**LẬP TRÌNH MẠNG**

TCP/IP hoặc Transmission Control Protocol/Internet Protocol (Giao thức điều khiển truyền vận/giao thức mạng) là một bộ các giao thức trao đổi thông tin được sử dụng để kết nối các thiết bị mạng trên Internet. TCP/IP có thể được sử dụng như là một giao thức trao đổi thông tin trong một mạng riêng.

Có hai giao thức mạng chính trong bộ giao thức mạng phục vụ các chức năng cụ thể.

* **TCP** xác định cách các ứng dụng tạo kênh giao tiếp trong mạng. Ngoài ra, nó cũng quản lý cách các tin được phân thành các gói nhỏ trước khi được chuyển qua Internet và được tập hợp lại theo đúng thứ tự tại địa chỉ đến.
* **IP** xác định cách gán địa chỉ và định tuyến từng gói để đảm bảo nó đến đúng nơi. Mỗi gateway trên mạng kiểm tra [địa chỉ IP](https://quantrimang.com/dia-chi-ip-la-gi-143625) này để xác định nơi chuyển tiếp tin nhắn.

TCP/IP được chia thành bốn tầng, mỗi tầng bao gồm các giao thức cụ thể.

* **Tầng ứng dụng** cung cấp các ứng dụng với trao đổi dữ liệu được chuẩn hóa. Các giao thức của nó bao gồm Giao thức truyền tải siêu văn bản (HTTP), Giao thức truyền tập tin (File Transfer Protocol - FTP), Giao thức POP3, Giao thức truyền tải thư tín đơn giản (Simple Mail Transfer Protocol - SMTP) và Giao thức quản lý mạng đơn giản (Simple Network Management Protocol - SNMP).
* **Tầng giao vận** chịu trách nhiệm duy trì liên lạc đầu cuối trên toàn mạng. TCP xử lý thông tin liên lạc giữa các máy chủ và cung cấp điều khiển luồng, ghép kênh và độ tin cậy. Các giao thức giao vận gồm giao thức TCP và giao thức UDP (User Datagram Protocol), đôi khi được sử dụng thay thế cho TCP với mục đích đặc biệt.
* **Tầng mạng**, còn được gọi là tầng Internet, có nhiệm vụ xử lý các gói và kết nối các mạng độc lập để vận chuyển các gói dữ liệu qua các ranh giới mạng. Các giao thức tầng mạng gồm IP và ICMP (Internet Control Message Protocol), được sử dụng để báo cáo lỗi.
* **Tầng vật lý** bao gồm các giao thức chỉ hoạt động trên một liên kết - thành phần mạng kết nối các nút hoặc các máy chủ trong mạng. Các giao thức trong lớp này bao gồm Ethernet cho mạng cục bộ (LAN) và Giao thức phân giải địa chỉ (Address Resolution Protocol - ARP).
* **Ưu điểm của TCP/IP**
* TCP/IP không thuộc và chịu sự kiểm soát của bất kỳ công ty nào, do đó bộ giao thức mạng này có thể dễ dàng sửa đổi. Nó tương thích với tất cả các hệ điều hành, vì vậy có thể giao tiếp với các hệ thống khác. Ngoài ra, nó còn tương thích với tất các các loại phần cứng máy tính và mạng.
* TCP/IP có khả năng mở rộng cao và như một giao thức có thể định tuyến, nó có thể xác định đường dẫn hiệu quả nhất thông qua mạng.

# Giao thức TCP (Tranmission Control Protocol)

* TCP là giao thức truyền tải hướng kết nối (connection-oriented), nghĩa là phải thực hiện thiết lập kết nối với đầu xa trước khi thực hiện truyền dữ liệu. Tiến trình thiết lập kết nối ở TCP được gọi là tiến trình ***bắt tay 3 bước*** (threeway handshake).
* Cung cấp cơ chế báo nhận (Acknowledgement) :Khi A gửi dữ liệu cho B, B nhận được thì gửi gói tin cho A xác nhận là đã nhận. Nếu không nhận được tin xác nhận thì A sẽ gửi cho đến khi B báo nhận thì thôi.
* Cung cấp cơ chế đánh số thứ tự gói tin (sequencing) cho các đơn vị dữ liệu được truyền, sử dụng để ráp các gói tin chính xác ở điểm nhận và loại bỏ gói tin trùng lặp.
* Có các cơ chế điều khiển luồng thích hợp (flow control) để tránh nghẽn xảy ra.
* Hỗ trợ cơ chế full-duplex ( truyền và nhận dữ liệu cùng một lúc)
* Phục hồi dữ liệu bị mất trên đường truyền ( A gửi B mà không thấy xác nhận sẽ gửi lại) .

Giả sử host A muốn truyền dữ liệu cho host B thông qua một kết nối TCP. Trước khi thực hiện truyền , host A cần phải thiết lập kết nối TCP với host B việc này được tiến hành thông qua quá trình bắt tay 3 bước như s:

* **Bước 1**: Host A gửi cho B một gói tin có cờ SYN được bật lên, với số thứ tự được đánh là 100. Segment đầu tiên này không chứa phần dữ liệu nên không có phần data, tuy nhiên số lượng byte dữ liệu vẫn được tính là một byte cho hoạt động gửi cờ SYN.
* **Bước 2**: Host B nhận được gói tin thì B gửi lại gói tin có cờ SYN được bật lên, kèm theo đó là cờ ACK để xác nhận.

Giả sử host B thiết lập segment có số thứ tự là 300. Segment trả lời từ Host B này cũng không có dữ liệu nhưng vẫn được tính là 1 byte cho phần data. Khi phản hồi lại host A, host B cũng cần phải chỉ rõ trong trường ACK sequence số thứ tự của byte kế tiếp mà nó muốn nhận từ host A. Do segment SYN do A gửi qua được tính là 1 byte nên B sẽ mong muốn nhận byte tiếp theo là byte thứ 101 từ A , do đó ACK sequence được đánh số là 101. (SEQ=300, ACK=101)

* **Bước 3**: Sau khi kết nối đã được thiết lập thì A gửi lại gói tin để đáp ứng nhu cầu của B. Gói tin được đánh số SEQ = 101 để đáp ứng nhu cầu của B. ACK =301 dùng để báo là đã nhận được gói tin có SEQ = 300.

# Giao thức UDP (User Datagram Protocol)

* Ngược lại với giao thức TCP thì UDP là giao thức truyền tải hướng không kết nối (connectionless). Nó sẽ không thực hiện thao tác xây dựng kết nối trước khi truyền dữ liệu mà thực hiện truyền ngay lập tức khi có dữ liệu cần truyền (kiểu truyền best effort) => truyền tải rất nhanh cho dữ liệu của lớp ứng dụng.
* Không đảm bảo tính tin cậy khi truyền dữ liệu và không có cơ chế phục hồi dữ liệu ( nó không quan tâm gói tin có đến đích hay không, không biết gói tin có bị mất mát trên đường đi hay không) => dễ bị lỗi.
* Không thực hiện các biện pháp đánh số thứ tự cho các đơn vị dữ liệu được truyền…
* Nhanh và hiệu quả hơn đối với các dữ liệu có kích thước nhỏ và yêu cầu khắt khe về thời gian.
* Bản chất không trạng thái nên UDP hữu dụng đối với việc trả lời các truy vấn nhỏ với số lượng lớn người yêu cầu.
* UDP hoạt động tương tự như TCP nhưng nó không cung cấp kiểm tra lỗi khi truyền gói tin.
* Khi một ứng dụng sử dụng UDP, các gói tin chỉ được gửi đến người nhận. Người gửi không đợi để đảm bảo người nhận nhận được gói tin hay không, mà tiếp tục gửi các gói tiếp theo. Nếu người nhận miss mất một vài gói tin UDP thì gói tin đó coi như bị mất vì người gửi sẽ không gửi lại chúng. => Các thiết bị có thể giao tiếp nhanh hơn.

**CODE**

**Lab3. UDP**

*Client*

Text

Description automatically generated

Text

Description automatically generated

**Lab04 Gửi thông tin Object**

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Lab06 chat client – server

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Gửi Object bằng cả TCP và UDP

ClientClass

Text

Description automatically generated

Text

Description automatically generated

Sinhvien.cs

Text

Description automatically generated

Text

Description automatically generated

ServerClass

Text

Description automatically generated

Text

Description automatically generated

SimpleClientTCP

Text

Description automatically generated

SimpleServerTCP

Text

Description automatically generated

SimpleClientUDP

Text

Description automatically generated

SimpleServerUDP

Text

Description automatically generated