
- Phương pháp sử dụng khóa công khai (Public key)

4.3.3.2. Phương pháp đổi chỗ

Phương pháp sắp xếp lại các ký tự trong văn bản gốc để được văn bản mật mã. Các kỹ thuật dùng trong phương pháp này gồm:

- *Đảo ngược toàn bộ văn bản gốc*: văn bản gốc được viết heo thức tự ngược lại để tạo ra văn bản mật mã. Đơn giản và không an tòn.
- *Mã hóa theo dạng hình học:* Văn bản gốc được sắp xếp theo một dạng hình học nào đó, thường là một ma trận 2 chiều:

Ví dụ: Ta sẽ mã hoá xâu "KHOA KTCN HUONG TOI TUONG LAI" theo phương pháp này. Đầu tiên xâu đã cho được viết dưới dạng ma trận 4x6 như sau:

Cột	1	2	3	4	5	6
	K	H	O	A	K	T
	C	N	H	U	O	N
	G	T	O	I	T	U
	O	N	G	L	A	I

Sau đó, nếu viết các ký tự ra theo thứ tự các cột là 2, 3, 5, 1, 4, 6 thì ta được văn bản mã hoá là:

HOKKAT NHOCUN TOTGIU NGAOLI

- *Đổi chỗ cột:* Trước hết đổi chỗ trong xâu văn bản gốc thành dạng chữ nhật theo các cột, sao đó sắp xếp lại các cột và lấy ra theo chiều ngang.

Ví dụ: Ta sẽ mã hoá xâu "GIA CA THI TRUONG CO CHIEU HUONG TANG NHANH" theo phương pháp đổi chỗ cột. Đầu tiên xâu đã cho được viết dưới dạng ma trân 7x5 như sau:

Cột	1	2	3	4	5
Văn bản	G	I	C	H	N
	I	T	O	U	G
	A	R	C	O	N
	C	U	H	N	H
	A	O	I	G	A
	T	N	E	T	N
	H	G	U	A	H

Sau đó, nếu chuyển vị các cột theo thứ tự: 3, 5, 2, 1, 4 thì ta được văn bản mã hoá là:

CNIGH OGTIU CNRAO HHUCN IAOAG ENNTT UHGHA

- *Phương pháp hoán vị theo một chu kỳ cố định:*

Giả sử văn bản gốc M = $m_1m_2..m_n$ được chia thành các khối, mỗi khối gồm *d* ký tự. Cho *f* là một hàm hoán vị trên *d* ký tự, khi đó khoá mã hoá được biểu diễn bởi *K(d,f)*.

Văn bản gốc được viết thành: M = $m_1m_2..m_d\ m_{d+1}m_{d+2}..m_{2d,..}$

sẽ được mã hoá thành: $E_k(M) = m_{f(1)}m_{f(2)}..m_{f(d)}\ m_{f(d+1)}m_{f(d+2)}..m_{f(2d),...}$

trong đó $m_{f(1)}m_{f(2)}..m_{f(d)}$ là một hoán vị của $m_1m_2..m_d$.

Ví dụ: Ta sẽ mã hoá xâu "AN TOAN BAO MAT THONG TIN TREN MANG", với chu ky $d = 6$, và hàm *f* hoán vị dãy $i = 123456$ thành $f_i = 352164$:

Ta viết xâu đã cho dưới dạng: "ANTOAN BAOMAT THONGT INTREN MANG"

Xâu mã hoá trở thành: "TANANO OAABTM OGHTTN TENINR NAMG"

4.3.3.3. Phương pháp thay thế

Phương pháp này mã hoá bằng cách thay thế mỗi ký tự trong văn bản gốc bằng một ký tự khác nào đó (một chức cái, một số, hay một ký hiệu).

Có nhiều kỹ thuật thay thế: thay thế đơn giản, thay thế đồng âm, thay thế đa mẫu tự, thay thế đa sơ đồ.

– *Thay thế đơn giản* (simple substitution): mỗi ký tự trong văn bản gốc được thay thế bởi một ký tự tương ứng trong văn bản mật mã. Hàm mã hoá là một ánh xạ 1-1 từ văn bản gốc đến văn bản mật mã được sử dụng trong toàn bộ văn bản.

– *Thay thế đồng âm* (homophonic substitution): mỗi ký tự trong văn bản gốc được thay bằng một số ký tự trong văn bản mật mã, sơ đồ ánh xạ là 1-n (one-to-many).

– *Thay thế đa mẫu tự* (polyalphabetic substitution): Nhiều chứ cái mật mã được dùng để chuyển đổi từ văn bản gốc sang văn bản mật mã. Ánh xạ 1-1 như trong trường hợp thay thế đơn giản, nhưng có thể thay đổi trong phạm vị 1 thông điệp.

– *Thay thế đa sơ đồ* (polygram substitution): đây là mật mã ttỏng quát nhất, cho phép thay thế tuỳ ý các nhóm ký tự của văn bản gốc.

Vì khuôn khổ giáo trình, ở đây ta chỉ trình bày phương pháp thay thế đơn giản.

Giả sử A là bộ chữ cái gốc có *n* ký tự: $\{a_0,a_1,a_2,...,a_{n-1}\}$. Ta thay thế mỗi ký tự của A bằng một ký tự tương ứng trong bộ chữ cái mật mã C được sắp xếp thứ tự, ký hiệu là $\{f(a_0), f(a_1),f(a_2),...,f(a_{n-1})\}$. F là một hàm ánh xạ 1-1 từ 1 ký tự của A sang 1 ký tự của C.

Lúc đó, một thông điệp gốc M = m_1m_2... sẽ được viết dưới dạng như sau: $E_k(M) = f(m_1) f(m_2)$,.. Thường thì C là một sắp xếp lại của A.

Ví dụ về các dạng mã hóa thay thế đơn giản: bộ mã ASCII

Một trong những dạng mật mã thay thế đơn giản phổ dụng khác là bảng chữ cái chuyển dịch. Ở đây các chữ cái trong bảng được chuyển dịch sang phải k vị trí (modul theo kích thước của bảng chữ). Nghĩa là nếu *a* là 1 ký tự của bảng chữ A thì $f(a) = (a+k) \bmod n$. Nếu A là bảng chữ cái tiếng Anh chuẩn thì $n = 26$. Đây cũng chính là phương pháp mật mã Caesar, với $k = 3$.

Ví dụ: với $k=3$, bảng chữ cái gốc là:

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 cái mật mã sẽ là:

D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Khi đó, dòng chữ: KHOA KTCN trở thành NKRD NWFQ.

4.3.3.4. Phương pháp sử dụng chuẩn mật mã (DES)

Giới thiệu chung về DES

Chuẩn mã hoá dữ liệu DES được Văn phòng tiêu chuẩn của Mỹ (U.S National Bureau for Standards) công bố năm 1971 để sử dụng trong các cơ quan chính phủ liên bang. Giải thuật được phát triển tại Công ty IBM dựa trên hệ mã hoá LUCIFER của Feistel.

DES là thuật toán mã hoá khối (block algorithm), với cỡ của một khối là 64 bít. Một khối 64 bít bản rõ được đưa vào, sau khi mã hoá dữ liệu đưa ra là một khối bản mã 64 bít. Cả mã hoá và giải mã đều sử dụng cùng một thuật toán và khoá.

Khoá mã hoá có độ dài 64 bít, trong đó có 8 bít chẵn lẻ được sử dụng để kiểm soát lỗi. Các bít chẵn lẻ nằm ở các vị trí 8, 16, 24,... , 64. Tức là cứ 8 bít khoá thì trong đó có 1 bít kiểm soát lỗi, bít này qui định số bít có giá trị "1" của khối 8 bít đó theo tính bù chẵn.

Nền tảng để xây dựng khối của DES là sự kết hợp đơn giản của các kỹ thuật thay thế và hoán vị bản rõ (bản gốc) dựa trên khoá, đó là các vòng lặp. DES sử dụng 16 vòng lặp, nó áp dụng cùng một kiểu kết hợp của các kỹ thuật trên khối bản rõ 16 lần (hình 4.7).

Thuật toán chỉ sử dụng các phép toán số học và lôgíc trên các số 64 bít, vì vậy nó dễ dàng thực hiện vào những năm 1970 trong điều kiện về công nghệ phần cứng lúc bấy giờ. Ban đầu, sự thực hiện các phần mềm kiểu này rất thô sơ, nhưng hiện tại thì việc đó đã tốt hơn, và với đặc tính lặp đi lặp lại của thuật toán đã tạo nên ý tưởng sử dụng chíp với mục đích đặc biệt này.

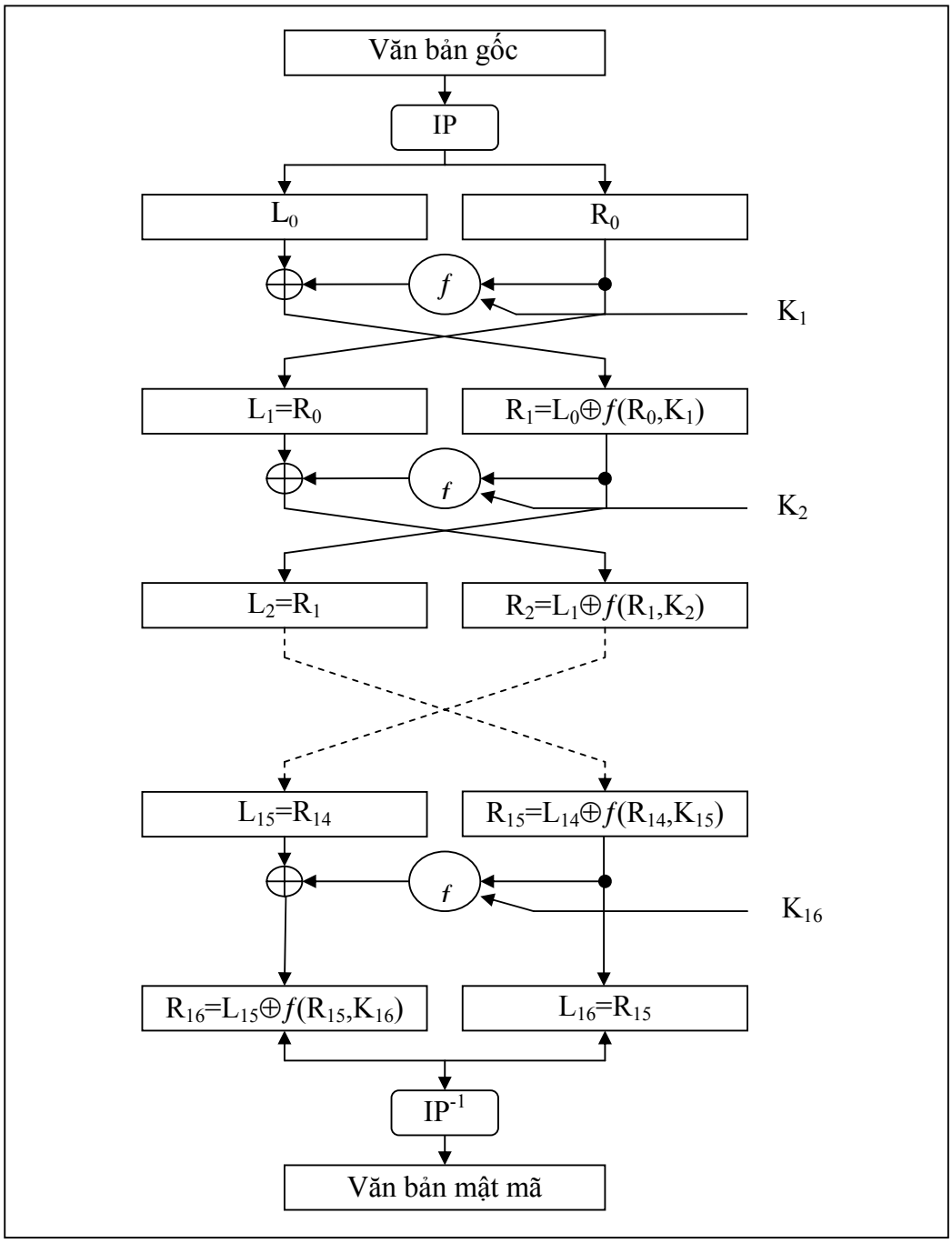
Tóm lại DES có một số đặc điểm sau:

- Sử dụng khoá 56 bít.
- Xử lý khối vào 64 bít, biến đổi khối vào thành khối ra 64 bít.
- Mã hoá và giải mã được sử dụng cùng một khoá.
- DES được thiết kế để chạy trên phần cứng.

DES thường được sử dụng để mã hoá các dòng dữ liệu mạng và mã hoá dữ liệu được lưu trữ trên đĩa.

Mô tả thuật toán

DES thực hiện trên từng khối 64 bít bản rõ. Sau khi thực hiện hoán vị khởi đầu, khối dữ liệu được chia làm hai nửa trái và phải, mỗi nửa 32 bít. Tiếp đó, có 16 vòng lặp giống hệt nhau được thực hiện, được gọi là các hàm f, trong đó dữ liệu được kết hợp với khoá. Sau 16 vòng lặp, hai nửa trái và phải được kết hợp lại và hoán vị cuối cùng (hoán vị ngược) sẽ kết thúc thuật toán.



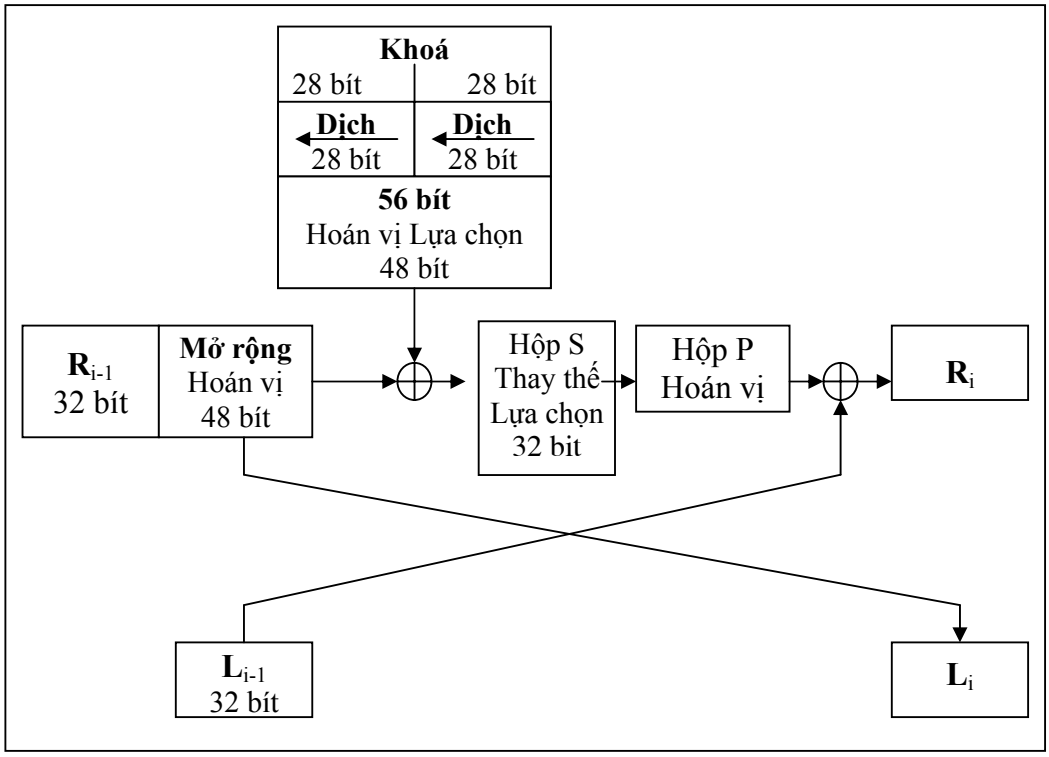
Hình 4.7. Thuật toán DES

Trong mỗi vòng lặp, các bít của khoá được dịch đi và có 48 bít được chọn ra từ 56 bít của khoá. Nửa phải của dữ liệu được mở rộng thành 48 bít bằng một phép hoán vị mở rộng, tiếp đó khối 48 bít này được kết hợp với khối 48 bít đã được thay đổi và hoán vị của khoá bằng toán tử XOR. Khối kết quả của phép tính XOR được lựa chọn ra 32 bít bằng cách sử dụng thuật toán thay thế và hoán vị lần nữa. Đó là bốn thao tác tạo nên hàm *f*. Tiếp đó, đầu ra của hàm *f* được kết hợp với nửa trái bằng một toán tử XOR. Kết quả của các bước thực hiện này trở thành nửa phải mới; nửa phải cũ trở thành nửa trái mới. Sự thực hiện này được lặp lại 16 lần, tạo thành 16 vòng của DES (hình 4.7).

Nếu B_i là kết quả của vòng thứ i, L_i và R_i là hai nửa trái và phải của B_i, K_i là khoá 48 bít của vòng thứ i, và *f* là hàm thực hiện thay thế, hoán vị và XOR với khoá, ta có biểu diễn của một vòng sẽ như sau:

$L_i=R_{i-1}$

$R_i=L_{i-1}$ XOR $f(R_{i-1},K_i)$



Hình 4.8. Một vòng lặp DES

Chi tiết về thuật toán DES được trình bày trong các tài liệu về Lý thuyết mật mã, đề nghị bạn đọc tự tìm tài liệu nghiên cứu thêm.

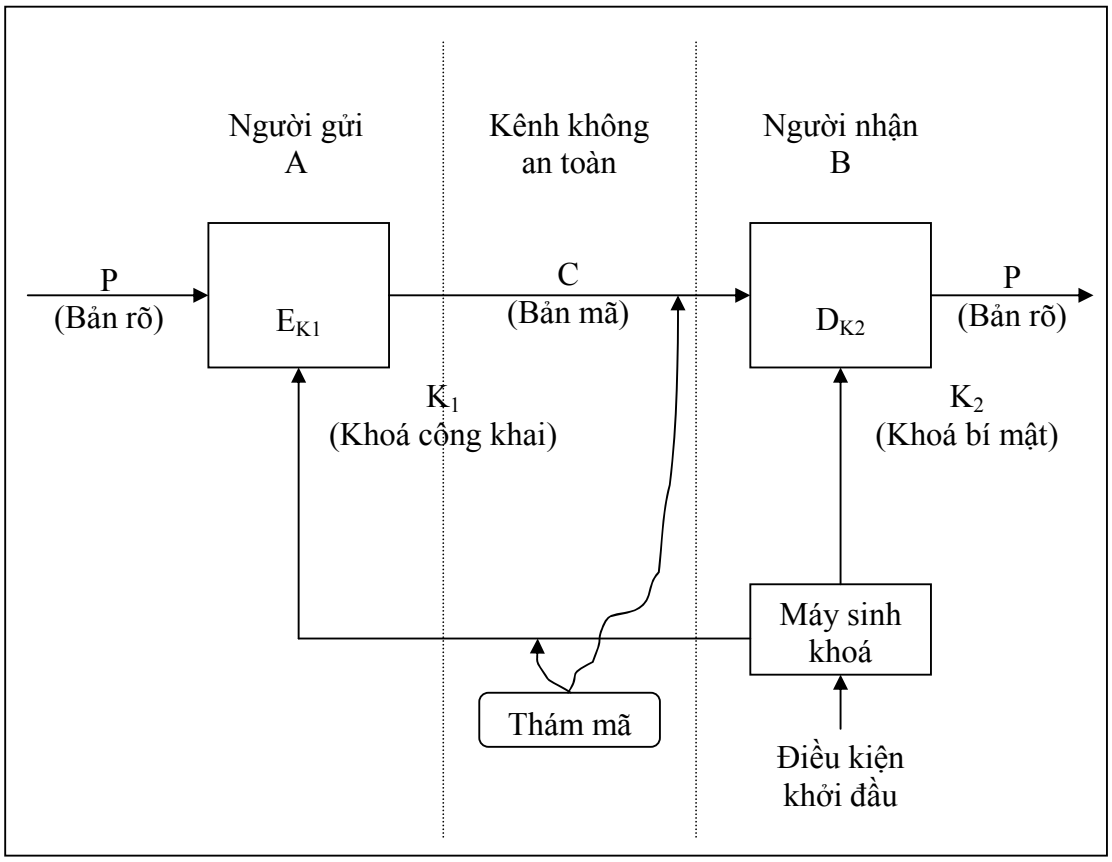
4.3.3.4. Phương pháp sử dụng khóa công khai (Public key)

Đối với các hệ mã hoá cổ điển thì nếu biết khoá mã E_k thì cũng biết được khoá giải D_k. Nói một cách cụ thể thì khoá giải có thể suy ra trực tiếp từ khoá mã hoặc có thể tính toán từ khoá mã không mấy khó khăn, và ngược lại.

Trong các lĩnh vực ứng dụng thương mại như chuyển ngân điện tử và thư điện tử thì bài toán phân phối khoá là một vấn đề lớn đối với các hệ mã hoá cổ điển. Khuynh hướng cung cấp các khoá dài mà nó phải được thay đổi thường xuyên, trong khi vẫn tiếp tục duy trì cả tính an toàn lẫn hiệu quả chi phí sẽ cản trở rất nhiều đến việc phát triển các hệ thống như vậy. Tuy nhiên các phương pháp gần đây được phát triển sẽ hứa hẹn việc loại bỏ hoàn toàn bài toán phân phối khoá. Những hệ thống mã hoá như vậy được gọi là các hệ mã hoá công khai.

Ý tưởng về mã hoá công khai (mã hoá phi đối xứng - đối lập với các hệ mã hoá cổ điển) thuộc về Whitfield Diffie và Martin Hellman. Diffie và Hellman lần đầu tiên đưa ra ý tưởng này vào năm 1976.

Thuật toán mã hoá công khai sử dụng khoá để mã hóa và khoá để giải mã là khác nhau. Các khoá này tạo thành một cặp chuyển đổi ngược nhau. Việc biết khoá mã E_k không nhất thiết làm lộ D_k. Cụ thể hơn người thám mã có thể biết E_k và do đó có thể tính được D_k, tuy nhiên việc tính D_k từ E_k là không khả thi với hầu hết các khoá k, bởi vì việc đó phải mất đến hàng tỷ năm. Do vậy trên thực tế bản mã C vẫn được giữa an toàn mặc dù khoá mã E_k được công bố rộng dãi. Khoá mã có thể công khai nhưng khoá giải phải được giữ bí mật.



Hình 4.9. Mã hoá công khai

Mặc dù khoá mã (khoá công khai) và khoá giải (khoá bí mật) thực hiện các thao tác ngược nhau và do đó có liên quan đến nhau nhưng phải làm sao để không thể suy ra khoá riêng từ khoá công khai. Yêu cầu đó có thể đạt được nhờ các hàm toán học đặc biệt gọi là các hàm sập bẫy một chiều (trapdoor one-way function). Các hàm này có đặc điểm là không thể chỉ dựa vào mô tả cả hàm mà còn phải biết được cách xây dựng hàm thì mới có thể suy ra được nghịch đảo của nó.

Một số hệ mã hoá công khai:

1. Hệ Merkle-Hellman Knapsack
2. Hệ Rivest-Shamir-Adleman (RSA)
3. Hệ EL Gamal và Digital Signature Algrithms

Các hệ mã hoá công khai này đều dựa trên cơ sở những vấn đề phức tạp thuộc lĩnh vực lý thuyết số, đó là các thuật toán số học được thực hiện trên các số nguyên tố rất lớn.

Thuật toán mã hoá RSA

RSA là tên viết tắt từ tên các tác giả của nó Ron Rivest, Adi Shamir, Leonard Adleman - những người đầu tiên giới thiệu thuật toán này năm 1978.

Bí mật của mã RSA là ở việc rất khó khăn khi phân tích ra thừa số nguyên tố của các số lớn. Khoá công cộng và khoá riêng là các hàm số của một cặp số nguyên tố rất lớn nào đó (có từ 100 đến 200 chữ số, thậm chí có thể lớn hơn). Việc giải mã từ một khoá và thông điệp đã mã hoá là tương đương với việc phân tích một số rất lớn ra thừa số nguyên tố.

Để tổng hợp hai khoá, ta phải chọn hai số nguyên tố lớn p và q. Sau đó tính: $n = p*q$ (p, q được giữ bí mật).

Chọn ngẫu nhiên một khoá mã, e, sao cho e và $(p-1)(q-1)$ là nguyên tố cùng nhau. Sau đó dùng thuật toán Euclid để tính khoá giải mã d thoả mãn:

$$e*d = 1 \pmod{(p-1)(q-1)}$$ hay nói cách khác $$d = e^{-1} \pmod{(p-1)(q-1)}$$

Ta cần chú ý rằng d và n cũng nguyên tố cùng nhau. Số e và n chính là khoá công cộng còn d là khoá bí mật. Số p và q không còn cần thiết nữa, chúng có thể bỏ đi nhưng không bao giờ được lộ ra.

Để mã hoá một bản rõ m, trước tiên ta phải chuyển các chữ cái thành các số tương ứng và chia nhỏ m thành các khối với độ dài của mỗi khối đều nhỏ hơn n. Với dữ liệu nhị phân, chọn luỹ thừa của 2 số lớn nhất mà vẫn nhỏ hơn n. Như vậy nếu p và q là số nguyên tố 100 chữ số thì n có khoảng 200 chữ số, và mỗi khối bản rõ, m_i, phải nhỏ hơn 200 chữ số.

Thông điệp đã mã hoá c sẽ được tạo ra từ các khối đã mã hoá c_i có kích cỡ hay độ dài tương tự.

Công thức mã hoá đơn giản là:

$$c_i = m_i^e \pmod{n}$$

Quá trình giải mã làm ngược lại. Để giải mã một thông điệp, ta lấy mỗi khối đã mã hoá c_i và tính:

$m_i = c_i^d \pmod n$

Thật vậy:

$c_i^d = (m_i^e)^d = m_i^{ed} = m_i^{k(p-1)(q-1)+1} = m_i.m_i^{k(p-1)(q-1)} = m_i.1 = m_i$

(tất cả lấy theo mod n).

Ta có thể tổng kết theo bảng sau:

KHOÁ MÃ HOÁ (PUBLIC KEY):

n: là tích của 2 số nguyên tố, p và q (p và q là bí mật chính)

e: là số nguyên tố cùng nhau với $(p-1)(q-1)$

KHOÁ GIẢI MÃ (PRIVATE KEY):

d: $e^{-1} \pmod{(p-1)(q-1)}$

MÃ HOÁ (ENCRYTING):

$c = m^e \pmod n$

GIẢI MÃ (DECRYPTING)

$m = c^d \pmod n$

Sau đây là một ví dụ ngắn để minh hoạ rõ thuật toán mã hoá RSA. Giả sử ta chọn $p = 47$ và $q = 71$ thì:

$n = p.q = 3337$

Số e phải nguyên tố cùng nhau với giá trị:

$(p-1)(q-1) = 46.70 = 3220$

Chọn e (một cách ngẫu nhiên) là 79. Từ đó ta suy ra d:

$d = 79^{-1} \pmod{3220} = 1019$

Số này tính được nhờ thuật toán Euclid mở rộng. e và n là khoá công cộng và d là khoá riêng, p và q được bỏ đi.

Để mã hoá một bản rõ nào đó, ví dụ:

$m = 6882326879666683$

Đầu tiên ta chia bản rõ đó thành các khối nhỏ. Ta chọn cách chia thành các khối nhỏ có 3 chữ số là tốt nhất. Như vậy ta có tất cả 6 khối bản rõ:

$m_1 = 688$

$m_2 = 232$

$m_3 = 687$

$m_4 = 966$

$m_5 = 668$

Khối đầu tiên được mã hoá thành:

$688^{79} \pmod{3337} = 1570 = c_1$

Tương tự thực hiện trên các khối khác, sau đó tổng hợp lại ta sẽ được bản rõ hoàn chỉnh:

c = 1570 2756 2714 2276 2423 158

Việc giải mã tiến hành tương tự với khoá giải mã d =1019.

Ví dụ:

$1570^{1019} \pmod{3337} = 688 = m_1$

4.3.3.5. So sánh các phương pháp mật mã

Để đánh giá một giải thuật mã hoá cần dựa vào các yếu tố như độ phức tạp, thời gian mã hoá và vấn đề phân phối khoá trong môi trường nhiều người sử dụng.

Các phương pháp mã hoá cổ điển như phương pháp đổi chỗ và thay thế là đơn giản và được sử dụng sớm nhất. Nhược điểm của chúng là độ an toàn không cao do thường không đạt được độ phức tạp cần thiết đồng thời rất dễ bị lộ khoá do cả người gửi lẫn người nhận đều phải biết khoá.

Còn các phương pháp mã hoá sử dụng khoá công khai (như RSA) mặc dù khắc phục được vấn đề phân phối khoá song lại có chi phí cao và khá chậm chạp. Trong khi thời gian mã hoá của phương pháp DES chỉ đòi hỏi vài micro giây thì phương pháp RSA lại đòi hỏi tới vài mili giây, do đó hạn chế thông lượng ở mức 50 Kb/s.

Hiện nay nhìn chung phương pháp mã hoá DES được sử dụng rộng rãi nhất, tuy người ta vẫn còn bàn cãi xem độ dài của khoá 48 bít đã đủ chưa và các phép thay thế đã đủ phức tạp chưa để đạt được độ an toàn thông tin mong muốn.

Một số bài tập

1. Cho xâu ký tự: “KINH TE THI TRUONG”, hãy dùng phương pháp đổi chỗ theo mẫu hình học để mã hoá với ma trận có kích thước 3x5, và hoán vị cột là 1,3,5,4,2.
2. Cho xâu ký tự: “HOI NHAP VAO WTO DE PHAT TRIEN”, hãy dùng phương pháp đổi chỗ cột để mã hoá với ma trận có kích thước 4x6, và hoán vị cột là 3,1,2,5,4,6.
3. Cho xâu ký tự: “KINH TE NUOC TA DANG TANG TRUONG MANH”, hãy dùng phương pháp hoán vị theo chu kỳ cố định để mã hoá, cho chu kỳ d = 6, hoán vị trong mỗi chu kỳ là 1,5,2,4,6,3.
4. Cài đặt các thuật toán mã hoá, viết hàm giải mã cho các phương pháp mã hoá trên.

4.5.1. Khái niệm hiệu năng và các độ đo hiệu năng mạng

Khái niệm hiệu năng mạng

Theo nghĩa chung, hiệu năng là một độ đo công việc mà một hệ thống thực hiện được. Hiệu năng chủ yếu được xác định bởi sự kết hợp của các nhân tố: tính sẵn sàng để dùng (availability), thông lượng (throughput) và thời gian đáp ứng (response time). Đối với mạng máy tính, hiệu năng cũng còn được xác định dựa trên các nhân tố khác nữa, thí dụ: thời gian trễ (delay), độ tin cậy (reliability), tỉ suất lỗi (error rate), hiệu năng của ứng dụng v.v.

Tuỳ theo mục đích nghiên cứu cụ thể, hiệu năng có thể chỉ bao gồm một nhân tố nào đó hoặc là sự kết hợp một số trong các nhân tố nêu trên.

Các độ đo hiệu năng mạng

Có thể phân các độ đo hiệu năng thành hai loại: các độ đo hướng tới người sử dụng và các độ đo hướng tới hệ thống. Trong các độ đo hướng tới người sử dụng, *thời gian đáp ứng* (response time) thường được sử dụng trong các hệ thời gian thực hoặc các môi trường hệ thống tương tác. Đó là khoảng thời gian từ khi có một yêu cầu (request) đến hệ thống cho đến khi nó được hệ thống thực hiện xong. Trong các hệ thống tương tác, đôi khi người ta sử dụng độ đo *thời gian phản ứng của hệ thống* (system reaction time) thay cho thời gian đáp ứng. Đó là khoảng thời gian tính từ khi input đến hệ thống cho đến khi yêu cầu chứa trong input đó nhận được khe thời gian (slice time) phục vụ đầu tiên. Độ đo này đo mức độ hiệu dụng của bộ lập lịch của hệ thống trong việc nhanh chóng cung cấp dịch vụ cho một yêu cầu mới đến. Trong các hệ thống mạng máy tính, các đại lượng thời gian đáp ứng, thời gian phản ứng của hệ thống đều được xem là các biến ngẫu nhiên, vì vậy người ta thường nói về phân bố, kỳ vọng, phương sai... của chúng.

Các độ đo hướng tới hệ thống điển hình là *thông lượng* (throughput) và *thời gian trễ* (delay time, delay). Thông lượng được định nghĩa là số đơn vị thông tin tính trung bình được vận chuyển qua mạng trong một đơn vị thời gian. Đơn vị thông tin ở đây có thể là bit, byte hay gói số liệu... Nếu các đơn vị thông tin đi vào mạng theo một cơ chế độc lập với trạng thái của mạng, thì thông lượng cũng chính bằng tốc độ đến trung bình nếu mạng vẫn còn có khả năng vận chuyển, không dẫn đến trạng thái bị tắc nghẽn. Một số trường hợp người ta sử dụng đại lượng không thứ nguyên *Hệ số sử dụng đường truyền* (Line Utilization) hay còn gọi *thông lượng chuẩn hoá*, đó là tỉ số của thông lượng trên năng lực vận chuyển của đường truyền (line capacity). Thời gian trễ là thời gian trung bình để vận chuyển một gói số liệu qua mạng, từ nguồn tới đích. Cũng có trường hợp người ta sử dụng đại lượng *thời gian trễ chuẩn hoá*, đó là tỉ số của thời gian trễ trên một tham số thời gian nào đó, thí dụ thời gian cần thiết để truyền một gói tin (packet transmition time).

4.5.2. Tầm quan trọng của việc đánh giá hiệu năng mạng máy tính

Trong suốt lịch sử tiến hoá của mạng máy tính, vấn đề đánh giá và dự đoán hiệu năng mạng luôn thu hút sự quan tâm của những người nghiên cứu và thiết kế mạng; mục đích chính là ***để nắm được và cải thiện đặc trưng giá - hiệu năng (cost-performance)***. Yêu cầu đánh giá và dự đoán hiệu năng mạng đặt ra ngay từ khi người ta thiết kế kiến trúc của hệ thống cho đến khi mạng đã được lắp đặt và đưa vào hoạt động. Trong giai đoạn đầu của quá trình thiết kế, người ta thường phải dự đoán hai điều. Thứ nhất là bản chất của các ứng dụng sẽ chạy trên mạng và các yêu cầu dịch vụ mà các ứng dụng này đòi hỏi hệ thống mạng phải đáp ứng. Điều dự đoán thứ hai liên quan tới việc lựa chọn một trong các thiết kế kiến trúc, dựa trên các công nghệ phần cứng và phần mềm sẽ được phát triển và đưa ra thị trường trong tương lai, khi hệ thống mạng bước vào giai đoạn triển khai thực hiện.

Sau khi đã lựa chọn kiến trúc và bắt đầu thiết kế, triển khai hệ thống mạng, việc dự đoán và đánh giá hiệu năng sẽ trở nên cụ thể hơn. Thí dụ sẽ chọn đường truyền vật lý như thế nào, các đặc tính của đường truyền được chọn sẽ ảnh hưởng thế nào đến hiệu năng của mạng. Các kỹ thuật được dùng để dự đoán và đánh giá hiệu năng mạng trong giai đoạn thiết kế và triển khai thực hiện có khi chỉ là các tính toán bằng tay, nhưng cũng có khi là các mô phỏng rất tinh vi. Việc so sánh hiệu năng dự đoán với hiệu năng thực tế đạt được thường giúp cho nhà nghiên cứu thấy được các khiếm khuyết chính trong thiết kế hoặc các lỗi trong việc lập trình hệ thống. Ngày nay, việc dự đoán và đánh giá hiệu năng thường được người ta coi là một phần không thể thiếu được của công việc thiết kế và triển khai thực hiện hệ thống.

Định cấu hình mạng: Sau khi mạng đã được triển khai thực hiện, việc dự đoán và đánh giá hiệu năng mạng đối với các ứng dụng cụ thể cũng có ý nghĩa quan trọng. Nhằm đạt được sự tối ưu hoá, nhà sản xuất phải chỉ ra được các cách kết hợp và tổ chức phần cứng và phần mềm mạng để đem lại một giải pháp tốt nhất cho các yêu cầu của khách hàng, việc này thường được gọi là định cấu hình mạng. Mặc dù có thể vẫn sử dụng các công cụ và phương pháp đã được sử dụng trong giai đoạn phát triển hệ thống, nhưng cần phải bổ sung thêm một số yếu tố nữa. Đặc điểm môi trường của người sử dụng sản phẩm mạng cần được biểu diễn bằng các tham số định lượng và đưa vào mô hình mô phỏng hiệu năng.

Tinh chỉnh hệ thống: Sau khi hệ thống sản phẩm đã được lắp đặt tại địa điểm của khách hàng, nhà cung cấp sản phẩm cần phải làm sao cho hệ thống mà họ bán cho khách hàng đạt được hiệu năng hoạt động như họ đã hứa hẹn khi chào hàng, việc này được gọi là tinh chỉnh hệ thống. Đối với các hệ thống mạng, việc tìm ra được điểm làm việc tối ưu và ổn định trên toàn mạng là rất khó.

4.5.3. Các phương pháp đánh giá hiệu năng mạng

Có nhiều phương pháp đánh giá hiệu năng mạng máy tính, có thể chia chúng làm ba loại: mô hình Giải tích (Analytic Models), mô hình Mô phỏng (Simulation Models) và Đo hiệu năng (Measurement).

Mô hình Giải tích

Trong các mạng chuyển mạch gói, gói số liệu là các khối dữ liệu có chiều dài thay đổi được, được truyền qua mạng từ nguồn tới đích theo một con đường nào đó do hệ thống mạng quyết định. Các tài nguyên mạng sẽ được chia sẻ giữa các gói số liệu khi chúng đi qua mạng. Số lượng và chiều dài các gói số liệu đi vào hoặc đi qua mạng tại mọi thời điểm, thời gian kéo dài các cuộc kết nối v.v., tất cả các tham số này nói chung, thay đổi một cách thống kê. Vì vậy, để nêu ra các tiêu chuẩn đo lường định lượng về hiệu năng, cần phải sử dụng các khái niệm về xác suất để nghiên cứu sự tương tác của chúng với mạng. Lý thuyết Hàng đợi đóng vai trò mấu chốt trong việc phân tích mạng, bởi vì đó là công cụ Toán học thích hợp nhất để phát biểu và giải các bài toán về hiệu năng. Theo phương pháp này, chúng ta viết ra các mối quan hệ hàm giữa các tiêu chuẩn hiệu năng cần quan tâm và các tham số của hệ thống mạng bằng các phương trình có thể giải được bằng giải tích.

Mô phỏng

Theo nghĩa chung nhất, mô phỏng là sự bắt chước một hay nhiều khía cạnh của sự vật có thực, bằng một cách nào đó càng giống càng tốt. Trong các lĩnh vực nghiên cứu hiện đại, như lĩnh vực đánh giá hiệu năng mạng, mô phỏng được hiểu là một kỹ thuật sử dụng máy tính điện tử số để làm các thí nghiệm về mạng có liên quan đến thời gian. Mô hình Mô phỏng mô tả hành vi động của mạng, ngay cả khi người nghiên cứu chỉ quan tâm đến giá trị trung bình của một số độ đo trong trạng thái dừng. Cấu trúc và độ phức tạp của bộ mô phỏng phụ thuộc vào phạm vi của thí nghiệm mô phỏng. Nó thường được xây dựng có cấu trúc, cho phép mô-đun hoá chương trình mô phỏng thành tập các chương trình con, sao cho việc sửa đổi, bổ sung các chương trình con được dễ dàng. Ngoài ra, chương trình mô phỏng cũng phải được xây dựng sao cho đạt được tốc độ cao nhằm làm giảm thời gian chạy mô phỏng càng nhiều càng tốt.

Đo

Đó là phương pháp xác định hiệu năng dựa trên việc đo trên mạng thực các tham số mạng cấu thành độ đo hiệu năng cần quan tâm. Việc đo hiệu năng nhằm thực hiện một trong các nhiệm vụ sau. Một là, giám sát hiệu năng của mạng . Hai là, thu thập số liệu để lập mô hình dữ liệu vào cho các phương pháp đánh giá hiệu năng bằng giải tích hoặc mô phỏng. Nhiệm vụ thứ ba là kiểm chứng các mô hình khác dựa trên các số liệu đo được. Đo hiệu năng không chỉ quan trọng trong các giai đoạn triển khai thực hiện và tích hợp hệ thống mà còn cả trong các giai đoạn lắp đặt và vận hành hệ thống. Bởi vì sau khi lắp đặt và đưa vào sử dụng, mỗi một hệ thống cụ thể sẽ có một tải hệ thống và các độ đo hiệu năng được quan tâm riêng của nó, cho nên sau khi lắp đặt, người ta thường phải điều chỉnh cấu hình cho phù hợp. Các tham số cấu hình sẽ được chọn sau khi các phép đo hiệu năng cho thấy các tham số cấu hình này làm cho hệ thống đạt được hiệu năng tốt nhất. Trong thực tế, mọi người đều thừa nhận tầm quan trọng của việc đo và đánh giá hiệu năng. Chúng ta có thể thấy rõ điều này qua việc, hầu như tất cả các hệ thống mạng đều tích hợp bên trong nó các công cụ đo và đánh giá hiệu năng; nhờ đó có thể đo hiệu năng bất cứ lúc nào trong suốt vòng đời của hệ thống.

So sánh các ph ng pháp ánh giá hiệ ng

Mô hình Giải tích: Nếu có thể sử dụng mô hình Giải tích thì đó là điều tốt nhất, bởi vì chúng ta có thể thay đổi các tham số hệ thống và cấu hình mạng trong một miền rộng với chi phí thấp mà vẫn có thể đạt được các kết quả mong muốn. Tuy nhiên, các mô hình Giải tích mà chúng ta xây dựng thường là không thể giải được nếu không được đơn giản hoá nhờ các giả thiết, hoặc được phân rã thành các mô hình nhiều cấp. Các mô hình giải được thường rất đơn giản hoặc khác xa thực tế, cho nên phương pháp này thường chỉ được sử dụng ngay trong giai đoạn đầu của việc thiết kế mạng, giúp cho người thiết kế dự đoán được các giá trị giới hạn của hiệu năng. Ngoài ra, các kết quả của phương pháp này bắt buộc phải được kiểm nghiệm bằng kết quả của các phương pháp khác, như mô phỏng hoặc đo.

Mô phỏng: Trong những trường hợp mô hình Giải tích mà chúng ta nhận được, dù đã được đơn giản hoá, hoặc phân rã nhưng vẫn không thể giải được bằng Toán học, khi đó, nói chung, chúng ta sẽ chỉ còn một phương pháp là mô phỏng. Phương pháp mô phỏng có thể được sử dụng ngay trong giai đoạn đầu của việc thiết kế hệ thống mạng, cho đến giai đoạn triển khai thực hiện và tích hợp hệ thống. Phương pháp này nói chung, đòi hỏi một chi phí rất cao cho việc xây dựng bộ mô phỏng cũng như kiểm chứng tính đúng đắn của nó. Tuy nhiên, sau khi đã xây dựng xong bộ mô phỏng, người nghiên cứu có thể tiến hành chạy chương trình mô phỏng bao nhiêu lần tuỳ ý, với độ chính xác theo yêu cầu và chi phí cho mỗi lần chạy thường là rất thấp. Các kết quả mô phỏng nói chung vẫn cần được kiểm chứng, bằng phương pháp giải tích hoặc đo, nhất là bằng phương pháp đo. Phương pháp mô hình Giải tích và mô hình Mô phỏng đóng vai trò rất quan trọng trong việc thiết kế và triển khai thực hiện hệ thống, đặc biệt là ở giai đoạn đầu.

Đo: Phương pháp đo chỉ có thể thực hiện được trên mạng thực, đang hoạt động, nó cũng đòi hỏi chi phí cho các công cụ đo và cho việc tiến hành đo. Việc đo cần được tiến hành tại nhiều điểm trên mạng thực, ở những thời điểm khác nhau và cần lặp đi lặp lại trong một khoảng thời gian đủ dài, thậm chí có thể dài đến hàng tháng. Ngoài ra, người nghiên cứu phải có kiến thức về Lý thuyết thống kê thì mới có thể rút ra được các kết luận hữu ích từ các số liệu thu thập được. Mặc dầu vậy, bằng phương pháp đo có thể vẫn không phát hiện ra được hoặc dự đoán được các hành vi đặc biệt của mạng.

4.6. CÂU HỎI VÀ BÀI TẬP

(đang tiếp tục bổ sung)

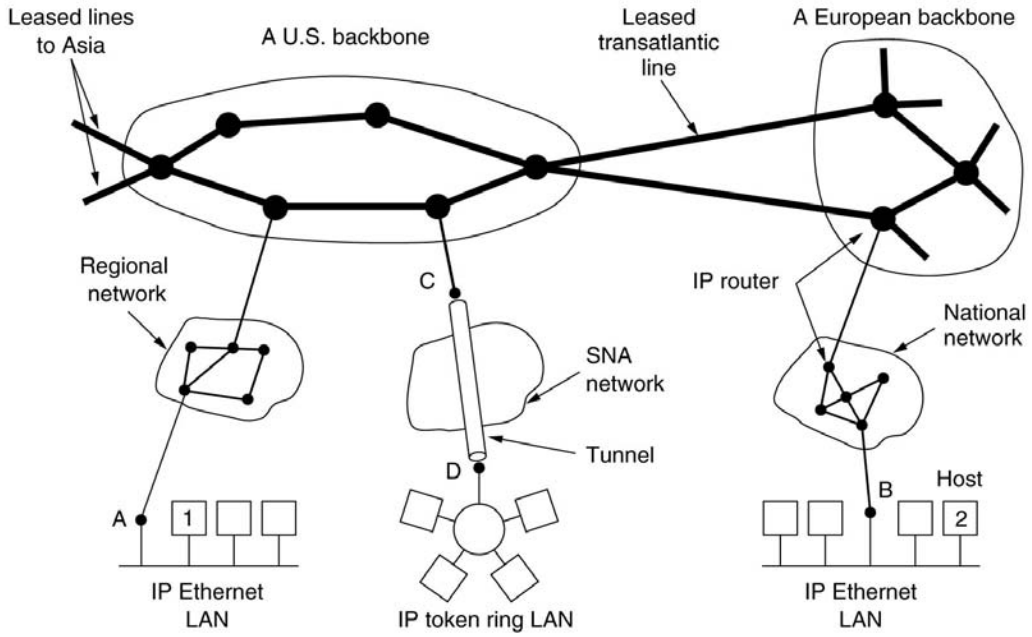
CHƯƠNG 5. TCP/IP VÀ INTERNET

5.1. GIỚI THIỆU CHUNG VỀ INTERNET

5.1.1. Lịch sử phát triển của mạng Internet và bộ giao thức TCP/IP

Tiền thân của mạng Internet là ARPANET, xuất phát từ một mạng thí nghiệm được Robert L.G đề xuất vào năm 1967. Cơ quan quản lý dự án nghiên cứu phát triển ARPA thuộc Bộ Quốc phòng Mỹ đã liên kết mạng tại 4 địa điểm đầu tiên vào tháng 7 năm 1968 bao gồm: Viện nghiên cứu Stanford, Đại học tổng hợp California ở Los Angeles, Đại học tổng hợp Utah và Đại học tổng hợp California ở Santa Barbara (UCSB). Đó chính là mạng liên khu vực (WAN) đầu tiên được xây dựng.

Năm 1983, giao thức TCP/IP chính thức được coi như một chuẩn đối với ngành quân sự Mỹ và tất cả các máy tính nối với ARPANET phải sử dụng chuẩn mới này. ARPANET phát triển rất nhanh, mọi trường đại học đều muốn gia nhập, việc quản lý mạng trở nên khó khăn. Vì vậy, năm 1984, ARPANET được chia ra thành hai phần: phần thứ nhất cho các địa điểm quân sự, được gọi là MILNET; phần thứ hai là một ARPANET mới, cho các địa điểm phi quân sự, dành cho việc nghiên cứu và phát triển. Tuy nhiên hai mạng này vẫn được liên kết với nhau nhờ giao thức liên mạng IP.



Hình 5.1. Mạng Internet trải rộng toàn cầu

Giao thức TCP/IP ngày càng thể hiện rõ các điểm mạnh của nó, quan trọng nhất là khả năng liên kết các mạng khác với nhau một cách dễ dàng. Chính điều này cùng với các chính sách mở cửa đã cho phép các mạng dùng cho nghiên cứu và thương mại kết nối được với ARPANET, thúc đẩy việc tạo ra một siêu mạng (SuperNetwork).

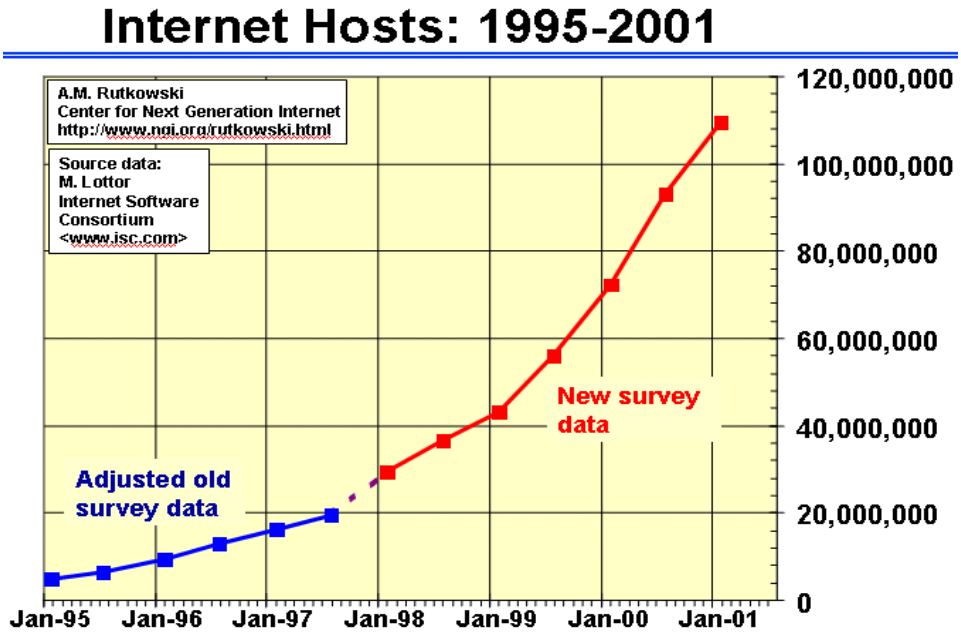
Mốc lịch sử quan trọng của Internet được xác lập vào giữa thập kỷ 80 khi Hội đồng Khoa học Quốc gia Mỹ NSF (National Science Foundation) thành lập mạng liên kết các trung tâm máy tính lớn với nhau gọi là NSFNET. Nhiều doanh nghiệp đã chuyển từ ARPANET sang NSFNET và do đó sau gần 20 năm hoạt động, ARPANET không còn hiệu quả đã ngừng hoạt động vào khoảng năm 1990.

Sự phát triển của mạng xương sống NSFNET và những mạng vùng khác đã tạo ra một môi trường thuận lợi cho sự phát triển của Internet. Đến năm 1995, NSFNET thu lại thành một mạng nghiên cứu còn Internet thì vẫn tiếp tục phát triển.

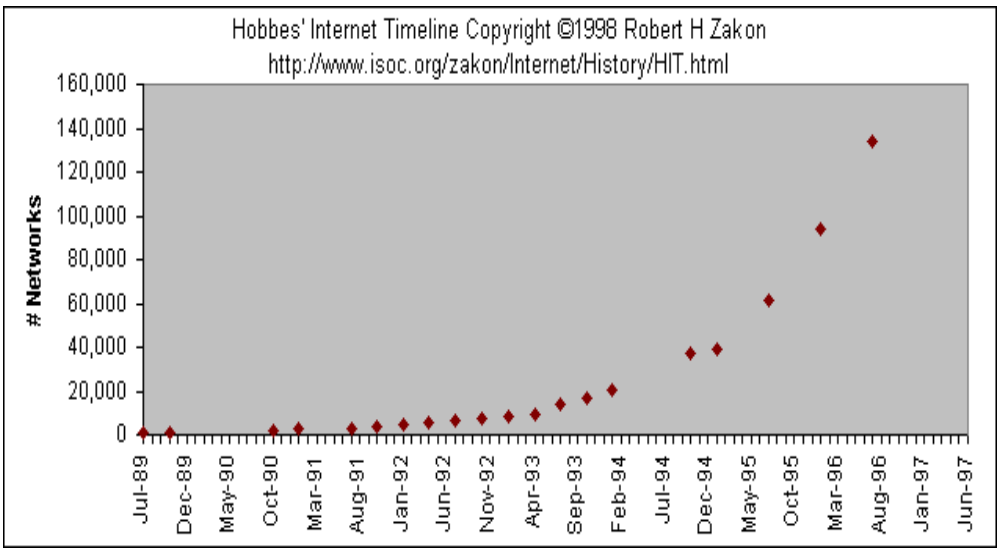
Với khả năng kết nối mở, Internet đã trở thành một mạng lớn nhất trên thế giới, mạng của các mạng, xuất hiện trong mọi lĩnh vực thương mại, chính trị, quân sự, nghiên cứu, giáo dục, văn hoá, xã hội... Cũng từ đó các dịch vụ trên Internet không ngừng phát triển. Ngày nay khi cơ sở hạ tầng của mạng Internet được nâng cao (đặc biệt là về băng thông) đã làm cho nhu cầu của các ứng dụng đa phương tiện qua mạng tăng lên nhanh chóng.

5.1.2. Sự tăng trưởng của Internet

Các biểu đồ dưới đây (hình 5.2 và 5.3) cho ta thấy được tốc độ tăng trưởng mẽ của số lượng máy (hosts) và mạng (networks) kết nối vào Internet.



Hình 5.2. Sự tăng trưởng thuê bao Internet từ 1995-2001



Hình 5.3. Sự tăng lên của các mạng kết nối vào Internet từ 1989-1997

5.1.3. Các yếu tố thúc đẩy sự tăng trưởng của Internet

- Sử dụng TCP/IP trong Unix, hệ điều hành được sử dụng phổ biến từ 1983.
- PC ra đời năm 1980, sức mạnh tính toán ngày càng cao, trong khi giá ngày càng rẻ.
- NSFNET - mạng xương sống của Mỹ, tốc độ cao, ra đời năm 1986.
- Hệ thống tên miền ra đời làm cho việc truy nhập Internet trở nên đơn giản và thuận tiện.
- Kiến trúc đơn giản của bộ giao thức và tính linh hoạt của Internet.
- Các ứng dụng trên Internet không ngừng phát triển, đáp ứng nhu cầu đa dạng của người dùng

5.2. KIẾN TRÚC MẠNG INTERNET

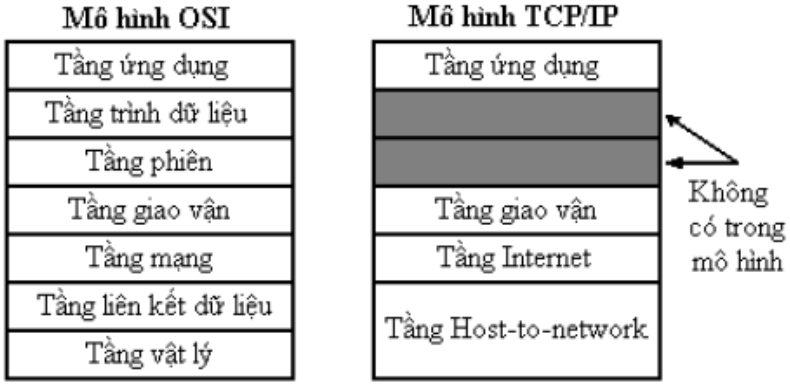
5.2.1. Mô hình TCP/IP

Giao thức TCP/IP được phát triển từ mạng ARPANET và Internet và được dùng như giao thức mạng và giao vận trên mạng Internet. TCP (Transmission Control Protocol) là giao thức thuộc tầng giao vận 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.

Các tầng của mô hình tham chiếu TCP/IP

Bộ quốc phòng Mỹ gọi tắt là DoD (Department of Defense) đã tạo ra mô hình tham chiếu TCP/IP vì muốn một mạng có thể tồn tại trong bất cứ điều kiện nào, ngay cả khi có chiến tranh hạt nhân. DoD muốn các gói dữ liệu xuyên suốt mạng vào mọi lúc, dưới bất cứ điều kiện nào, từ bất cứ một điểm đến một điểm khác. Đây là một bài toán thiết kế cực kỳ khó khăn mà từ đó làm nảy sinh ra mô hình TCP/IP, vì vậy đã trở thành chuẩn Internet để phát triển.



Hình 5.4. So sánh mô hình OSI và mô hình TCP/IP

Tầng ứng dụng

Các nhà thiết kế TCP/IP cảm thấy rằng các giao thức mức cao nên bao gồm các tầng trình bày và tầng phiên. Để đơn giản, họ tạo ra một tầng ứng dụng kiểm soát các giao thức mức cao, các vấn đề của tầng Trình diễn, mã hoá và điều khiển hội thoại. TCP/IP tập hợp tất cả các vấn đề liên quan đến ứng dụng vào trong một tầng, và đảm bảo dữ liệu được đóng gói một cách thích hợp cho tầng kế tiếp.

Tầng Giao vận

Tầng vận chuyển đề cập đến các vấn đề chất lượng dịch vụ như độ tin cậy, điều khiển luồng và sửa lỗi. Một trong các giao thức của nó là TCP, TCP cung cấp các phương thức linh hoạt và hiệu quả để thực hiện các hoạt động truyền dữ liệu tin cậy, hiệu xuất cao và ít lỗi. TCP là giao thức có tạo cầu nối (connection-oriented). Nó tiến hành hội thoại giữa nguồn và đích trong khi bọc thông tin tầng ứng dụng thành các đơn vị gọi là segment.Tạo cầu nối không có nghĩa là tồn tại một mạch thực sự giữa hai máy tính, thay vì vậy nó có nghĩa là các segment của tầng 4 di chuyển tới và lui giữa hai host để công nhận kết nối tồn tại một cách luận lý trong một khoảng thời gian nào đó. Điều này coi như chuyển mạch gói (packet switching).

Tầng Internet

Mục tiêu của tầng Internet là truyền các gói tin bắt nguồn từ bất kỳ mạng nào trên liên mạng và đến được đích trong điều kiện độc lập với đường dẫn và các mạng mà chúng đã trải qua. Giao thức đặc trưng khống chế tầng này được gọi là IP. Công việc xác định đường dẫn tốt nhất và hoạt động chuyển mạch gói diễn ra tại tầng này.

Tầng truy xuất mạng

Tên của tầng này có nghĩa khá rộng và có phần hơi rối rắm. Nó cũng được gọi là tầng host-to-network. Nó là tầng liên quan đến tất cả các vấn đề mà một gói IP yêu cầu

để tạo một liên kết vật lý thực sự, và sau đó tạo một liên kết vật lý khác. Nó bao gồm các chi tiết kỹ thuật LAN và WAN, và tất cả các chi tiết trong tầng liên kết dữ liệu cũng như tầng vật lý của mô hình OSI.

Mô hình TCP/IP hướng đến tối đa độ linh hoạt tại tầng ứng dụng cho người phát triển phần mềm. Tầng vận chuyển liên quan đến hai giao thức TCP và UDP (User Datagram Protocol). Tầng cuối cùng, tầng truy xuất mạng liên kết đến các kỹ thuật LAN hay WAN đang được dùng.

Trong mô hình TCP/IP không cần quan tâm đến ứng dụng nào yêu cầu các dịch vụ mạng, và không cần quan tâm đến giao thức vận chuyển nào đang được dùng, chỉ có một giao thức mạng IP. Đây là một quyết định thiết kế có cân nhắc kỹ. IP phục vụ như một giao thức đa năng cho phép bất kỳ máy tính nào, ở bất cứ đâu, truyền dữ liệu vào bất cứ thời điểm nào.

So sánh mô hình OSI và mô hình TCP/IP

Các điểm giống nhau:

- Cả hai đều theo kiến trúc phân tầng.
- Cả hai đều có tầng ứng dụng, qua đó chúng có nhiều dịch vụ khác nhau.
- Cả hai có các tầng mạng và tầng vận chuyển có thể so sánh được.
- Kỹ thuật chuyển mạch gói được chấp nhận.

Các điểm khác nhau:

- TCP/IP tập hợp các tầng trình bày và tầng phiên vào trong tầng ứng dụng của nó.
- TCP/IP tập hợp tầng vật lý và tầng liên kết dữ liệu trong OSI vào một tầng.
- TCP/IP biểu hiện đơn giản hơn vì có ít tầng hơn.

Các giao thức TCP/IP là các chuẩn cơ sở cho Internet phát triển, như vậy mô hình TCP/IP chiếm được niềm tin chỉ vì các giao thức của nó. Ngược lại, các mạng thông thường không được xây dựng dựa trên OSI, ngay cả khi OSI dùng như một hướng dẫn.

5.2.2. Họ giao thức TCP/IP

Bao gồm 2 phần chính (hình 5.3):

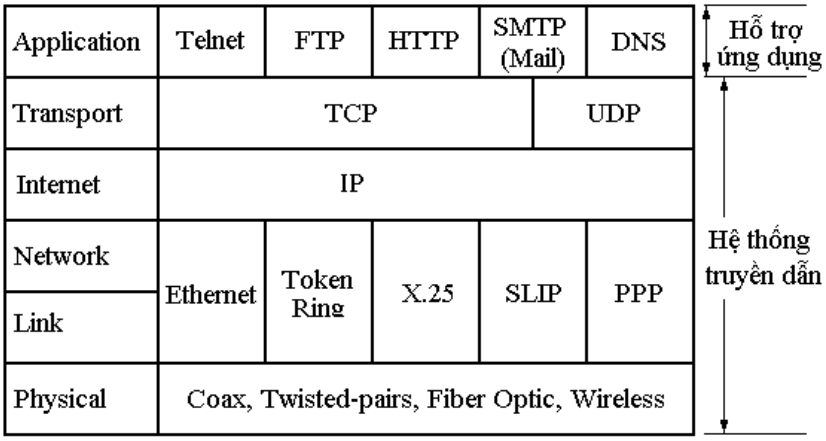
- Các giao thức tạo thành hệ thống truyền dẫn
- Các giao thức hỗ trợ ứng dụng

Tầng liên mạng (Internet)

- Sử dụng giao thức connectionless – IP, là hạt nhân hoạt động của hệ thống truyền dẫn Internet.
- Các thuật toán định tuyến RIP, OSFD, BGP
- Cho phép kết nối một cách mềm dẻo và linh hoạt các loại mạng "vật lý" khác nhau; như: Ethernet, Token Ring, X25 v.v. dựa trên địa chỉ IP.

Tầng giao vận (Transport)

- TCP – Là giao thức hướng nối (connection-oriented)
- UDP – Là giao thức không hướng nối (connectionless)



Hình 5.5. Bộ giao thức TCP/IP

5.3. GIAO THỨC TCP

5.3.1. Giới thiệu

Đặc trưng công nghệ: TCP là một giao thức hướng nối (connection-eoriented), tin cậy (reliable):

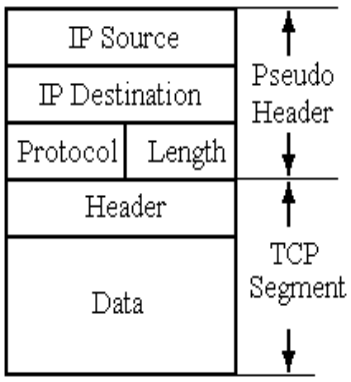
- Vận chuyển end-to-end, tin cậy, đúng thứ tự, thông qua các "phương tiện":
  - Dùng cơ chế báo nhận (ACK)
  - Dùng số thứ tự các gói tin (Sequence number)
  - Dùng phương pháp kiểm soát lỗi mã dư vòng (CRC)
- Điều khiển lưu lượng (flow control + congestion control) bằng cửa sổ trượt có kích thước thay đổi.

Do vậy TCP là một giao thức tương đối phức tạp.

5.3.2. Cấu trúc gói số liệu TCP

Gồm 2 phần:

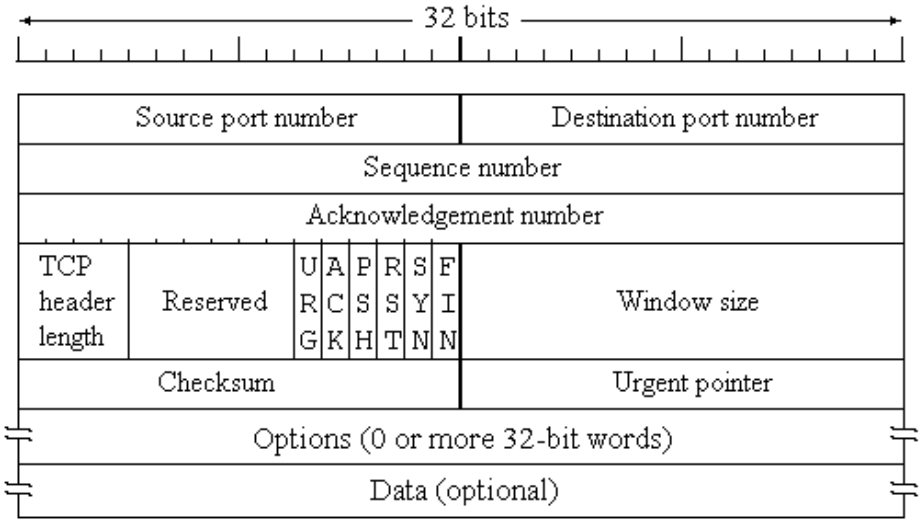
- Tiêu đề giả (Pseudo Header TCP), cần thiết để xây dựng IP packet. Bao gồm:
  - IP Source - Địa chỉ nguồn (32 bit)
  - IP Destination - Địa chỉ đích (32 bit)
  - Protocol = 0x06 (thuộc giao thức TCP)
  - Length: độ dài của TCP segment.
- Gói số liệu TCP thực (TCP Segment)



Cấu trúc TCP segment:

- Source/Destination port number: 2 điểm cuối của kết nối TCP. Port number + IP add → socket (48 bit)

- Seq. number = số thứ tự byte đầu tiên của data so với byte đầu của dòng dữ liệu của thực thể gửi. Giá trị ban đầu = ISN+1 (Initial Sequence Number)
- ACK: Byte tiếp theo có thể nhận (stt byte cuối cùng đã nhận đúng + 1)
- TCP header length: đơn vị 32 bit; đó cũng chính là data offset.
- Resered = 0: để dùng trong tương lai.
- Flags (6):
  - URG =1: có sử dụng trường Urgent pointer.
  - ACK =1: trường Ack đúng.
  - PSH =1: thực thể nhận được y/c chuyển ngay segment này
  - RST =1: Reset kết nối; từ chối kết nối v.v.
  - SYN =1: đồng bộ trường Seq. , dùng để thiết lập kết nối TCP
  - FIN =1: thông báo thực thể gửi đã kết thúc việc gửi số liệu.
- Window size: Độ lớn cửa sổ nhận, cho bên sender biết có thể gửi tiếp bao nhiêu byte, tính từ byte được biên nhận (ack).
- Checksum: checksum của cả TCP segment + Pseudo header. Trước khi tính, trường này = 0. (Tổng các word 16 bit kiểu bù 1, kết quả thu được lại tính bù 1 - XOR).
- Urgent pointer: byte trong trường data của TCP segment cần được xử lý đầu tiên.
- Options: Các tuỳ chọn. Hiện nay tuỳ chọn duy nhất được dùng là MSS (Maximum Segment Size). Giá trị default = 536 byte payload + 20 byte header = 556 byte.
- Pad (ở hình trên slide trước không vẽ): chèn thêm để chiều dài trường Options là bội của 32 bit.
- Data: số liệu của ứng dụng TCP



Hình 5.6. Cấu trúc gói tin TCP segment

5.3.3. Thiết lập và kết thúc kết nối TCP

Thiết lập kết nối

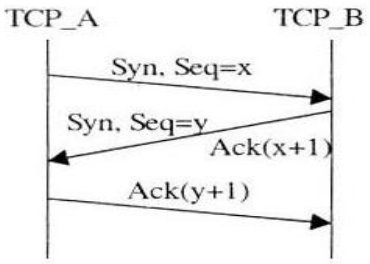
Bắt tay 3 bước (Three-way Handshake):

- A yêu cầu kết nối, gửi CONNECTION REQUEST TPDU:
  - SYN=1, Seq = x (ISN)
  - Source port number, Destination port number
- B nhận được, gửi lại ACK TPDU

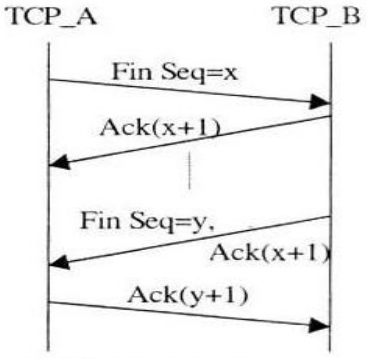
ACK (x+1), SYN=1, Seq = y (ISN)

- A biên nhận ACK TPDU của B

ACK (y+1), Seq = x



a) Thiết lập kết nối



b) Kết thúc kết nối

Phương thức "Bắt tay 3 bước"

Lưu ý về ISN:

- Mỗi thực thể kết nối TCP sử dụng 1 ISN (32 bit) riêng
- ISN tăng theo thời gian, nhằm tránh các kết nối dùng nhầm các segment bị "ôi"

Kết thúc kết nối

- Đơn giản hơn việc thiết lập kết nối
- Có hai kiểu kết thúc kết nối:
  - Asymmetric release: giống trong hệ thống điện thoại.
  - Symmetric release: xử lý kết nối như là 2 kết nối 1 hướng riêng biệt, mỗi kết nối này có thể được giải phóng riêng biệt.

5.3.3. Điều khiển lưu lượng trong TCP

Giao thức TCP được xây dựng dựa trên các khái niệm được Cerf và Kahn đưa ra đầu tiên. Đó là giao thức hướng kết nối, kiểu đầu cuối - đầu cuối, tin cậy, được thiết kế phù hợp với kiến trúc phân lớp các giao thức. Giao thức TCP cung cấp sựtruyền thông tin cậy giữa hai tiến trình chạy trên hai máy tính ở các mạng khác nhau nhưng kết nối với nhau. Giao thức TCP có khả năng gửi và nhận liên tiếp các đơn vị dữ liệu chiều dài có thể thay đổi, được gọi là phân đoạn (segment), mỗi phân đoạn được đóng gói trong một "phong bì" và tạo nên một gói số liệu IP. Khi thiết kế TCP người ta giả thiết rằng TCP sẽ nhận được các dịch vụ vận chuyển gói số liệu không tin cậy (không có sự biên nhận cho các gói số liệu) do các giao thức ở các tầng bên dưới cung cấp. Về nguyên tắc, TCP phải có khả năng hoạt động bên trên một miền rất rộng các hệ thống truyền thông, từ các mạng có các đường truyền cố định, tới các mạng chuyển mạch gói và các mạng chuyển mạch cứng.

Dưới đây, chúng tôi trình bày tóm tắt các thuật toán đã được người ta đề xuất để khắc phục các nguyên nhân dẫn đến sự vi phạm nguyên lý "Bảo toàn các gói số liệu".

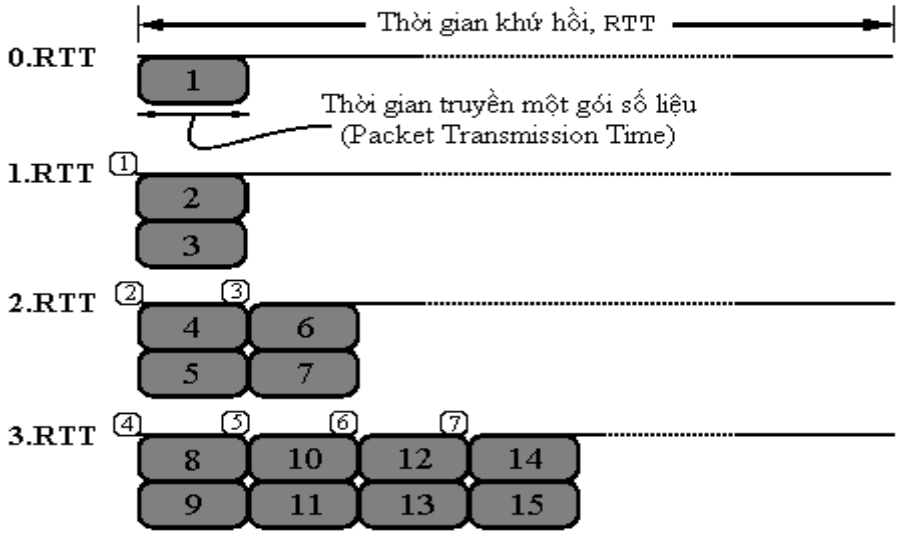
5.3.3.1. Khởi động chậm

Thuật toán khởi động chậm (SS, Slow Start) (hình 4.6) khắc phục nguyên nhân thứ nhất dẫn tới việc vi phạm nguyên lý "Bảo toàn các gói số liệu": ***Tăng dần lượng dữ liệu đang được vận chuyển trong mạng để đạt tới sự cân bằng***. Thuật toán được trình bày cụ thể hơn như sau:

- Bổ sung thêm tham số cửa sổ tắc nghẽn cwnd (congestion window) vào tập trạng thái của mỗi kết nối.
- Khi bắt đầu phát hoặc bắt đầu lại việc phát sau khi có gói số liệu bị mất, đặt cwnd bằng một gói số liệu.
- Mỗi khi nhận được một biên nhận mới, tăng cwnd lên một gói số liệu.
- Khi gửi, gửi số lượng gói số liệu là min của kích thước cửa sổ mà hai bên đã thoả thuận và cwnd.

Thực ra, theo cơ chế khởi động chậm, cửa sổ tăng lên theo hàm mũ, nó đạt tới kích thước W sau thời gian bằng $RTT.\log 2W$, trong đó RTT là thời gian khứ hồi và W tính bằng đơn vị gói số liệu. Điều này có nghĩa là cửa sổ mở đủ nhanh để ảnh hưởng không đáng kể đến hiệu năng, ngay cả trên các đường truyền có tích số dải thông x độ trễ lớn. Theo thuật toán này, *người gửi sẽ truyền dữ liệu với tốc độ khi cao nhất là gấp đôi giá trị cực đại có thể của đường truyền*. Chính vì vậy, giai đoạn khởi động chậm cần phải được kết thúc khi cửa sổ W đạt tới một ngưỡng nhất định.

Hình 5.7 minh hoạ sự tăng của cửa sổ trong cơ chế khởi động chậm. Trục thời gian được cắt thành các đoạn có chiều dài bằng khoảng thời gian khứ hồi RTT, các đoạn này được chồng lên nhau theo chiều đứng, hướng trên-dưới ứng với chiều tăng của thời gian. Các gói số liệu được biểu diễn bằng các hình chữ nhật màu xám, bên trong là số thứ tự của gói số liệu. Các hình vuông nhỏ, không tô màu, có đánh số, biểu diễn cho các gói số liệu biên nhận tương ứng. Trên hình vẽ có thể thấy rõ, mỗi khi có một biên nhận trở về, hai gói số liệu sẽ được phát ra: một gói tương ứng với biên nhận (vì mỗi biên nhận cho biết rằng đã có một gói số liệu rời khỏi mạng, do đó cần gửi đi một gói thế chỗ cho nó), còn gói thứ hai là do biên nhận đã làm tăng cửa sổ lên một đơn vị gói số liệu. Hai gói số liệu này được vẽ chồng lên nhau, thể hiện rằng chúng cần được phát đi đồng thời, tuy nhiên, trong thực tế chỉ có thể phát chúng đi lần lượt, do đó trong khi một gói số liệu đang được phát đi, gói số liệu còn lại trong cửa sổ sẽ phải nằm chờ trong hàng đợi. Khi cửa sổ mở rộng đến kích thước W, thì trong hàng đợi có thể có đến 2xW gói số liệu đang xếp hàng chờ được gửi đi.



Hình 5.7. Sự tăng của cửa sổ trong cơ chế khởi động chậm

5.3.3.2. Tính thời gian khứ hồi một cách thông minh

Tính thời gian khứ hồi một cách thông minh là cách khắc phục nguyên nhân thứ hai dẫn tới việc vi phạm nguyên lý "Bảo toàn các gói số liệu", đó là việc đưa vào mạng một gói tin mới trước khi có một gói tin cũ ra khỏi mạng. Như đã được trình bày tại tiểu mục 1.3.2.2, có hai sai lầm dẫn đến nguyên nhân thứ hai này, cách giải quyết chúng được trình bày dưới đây.

Cách giải quyết sai lầm thứ nhất: tính ước lượng thời gian khứ hồi bằng một bộ lọc dải thông thấp để tránh cho đại lượng này khỏi thăng giáng quá mạnh nhằm duy trì sự cân bằng. Đặc tả cho giao thức TCP, RFC-793 gợi ý tính ước lượng thời gian khứ hồi như sau:

$$RTT \leftarrow \alpha.RTT+(1-\alpha).M \qquad (1)$$

Trong đó RTT là ước lượng thời gian khứ hồi trung bình, M là số đo thời gian khứ hồi nhận được từ gói số liệu đã được biên nhận gần nhất và α là hệ số làm trơn của bộ lọc, giá trị mà người ta gợi ý nên sử dụng là α=0.9.

Sau khi ước lượng về RTT đã được cập nhật, thì thời gian hết giờ để phát lại gói số liệu tiếp theo, RTO (retransmit timeout) được tính như sau:

$$RTO = \beta.RTT \qquad (2)$$

Cần phải chọn β sao cho việc phát lại do hết giờ không bị sai lầm do thăng giáng của thời gian khứ hồi; nghĩa là làm cho xác suất thời gian khứ hồi của một gói tin lớn hơn RTO là rất nhỏ. Chính vì vậy, β cần được chọn không quá nhỏ, có thể sẽ dẫn đến việc phát lại vội vàng, khi gói tin vẫn đang ở trong mạng; β cũng không được chọn quá lớn, có thể sẽ dẫn đến việc phát lại quá chậm trễ, gói tin bị mất từ lâu, mà bên gửi vẫn chờ cho hết giờ rồi mới phát lại.

Trong các phiên bản TCP được cài đặt đầu tiên, người ta thường chọn β là một số cố định bằng 2. Tuy nhiên, các nghiên cứu thực nghiệm sau này cho thấy rằng, RTT thăng giáng trong một miền tương đối rộng, vì vậy không nên chọn β theo cách đơn giản như trên. Công trình đầu tiên đề xuất việc cải tiến thuật toán tính RTO của Jacobson được công bố năm 1988. Ông đã đề xuất cách làm cho β xấp xỉ tỉ lệ với độ lệch chuẩn của hàm mật độ xác suất thời gian đến của biên nhận. Cụ thể là, sử dụng độ lệch trung bình như một ước lượng rẻ (cheap estimator) của độ lệch chuẩn. Thuật toán này đòi hỏi phải tính một biến nữa là độ lệch được làm trơn D, như sau:

$$D = \alpha.D + (1-\alpha).|RTT-M| \qquad (3)$$

Trong đó, các tham số RTT và M hoàn toàn tương tự như trong biểu thức (1), còn α ở đây không nhất thiết phải có cùng giá trị như tham số α trong biểu thức đó. Jacobson đã chỉ ra rằng, mặc dù D không hoàn toàn giống độ lệch chuẩn, nhưng nó cũng là một xấp xỉ đủ tốt. Cách tính D như trên nhằm đạt được tốc độ cao nhất, chỉ sử dụng các phép tính cộng, trừ và dịch trên các số nguyên. Ngày nay, các phiên bản TCP đều sử dụng thuật toán này và tính thời gian hết giờ để phát lại như sau:

$$RTO = RTT + 4.D \qquad (4)$$

Sử dụng hệ số 4 có hai ưu điểm, thứ nhất là việc nhân với 4 sẽ được thực hiện bởi phép dịch, có tốc độ thực hiện cao; thứ hai là, xác suất một gói tin được biên nhận chậm hơn RTO là rất nhỏ, có thể bỏ qua.

Cách giải quyết sai lầm thứ hai: rút lui theo hàm mũ. Đây là cách giải quyết duy nhất đúng đắn, bởi vì theo cơ chế khởi động chậm, cửa sổ gửi tăng lên theo hàm mũ, cho nên cũng cần phải rút lui theo cách này cho đủ nhanh khi đã có dấu hiệu của tắc nghẽn. TCP sẽ đặt đồng hồ phát lại bằng khoảng thời gian rút lui và khoảng đó sẽ được tăng gấp đôi cứ mỗi lần bị hết giờ liên tiếp.

Thuật toán Slow Start (SS):

- Thực thể phát sử dụng thêm biến:
  - + cwnd (congestion window) - kích thước cửa sổ phát
  - + ssthresh (ss threshhold) - giới hạn trên của cwnd, nếu vượt qua → tắc nghẽn.
- Bắt đầu phát, cwnd := 1, đó là tốc độ "an toàn nhất"
- Nhận được mỗi ack, tăng cwnd lên 1 để thăm dò (Additive Increase)
  - + Không tăng cwnd quá Window Size mà bên nhận thông báo.
  - + Thực chất, cwnd tăng lên theo hàm mũ (theo thời gian).
- Khi cwnd ≥ ssthresh, chuyển sang CA

Hình 5.8. Thuật toán Slow Start (SS)

5.3.3.3. Tránh tắc nghẽn

Thuật toán tránh tắc nghẽn (CA, Congestion Avoidance) (hình 4.7) nhằm khắc phục nguyên nhân thứ ba dẫn tới việc vi phạm nguyên lý "Bảo toàn các gói số liệu". Một chiến lược ránh tắc nghẽn đã được đề xuất bao gồm hai thành phần: ***thứ nhất là các chính sách của mạng***: mạng phải có khả năng gửi tín hiệu đến cho các thực thể cuối của các kết nối (endpoint), báo cho chúng biết là tắc nghẽn đang xảy ra hoặc sắp xảy ra; ***thứ hai là các chính sách của endpoint***: các endpoint phải có chính sách giảm lưu lượng đưa vào mạng nếu nhận được các tín hiệu báo và tăng thêm lưu lượng đưa vào mạng nếu không nhận được tín hiệu báo này.

Chính sách của endpoint đối với tắc nghẽn: thích ứng với đường truyền

Đó chính là chính sách tăng theo cấp số cộng, giảm theo cấp số nhân, chính sách đó được giải thích như sau:

- Mỗi khi xảy ra sự kiện hết giờ, đặt giá trị cửa sổ tắc nghẽn cwnd bằng một phần hai giá trị cửa sổ hiện thời. Đó là sự giảm theo cấp số nhân.
- Mỗi khi nhận được một biên nhận cho gói số liệu mới, tăng cwnd thêm một lượng bằng 1/cwnd, đây là sự tăng theo cấp số cộng. (Trong giao thức TCP, kích thước cửa sổ và kích thước gói số liệu được tính bằng byte, vì thế sự tăng nói trên được

chuyển thành *maxseg*maxseg/cwnd*, trong đó *maxseg* là kích thước gói số liệu cực đại và *cwnd* là cửa sổ tắc nghẽn, được tính bằng bytes).

– Khi gửi, sẽ gửi đi số gói số liệu bằng số bé hơn trong hai số: kích thước cửa sổ mà bên nhận đã đề nghị và cwnd.

Trong thực tế, các thuật toán Khởi động chậm (SS) và Tránh tắc nghẽn (CA) đã được triển khai thực hiện cùng với nhau như là một thuật toán, thuật toán này được cài đặt trong Tahoe TCP - một phiên bản của TCP.

Chính sách của mạng đối với tắc nghẽn

Đó là các chính sách làm cho mạng, cụ thể là các gateways gửi tín hiệu báo tắc nghẽn tới các máy tính trên mạng càng sớm càng tốt, nhưng đừng quá sớm, tránh cho mạng khỏi bị thiếu lưu lượng vận chuyển. Gateway chỉ phải làm công việc loại bỏ các gói số liệu để báo cho các thực thể đã gửi các gói số liệu rằng: chúng đã sử dụng quá phần tài nguyên mạng dành cho chúng. Chính vì thế, các thuật toán tại gateway sẽ làm giảm tắc nghẽn ngay cả khi không phải sửa đổi giao thức giao vận ở các máy tính trên mạng, để thực hiện việc tránh tắc nghẽn. Đồng thời các máy tính trên mạng có triển khai thực hiện tránh tắc nghẽn sẽ nhận được phần dải thông hợp lý dành cho nó và chỉ bị mất một số lượng tối thiểu các gói số liệu.

Bởi vì tắc nghẽn tăng lên theo hàm mũ, cho nên việc phát hiện sớm là quan trọng. Nếu tắc nghẽn được phát hiện sớm, thì chỉ cần một vài điều chỉnh nhỏ đối với cửa sổ của người gửi cũng có thể giải quyết được vấn đề; ngược lại, sẽ phải điều chỉnh rất nhiều để mạng có thể chuyển hết đống gói số liệu tắc nghẽn trong mạng ra ngoài. Tuy nhiên, do bản chất luôn thăng giáng mạnh của lưu lượng, phát hiện tắcnghẽn sớm một cách tin cậy là một việc khó.

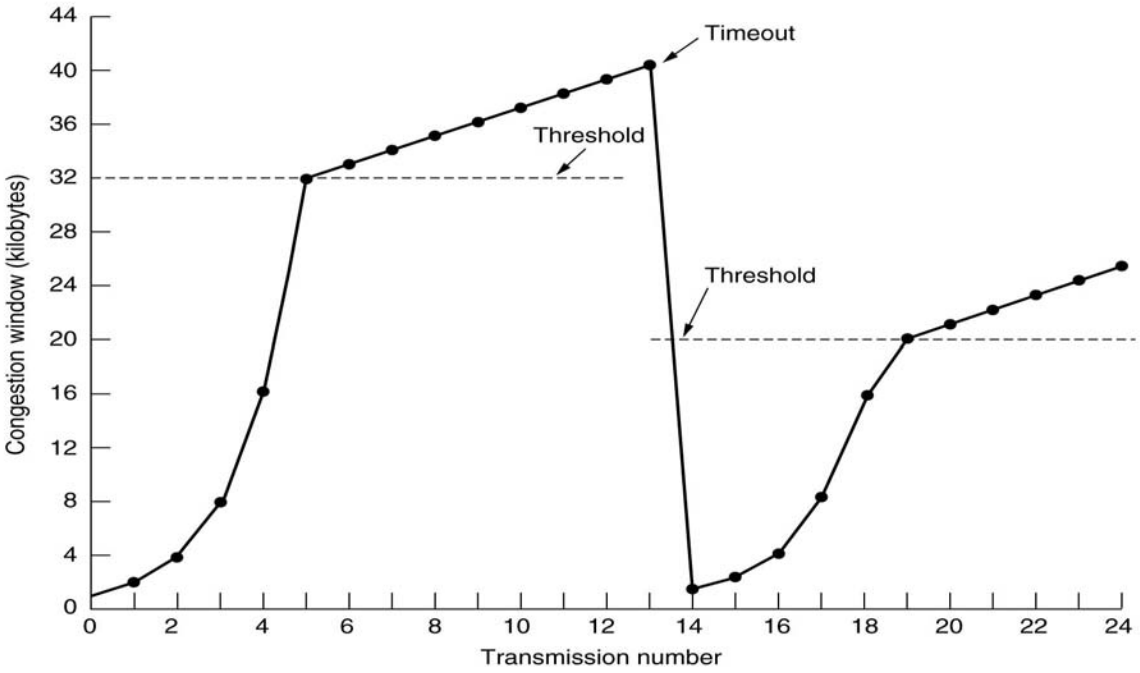
Thuật toán Congestion Avoidance (CA):

- **Dấu hiệu tắc nghẽn:**
  1. RTT tăng quá Timeout, là một giá trị mà thực thể gửi sử dụng để phán đoán là gói tin đã bị mất.
  2. Nhận được nhiều (3) Dup Ack (biên nhận lặp), điều đó cho biết đã có nhiều gói tin không đúng thứ tự đến đích, nghĩa là đã có gói tin bị mất.
- cwnd := cwnd + 1/cwnd với mỗi ack.
- Khi phát hiện dấu hiệu tắc nghẽn:
  + ssthresh := cwnd/2, cwnd := 1
  + RTO = RTO * 2 (Exponential backoff)
  + → SS

Hình 5.9. Thuật toán Congestion Avoidance

Nhận xét:

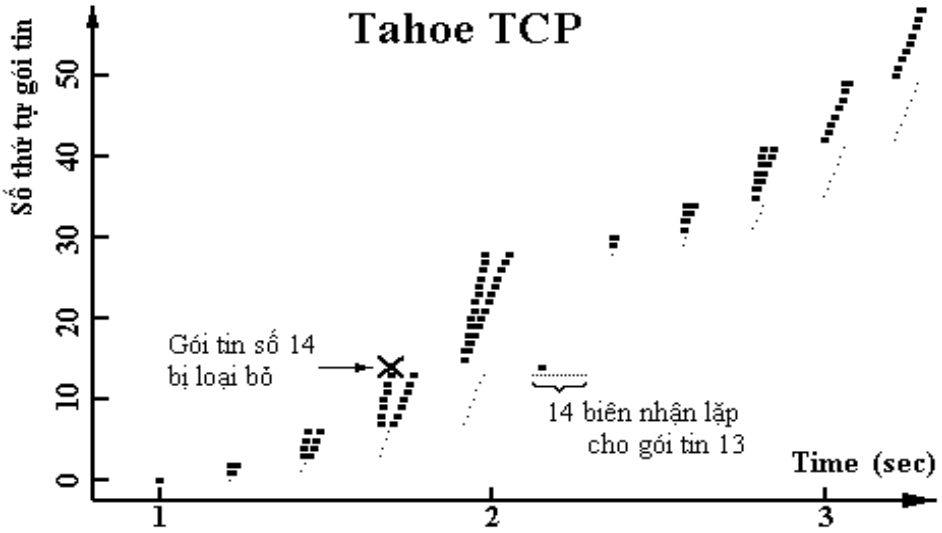
1. Trong giai đoạn CA, cwnd tăng tuyến tính:
   + Đảm bảo tận dụng băng thông có thể sử dụng được
   + Vẫn thăm dò tiếp khả năng sử dụng băng thông nhiều hơn
2. cwnd bị giảm theo cấp số nhân (Multiplicative Decreased)



Hình 5.10. Minh hoạ thuật toán SS và CA

Thuật toán Fast Retransmit (FRTX):

- Sau khi nhận được Dupack (>=3), TCP thực hiện phát lại nhanh, không chờ bị Timeout, sau đó chuyển ngay về SS.
- Đây là một cách "dự đoán thông minh" rằng, gói tin đã bị mất.

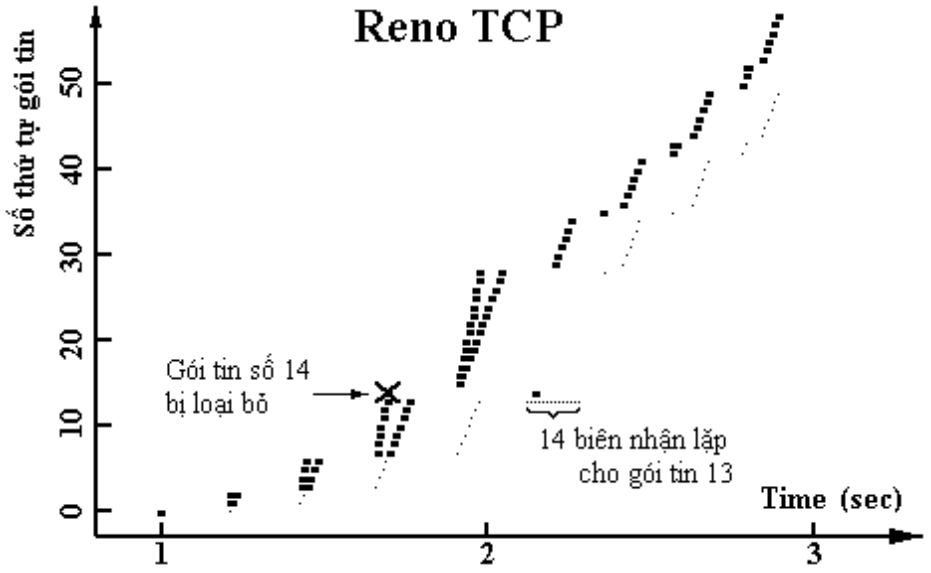


Hình 5.11. Giao thức Tahoe TCP

Thuật toán Fast Recovery (FRCV):

Cải tiến FRTX: thực hiện FRTX xong về CA chứ không về SS:

- ssthresh := cwnd/2, nhưng không nhỏ hơn 2 (gói tin)
- cwnd := cwnd + 3. Bên gửi "đoán": 3 dupack ứng với 3 gói tin đã được nhận đúng.
- Với mỗi dupack nhận được thêm, tăng cwnd := cwnd + 1



Hình 5.12. Giao thức Reno TCP

5.3.4. Giao thức UDP (User Datagram protocol)

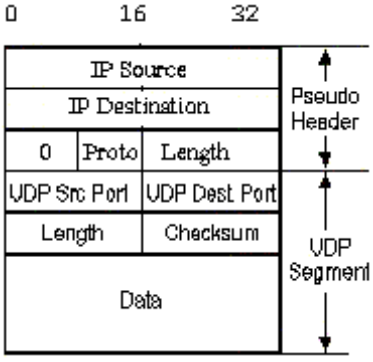
- Không hướng nối (connectionless)
- Không bảo đảm (unreliable): không có cơ chế kiểm tra STT phát, thu và kiểm soát lỗi.

→ Ưu điểm: đơn giản.

- Dành cho các ứng dụng:
  + Trong đó việc phân phát tin nhanh chóng là quan trọng hơn việc phân phối tin chính xác: Email, v.v.
  + Muốn tự cung cấp các chức năng flow control và error control
- Thống kê thực tế cho thấy: 99% các gói tin UDP được vận chuyển đến đích không bị lỗi.

Cấu trúc gói số liệu UDP

- Tương tự cấu trúc TCP segment
- "Header giả" giúp thực thể IP xây dựng IP packet.
- Lenght trong Pseudo header: độ dài toàn bộ gói số liệu UDP, kể cả "Pseudo header"
- Lenght trong UDP segment: độ dài UDP segment, min = 8.
- Checksum: tính cho toàn bộ gói số liệu UDP.



5.4. GIAO THỨC LIÊN MẠNG IP

5.4.1. Giới thiệu

Đặc trưng công nghệ:

Connectionless = Datagram

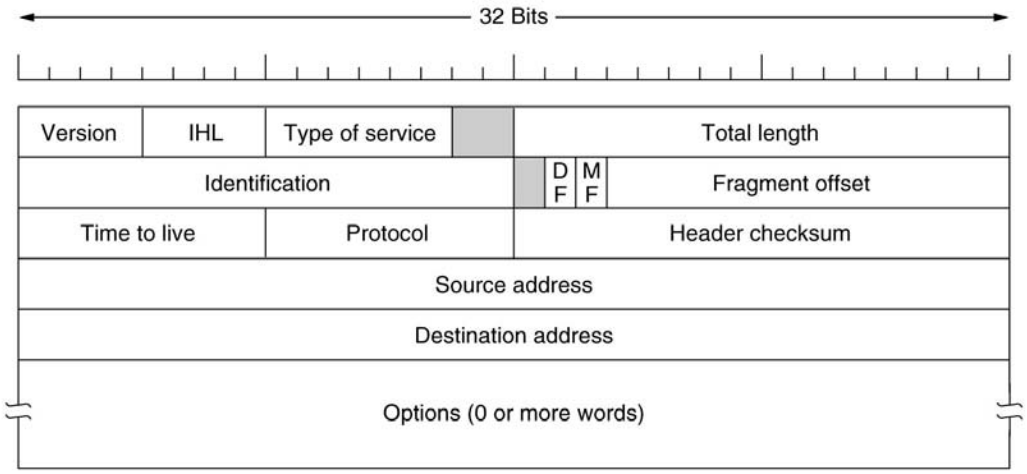
- Không phải thiết lập; giải phóng kết nối
- Packets có thể đi theo các con đường khác nhau
- Không có cơ chế phát hiện/khắc phục lỗi truyền

→ Giao thức đơn giản, độ tin cậy không cao.

Các chức năng chính:

- Định nghĩa khuôn dạng gói dữ liệu (IP packet)
- Định nghĩa phương thức đánh địa chỉ IP
- Chon đường (Routing)
- Cắt/hợp dữ liệu (Fragmentation/ Reassembly)

5.4.2. Cấu trúc gói số liệu IP



Hình 5.13. Cấu trúc gói tin IPv4

- Version (4 bit): IPv4 hoặc IPv6
- IHL (IP packet Header Length) (4 bit): đơn vị word 32 bit.
  - Min = 5 (không có thêm trường tuỳ chọn)
  - Max = 15 (trường tuỳ chọn là 40 byte)
  - Đối với một số tuỳ chọn, thí dụ để ghi con đường mà packet đã đi qua, 40 byte là quá nhỏ, không thể dùng được.
- Trường Type of service (8 bits): Dịch vụ và mức ưu tiên.
  - Ý nghĩa của nó được người ta thay đổi chút ít trong các năm qua.
  - Có thể có nhiều cách kết hợp khác nhau giữa độ tin cậy và tốc độ. Đối với tiếng nói được số hoá, việc phân phát nhanh quan trọng hơn phân phát chính xác. Đối với FTP, việc truyền không có lỗi quan trọng hơn việc truyền nhanh.
  - Bản thân chính trường này lại bao gồm một số trường, tính từ trái qua phải như sau:

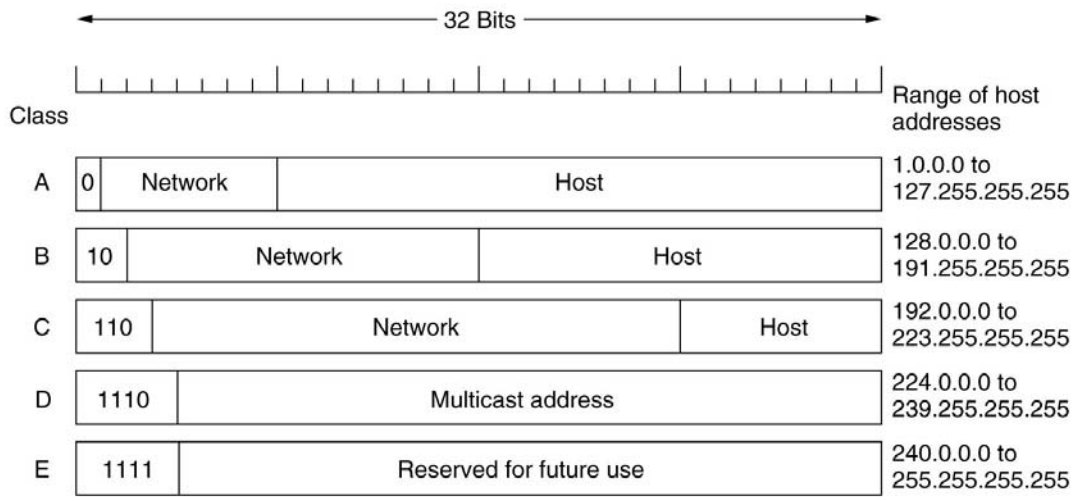
Precedence (3 bit đầu tiên): quyền ưu tiên; 0 = normal, ... , 7 = network control packet.

+ Cờ D, T và R (3 bit tiếp theo): cho phép host chỉ ra là nó quan tâm (cần) đến gì nhất trong tập hợp {Delay, Throughput và Reliability}.
  Trong thực tế, các router hiện nay lờ toàn bộ trường Type of service
+ 2 bit còn lại hiện nay chưa dùng đến.

- Trường Total Length (16 bits): Tổng chiều dài packet, kể cả header lẫn data, đơn vị = byte.
  – Max = 65535 byte
  – Hiện nay giới hạn trên là có thể chấp nhận được
  – Với các mạng Gigabit trong tương lai sẽ cần đến các datagram lớn hơn.
- Trường Identification (16 bit): từ định danh của datagram (IP packet)
  – Dùng cho host đích xác định được mảnh (fragment) thuộc về datagram nào.
  – Tất cả các mảnh của một datagram có cùng một giá trị của trường Identification.
- Trường Flags (2 bits): dùng cho quá trình Fragmentation/ Reassembly
  – Sau trường Identification là một bit không dùng đến. Flags gồm 2 trường 1 bit là DF và MF.
  – DF (Don't Fragment): lệnh cho các router đừng có phân mảnh datagram.
    • Datagram phải tránh mạng có kích thước packet nhỏ.
    • Tất cả các máy được yêu cầu chấp nhận việc phân mảnh đến 576 byte hoặc nhỏ hơn.
  – MF (More Fragments): Tất cả các mảnh của datagram, trừ mảnh cuối cùng phải có bit MF=1 → để biết được khi nào tất cả các mảnh của một datagram đã đến đích.
- Trường Fragment offset (13 bits): cho biết khoảng cách tương đối của gói tin IP trong gói tin bị phân mảnh.
  – Tất cả các mảnh của một datagram, trừ mảnh cuối cùng phải có chiều dài là bội số của 8 bytes - đơn vị cơ sở của mảnh.
  – 13 bit nên số mảnh lớn nhất của một datagram là 8192
- Trường Time to live – TTL (8 bits): con đếm thời gian sống của một packet
  – Khi TTL= 0, packet bị loại bỏ và một packet cảnh báo được gửi cho bên nguồn → Ngăn chặn các datagram đi lang thang mãi (nếu bảng chọn đường có lúc bị hỏng)
  – Giả thiết đơn vị là giây → max = 255s; thường được đặt = 30s
  – Phải được giảm đi một tại mỗi chặng (hop) và được giảm nhiều lần khi đứng xếp hàng một thời gian dài trong mỗi router.
  – Thực tế, nó chỉ đếm các chặng.
- Trường Protocol (8 bits): Chỉ loại số liệu giao thức mức trên nằm trong trường Data.
  – Cho biết cần trao datagram cho quá trình nào của tầng transport.
    + Một khả năng là TCP
    + Nhưng cũng có thể là UDP và các quá trình khác.
  – Việc đánh số các giao thức là trên phạm vi toàn cầu, trên toàn bộ Internet, được định nghĩa trong chuẩn RFC 1700.

- Trường Header checksum (16 bits):
  - Tính riêng cho header, giúp phát hiện các lỗi phát sinh trong bộ nhớ của router.
  - Được tính lại tại mỗi chặng (hop), bởi vì sau mỗi chặng có ít nhất là một trường bị thay đổi (trường TTL).
  - Cách tính: cộng tất cả các 16-bit halfwords sử dụng số dạng bù 1; sau đó lấy bù 1 của kết quả (phép toán XOR → tốc độ cao).
- Trường Source address, Destination address (32 bit):
  - Địa chỉ IP của bên gửi và nhận
  - Mỗi địa chỉ bao gồm: địa chỉ mạng và địa chỉ host trong mạng
- Trường Options: Tạo ra lối thoát cho các version sau:
  - Bổ sung thêm các thông tin không có trong version đầu tiên
  - Thí nghiệm thử các ý tưởng mới và để tránh việc phải dành (allocate) các bit của header cho các thông tin hiếm khi cần đến.
  - Chiều dài có thể thay đổi: 0..(15 – 5) x 32 bits
  - Mỗi Option bắt đầu bằng một mã 1 byte chỉ ra tuỳ chọn
  - Hiện thời có 5 tuỳ chọn (option) đã được định nghĩa
    + Security (an ning): Chỉ ra mức độ bí mật của datagram
    + Strict source routing: Chỉ ra con đường đầy đủ để đi theo
    + Loose source routing: Chỉ ra danh sách các router không được bỏ qua
    + Record route: Buộc mỗi router gắn địa chỉ IP của nó vào
    + Timestamp: Buộc mỗi router gắn địa chỉ IP và timestamp của nó vào
  - Tuy nhiên không phải mọi router đều hỗ trợ tất cả các tuỳ chọn này.
  - Padding: Được chèn thêm sao cho chiều dài Header = bội của 32 bits
- Trường Data (32 bits): Số liệu của giao thức tầng trên.

5.4.3. Các lớp địa chỉ IP



Hình 5.14. Cấu trúc địa chỉ IP

- Mỗi địa chỉ IP (IPv4) gồm 32 bits, được chia thành 3 vùng, đó là Class + Netid + Hostid, mỗi máy nối mạng phải có địa chỉ IP duy nhất trên Internet
- Những máy nối với nhiều mạng có các địa chỉ IP khác nhau trên từng mạng.

– Được chia thành 4 lớp: A, B, C, D và E (dự trữ), cấu trúc các lớp địa chỉ được chỉ rõ trong hình trên.
– Cách viết địa chỉ Internet: số thập phân có chấm (***Dotted Decimal Notation)***, tức là có dạng x.y.z.t, trong đó x, y, z, t có giá trị từ 0-255 (mỗi số tương ứng với 8 bits).
– Để tránh đụng độ, các địa chỉ mạng được NIC (Network Information Center) gán.

Theo cấu trúc trên:

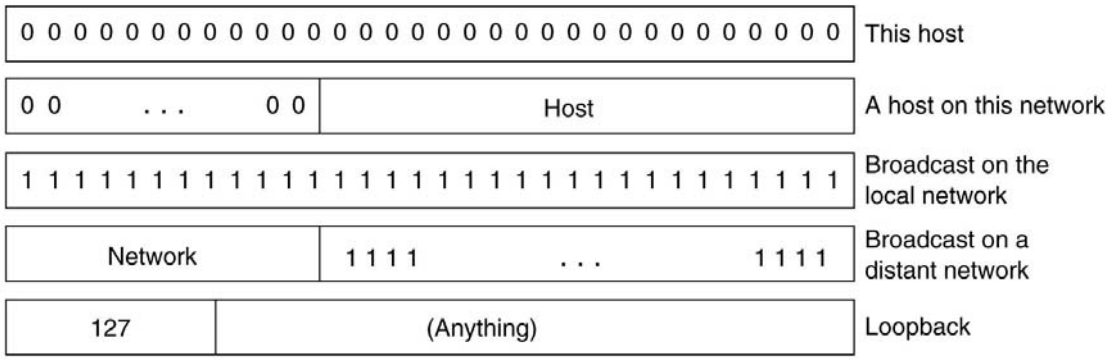
- Lớp A cho phép định danh tới 126 mạng, với số máy tối đa tới hơn 16 triệu máy trên mỗi mạng. Lớp này được dùng cho những mạng có số máy cực lớn.
- Lớp B cho phép định danh tới 16384 mạng, mỗi mạng có thể cho phép tối đa 65535.
- Lớp C cho phép định danh tới hơn 2 triệu mạng, với tối đa 254 host mỗi mạng. Lớp này được dùng cho những mạng nhỏ.
- Lớp D dùng để gửi một nhóm các host trên một mạng (địa chỉ broadcast).
- Lớp E là lớp dự phòng cho tương lai.

Để nhận dạng các lớp địa chỉ chỉ cần nhìn vào octet đầu tiên của địa chỉ IP (giá trị của x trong cấu trúc x.y.z.t). Thật vậy, lớp A có x ∈ 1÷126, lớp B có x ∈ 128÷191, lớp C có x ∈ 192÷223, lớp D có x ∈ 224÷239, lớp E có x ∈ 240÷255.

Một số địa chỉ IP đặc biệt

Dưới đây là một số các địa chỉ đặc biệt, không được dùng để cấp phát cho các host:

- 0.0.0.0: Dùng cho các host khi mới khởi động
- Netid = 0: dùng cho các host khi không biết địa chỉ mạng của nó
- Hostid = 1: là địa chỉ quảng bá trên mạng cục bộ được định danh bởi phần Netid, nghĩa là nếu một máy nào đó gửi một gói tin đến địa chỉ có dạng này, thì tất cả các máy trong mạng đều nhận được.
- 127.xx.yy.zz: packet gửi tới địa chỉ này không được đưa lên đường truyền, mà được xử lý cục bộ giống như packet đến. Cho phép gửi packet đến mạng cục bộ mà người gửi không cần biết địa chỉ của nó.



Hình 5.15. Các địa chỉ IP đặc biệt

5.4.4. Các bước thực hiện của giao thức IP

(đang tiếp tục bổ sung)

5.5. PHÂN CHIA MẠNG CON

Những ích lợi của việc phân chia mạng con

- Dễ quản lý hơn (vì số trạm ít hơn).
- Hạn chế miền quảng bá, tăng hiệu quả truyền thông trong mạng. Ví dụ: một mạng LAN có 10 máy, nếu dùng mạng lớp C sẽ có miền quảng bá tới 254 host, nếu dùng subnet mask thì miền quảng bá sẽ giảm xuống, hiệu quả mạng sẽ tăng lên.
- Tăng cường mức độ bảo mật mức thấp cho LAN: mỗi mạng có một danh sách truy cập, theo danh sách này mạng có thể cho phép hay từ chối truy cập vào nó.
- Có thể bán hoặc cho thuê các đại chỉ không được dùng đến: các công ty được sử hữu các mạng lớn lớp A, B có thể không dùng hết số địa chỉ của họ, họ có thể bán hoặc cho thuê chúng.

Áp dụng: phân chia mạng 132.198.0.0 thành 14 mạng

Subnet mask được xây dựng từ số bít 1 phần Netid cộng với phần mượn từ Hostid (số bit mược tối đa là số bit phần Hostid - 2). Tuỳ theo số lượng mạng con ta sẽ mượn số bit tương ứng, mượn n bit từ Hostid thì ta sẽ phân chia được 2^n mạng con.

Ta thấy 132.198.0.0 là một mạng thuộc lớp B.

Để phân chia thành 14 mạng con, ta cần mượn 4 bit phần Hostid, vậy subnet mask là:

1111 1111. 1111 1111. 1111 0000. 0000 0000

Hay 255.255.224.0

Sau đây là bảng các địa chỉ dùng riêng không được cấp cho các host trên mỗi mạng con khi thực hiện việc phân chia như trên:

Mạng con	Địa chỉ mạng con	Địa chỉ broadcast
1	132.198.16.0	132.198.31.255
2	132.198.32.0	132.198.47.255
3	132.198.48.0	132.198.63.255
4	132.198.64.0	132.198.79.255
5	132.198.80.0	132.198.95.255
6	132.198.96.0	132.198.111.255
7	132.198.112.0	132.198.127.255
8	132.198.128.0	132.198.143.255
9	132.198.144.0	132.198.159.255
10	132.198.160.0	132.198.175.255
11	132.198.176.0	132.198.191.255
12	132.198.192.0	132.198.207.255
13	132.198.208.0	132.198.223.255
14	132.198.224.0	132.198.239.255

5.6. ĐỊA CHỈ IPV6

(đang tiếp tục bổ sung)

5.7. INTRANET VÀ INTERNET

(đang tiếp tục bổ sung)

5.8. MỘT SỐ ỨNG DỤNG TRÊN INTERNET

(đang tiếp tục bổ sung)

5.9. CÂU HỎI VÀ BÀI TẬP

(đang tiếp tục bổ sung)

DANH MỤC TÀI LIỆU THAM KHẢO

[1]. Nguyễn Thúc Hải, "*Mạng máy tính và các hệ thống mở*", NXB Giáo dục, 1999.

[2]. Nguyễn Đình Việt, *Nghiên cứu phương pháp đánh giá và cải thiện hiệu năng giao thức TCP cho mạng máy tính*, Luận án tiến sĩ, Khoa Công nghệ, Đại học Quốc gia Hà nội, 2003.

[3]. Nguyễn Hồng Sơn, "*Giáo trình hệ thống mạng máy tính, CCNA Semester 1*", NXB Lao động- Xã hội, 2005.

[4]. Nguyễn Đình Việt, Slides bài giảng môn học "Truyền số liệu và Mạng máy tính", Đại học Công nghệ, Đại học Quốc gia Hà nội.

[5]. Đào kiến Quốc, "Bài giảng mạng LAN", ĐH Công nghệ, ĐH Quốc gia Hà nội.

[6]. Các website:

http://www.coltech.vnu.edu.vn

http://www.quantrimang.com

http://www.ebook.edu.vn

.....

[7]. Một số tài liệu tiếng Anh nữa...