

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1



MẠNG MÁY TÍNH

Sinh viên	Nguyễn Hoàng Hải
Mã sinh viên	K25DTCN134
Lớp	D25TXCN05-K

Yêu cầu:

- Điền đầy đủ thông tin họ tên, mã sinh viên, lớp;
- Font chữ Time News Roman, size 14, giãn dòng giữa các paragraph 1.15, top: 2 cm, bottom: 2cm, right: 3,5 cm, left: 1,5 cm.
- **Độ dài: ít nhất 10 trang A4;**
- Trả lời đúng trong tâm câu hỏi; kết hợp với hình ảnh minh họa, nên trả lời đầy đủ, rõ ràng, chi tiết;
- Nộp bài trên hệ thống LMS;
- Từng bạn nộp bài theo cá nhân; các bạn làm trên tệp docx và nộp file PDF lên hệ thống.
- ***Hạn nộp deadline: 22h00 ngày 12/10/2025. Bạn nào không nộp thì điểm thành phần sẽ là ZERO.***

Phần 1: Câu hỏi lấy điểm kiểm tra cá nhân

1) Phân loại mạng máy tính

Dựa vào phạm vi phân bố của các máy tính trong mạng người ta có thể phân ra các loại mạng, trong đó khái niệm mạng cục bộ (LAN - Local Area Network) và mạng diện rộng (WAN - Wide Area Network) thường hay được nhắc tới.

- Mạng LAN (Local Area Network):

Là mạng máy tính cục bộ, thường có phạm vi nhỏ, như trong một tòa nhà, văn phòng hoặc trường học.

Đặc điểm: Tốc độ cao, chi phí thấp, dễ triển khai.

- Mạng MAN (Metropolitan Area Network):

Là mạng có phạm vi rộng hơn LAN, có thể bao phủ một thành phố hoặc một khu vực đô thị.

Đặc điểm: Chạy trên các đường truyền cáp quang hoặc các công nghệ băng thông rộng khác.

- Mạng WAN (Wide Area Network):

Là mạng có phạm vi rất rộng, có thể bao phủ một quốc gia hoặc toàn cầu. Internet là một ví dụ điển hình của WAN.

Đặc điểm: Tốc độ không cao như LAN, thường yêu cầu các thiết bị chuyển mạch và định tuyến phức tạp.

- Mạng PAN (Personal Area Network):

Mạng có phạm vi rất nhỏ, thường kết nối các thiết bị cá nhân trong khoảng cách vài mét (ví dụ: Bluetooth, Wi-Fi Direct).

Đặc điểm: Sử dụng trong các tình huống kết nối cá nhân giữa các thiết bị như điện thoại, máy tính bảng, tai nghe, v.v.

2) Mô hình OSI và chức năng các tầng

Mô hình OSI ra đời nhằm mục đích cho phép hai hệ thống bất kì trao đổi thông tin với nhau mà không cần thay đổi bất cứ phần cứng hoặc phần mềm nào của mỗi hãng sản xuất.

Mô hình OSI được phân tầng với mục đích thiết kế các hệ thống mạng cho phép việc truyền thông thực hiện được qua tất cả các kiểu hệ thống máy tính khác nhau. Các tầng được thiết kế riêng biệt nhưng liên quan chặt chẽ với nhau, mỗi tầng định nghĩa một phần của quá trình truyền thông tin trên mạng. Nắm vững những quy tắc cơ bản của mô hình OSI là tiền đề vững chắc để thiết kế và phát triển các hệ thống thông tin trên mạng. Mô hình OSI gồm 7 tầng sau:

- + Tầng ứng dụng. (Application layer)
- + Tầng trình diễn (Presentation layer)
- + Tầng phiên (Session layer)
- + Tầng vận tải (Transport layer)
- + Tầng mạng (Network layer)
- + Tầng liên kết dữ liệu (Datalink layer)
- + Tầng vật lý (Physical layer)

+ **Tầng vật lý** thực hiện các chức năng cần thiết để truyền luồng bit dữ liệu đi qua các môi trường truyền dẫn, nó giải quyết các vấn đề liên quan đến đặc điểm kỹ thuật về cơ và điện của giữa giao diện của thiết bị mạng với môi trường truyền dẫn. Để thực hiện vai trò này, lớp vật lý cần phải xác định các thủ tục và các chức năng mà các thiết bị vật lý và thiết bị giao tiếp cần phải tuân thủ. Tầng vật lý thực hiện các chức năng sau:

- Đảm bảo giao tiếp với môi trường truyền dẫn
- Biểu diễn dữ liệu dưới dạng bit
- Tốc độ truyền dẫn
- Đồng bộ
- Quản lý kênh truyền

- Hình trạng vật lý
- Chế độ truyền dẫn

+ **Tầng liên kết dữ liệu** đảm bảo truyền tin tin cậy giữa hai thiết bị vật lý kết nối trực tiếp với nhau, dữ liệu tại tầng này gọi là khung (Frame). Tầng liên kết dữ liệu đảm nhiệm các chức năng sau:

- Tạo khung dữ liệu
- Quản lý địa chỉ vật lý
- Kiểm soát lưu lượng
- Kiểm soát lỗi
- Kiểm soát truy cập

+ **Tầng mạng** chịu trách nhiệm chuyển dữ liệu (dữ liệu của tầng mạng gọi là gói tin – packet) giữa các thiết bị đầu cuối của người sử dụng. Nếu như tầng vận tải đảm bảo liên kết đầu cuối tới mức tiến trình thì tầng mạng chỉ đảm bảo liên kết ở mức đầu cuối của người sử dụng. Theo định nghĩa ban đầu, tầng mạng giải quyết các vấn đề dẫn các gói tin qua một mạng. Một số ví dụ về các giao thức như vậy là X.25, và giao thức Host/IMP của mạng ARPANET. Với sự xuất hiện của khái niệm liên mạng, các chức năng mới đã được bổ sung cho tầng này, đó là chức năng dẫn đường cho dữ liệu từ mạng nguồn đến mạng đích. Nhiệm vụ này thường đòi hỏi việc định tuyến cho gói tin qua một mạng lưới của các mạng máy tính, đó là liên mạng. Tầng mạng đảm nhiệm các chức năng sau:

- Quản lý địa chỉ logic
- Định tuyến

+ **Tầng vận tải** chịu trách nhiệm chuyển toàn bộ bản tin từ nơi gửi đến nơi nhận một cách toàn vẹn. Nói cách khác, tầng vận tải đảm bảo liên kết giữa các tiến trình trên các máy tính khác nhau trên môi trường mạng. Có hai loại liên

kết: Liên kết có hướng (Connection Oriented) và liên kết vô hướng (Connectionless). Đối với liên kết có hướng, tầng vận tải tạo ra một kết nối logic giữa hai cổng đầu cuối: tất cả dữ liệu của cùng một bản tin được truyền theo đường kết nối đó. Kết nối có hướng gồm ba giai đoạn: thiết lập liên kết, truyền dữ liệu, giải phóng liên kết. Do phải truyền tất cả các dữ liệu trên một kết nối, tầng vận tải còn phải kiểm soát thứ tự truyền, lưu lượng, phát hiện và sửa lỗi. Tầng vận tải đảm nhiệm các chức năng sau:

- Thiết lập liên kết logic giữa các tiến trình trên thiết bị đầu cuối của người dùng
- Phân đoạn và tái hợp
- Kiểm soát kết nối
- Kiểm soát lưu lượng
- Kiểm soát lỗi

+ **Tầng phiên** đóng vai trò kiểm soát viên hội thoại giữa các tiến trình trên lớp ứng dụng qua mạng, nó đảm bảo nhiệm vụ thiết lập, duy trì và đồng bộ hóa tính tương tác giữa các tiến trình đồng cấp trên các máy tính khác nhau. Tầng phiên đảm nhiệm các chức năng sau:

- Kiểm soát hội thoại
- Đồng bộ hóa

+ **Tầng trình diễn** thực hiện các nhiệm vụ liên quan đến cú pháp và ngữ nghĩa của các thông tin được trao đổi giữa hai hệ thống. Tầng trình diễn có nhiệm vụ:

- Mã hóa/Giải mã dữ liệu (Encode/Decode)
- Nén/Giải nén
- Mã hóa/Giải mã bảo mật (Encrypt/Decrypt)

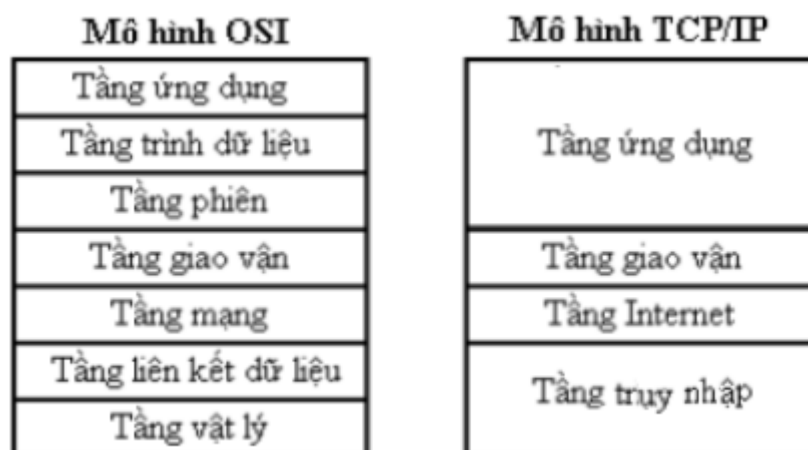
+ **Tầng ứng dụng** cung cấp các tiện ích để người dùng truy cập vào mạng như: các dịch vụ như gửi thư điện tử, truy cập và chuyển file từ xa.... Tầng ứng dụng cũng cung cấp các phương thức cho các ứng dụng khác (ví dụ truy nhập cơ sở

dữ liệu mô hình khách/chủ...). Tầng ứng dụng là tầng cao nhất trong mô hình OSI, do đó nó tạo ra dữ liệu thực sự chứ không có các thông tin điều khiển. Tầng ứng dụng cung cấp các dịch vụ sau:

- Thiết bị đầu cuối ảo của mạng
- Quản lý, truy cập và chuyển tập tin
- Các dịch vụ khác

3) Mô hình TCP/IP. So sánh mô hình TCP/IP và OSI

Mô hình TCP/IP đang được sử dụng rộng rãi nhất hiện nay (mạng Internet đang sử dụng mô hình này), hầu hết tất cả các hệ điều hành đều có cài đặt bộ giao thức TCP/IP. Bộ giao thức này được đặt tên theo hai giao thức chính của nó là giao thức điều khiển vận tải (TCP - Transmission Control Protocol) và giao thức liên mạng (IP - Internet protocol).



Hình 1: Các tầng trong bộ giao thức TCP/IP

Hai mô hình này có liên quan với nhau, nhưng không phải là hoàn toàn giống nhau.

Điểm khác biệt đầu tiên dễ thấy nhất là số lượng của các tầng cấp. Mô hình của Bộ Quốc Phòng Mỹ (DoD model) với bộ giao thức IP chỉ có bốn hoặc năm tầng (tầng liên kết có thể được coi như là một tầng riêng biệt, song cũng có thể được phân tách ra thành hai tầng, tầng vật lý và tầng liên kết dữ liệu) trong khi đó mô hình OSI lại dùng bảy tầng. So sánh tên của chúng một cách

chặt chẽ cho chúng ta thấy rằng, hai tầng tầng trình diễn và tầng phiên đã gộp lại vào tầng ứng dụng.

Các tầng của mô hình OSI không có nhiều chức năng đủ để phản ánh hoạt động của mô hình TCP/IP. Chẳng hạn, cần phải có một tầng nằm giữa tầng mạng và tầng vận tải để chỉ ra nơi tồn tại của giao thức ICMP (Internet Control Message Protocol) và IGMP (Internet Group Management Protocol). Tương tự như vậy cũng cần phải có một tầng ở giữa tầng mạng và tầng liên kết dữ liệu dành cho giao thức ARP (Address Resolution Protocol) và giao thức RARP (Reverse Address Resolution Protocol).

Bảng sau tóm tắt một số giao thức và vị trí của chúng trong mô hình OSI. Để thuận tiện, các phần tiếp theo sẽ trình bày các giao thức theo mô hình TCP/IP (vì đây là mô hình đang được áp dụng trên mạng Internet), đôi khi cũng sẽ dùng mô hình OSI để giải thích nguyên lý làm việc của các thiết bị trên mạng.

Tầng	Mô hình OSI	Mô hình TCP/IP	Giao thức
7	Tầng ứng dụng	Tầng ứng dụng	HTTP, SMTP, SNMP, FTP, Telnet, ECHO, SIP, SSH, NFS, RTSP, XMPP, Whois, ENRP
6	Tầng trình diễn		XDR, ASN.1, SMB, AFP, NCP
5	Tầng phiên		ASAP, TLS, SSH, ISO 8327 / CCITT X.225, RPC, NetBIOS, ASP
4	Tầng vận tải	Tầng vận tải	TCP, UDP, RTP, SCTP, SPX, ATP, IL
3	Tầng mạng	Tầng Internet	IP, ICMP, IGMP, IPX, BGP, OSPF, RIP, IGRP, EIGRP, ARP, RARP, X.25
2	Tầng liên kết dữ liệu	Tầng truy nhập mạng	Ethernet 802.2, Token ring, HDLC, Frame relay, ISDN, ATM, 802.11 WiFi, FDDI, PPP
1	Tầng vật lý		Ethernet 802.3 (10BASE-T, 100BASE-T, 1000BASE-T), SONET/SDH, T-carrier/E-carrier, 802.11

4) Các giao thức thường dùng tại lớp ứng dụng (HTTP, FTP, SMTP, ...)

Giao thức truy nhập trang web HTTP

Hyper Text Transfer Protocol (HTTP) - giao thức tầng ứng dụng của Web – là trái tim của Web. HTTP được triển khai trên cả hai phía máy khách và máy chủ. Các tiến trình máy khách và máy chủ trên các hệ thống đầu cuối khác nhau giao tiếp với nhau thông qua việc trao đổi các bản tin HTTP. HTTP quy định cấu trúc bản tin cũng như cách thức trao đổi bản tin giữa máy khách và máy chủ.

HTTP xác định cách thức trình duyệt yêu cầu trang web từ web máy chủ cũng như cách thức máy chủ gửi trang web được yêu cầu tới trình duyệt. Khi người dùng yêu cầu một đối tượng (ví dụ kích vào một siêu liên kết), browser sẽ gửi một bản tin HTTP tới máy chủ yêu cầu đối tượng đó. Máy chủ nhận được yêu cầu và trả lời bằng cách gửi lại một thông điệp trả lời chứa đối tượng được yêu cầu.

5) Nguyên tắc định tuyến trong mạng

Thực hành: (Mang tính khuyến khích, không bắt buộc)

(Yêu cầu chụp hình, diễn giải câu lệnh, làm rõ các bước thực hành)

- So sánh lệnh ping và traceroute; làm rõ ưu nhược điểm, sự khác nhau

```
C:\Users\HaiNH>ping facebook.com

Pinging facebook.com [2a03:2880:f115:83:face:b00c:0:25de] with 32 bytes of data:
Reply from 2a03:2880:f115:83:face:b00c:0:25de: time=27ms
Reply from 2a03:2880:f115:83:face:b00c:0:25de: time=26ms
Reply from 2a03:2880:f115:83:face:b00c:0:25de: time=27ms
Reply from 2a03:2880:f115:83:face:b00c:0:25de: time=27ms

Ping statistics for 2a03:2880:f115:83:face:b00c:0:25de:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 26ms, Maximum = 27ms, Average = 26ms
```

```
C:\Users\HaiNH>tracert facebook.com

Tracing route to facebook.com [2a03:2880:f115:83:face:b00c:0:25de]
over a maximum of 30 hops:
  1    1 ms    <1 ms    <1 ms    vnpt.vn [2001:ee0:4081:d980:4e12:e8ff:fea9:f5e0]
  2    2 ms     2 ms     2 ms    edge-star-mini6-shv-01-hkg1.facebook.com [2a03:2880:f115:83:face:b00c:0:25de]

Trace complete.
```

Tiêu chí	ping	traceroute
Mục đích	Kiểm tra xem máy đích có phản hồi hay không	Xác định đường đi qua các router trung gian đến máy đích
Giao thức sử dụng	ICMP Echo Request / Reply	UDP (Linux/macOS) hoặc ICMP (Windows) với TTL tăng dần
Cách hoạt động	Gửi gói ICMP đến đích và đợi phản hồi	Gửi gói tin với TTL tăng dần để phát hiện các bước nhảy (hops)
Ưu điểm	<ul style="list-style-type: none"> - Đơn giản, nhanh chóng - Dễ sử dụng - Xác minh nhanh tình trạng kết nối 	<ul style="list-style-type: none"> - Hiển thị chi tiết từng bước của đường đi - Hữu ích để xác định chỗ tắc hoặc chặn trong mạng
Nhược điểm	<ul style="list-style-type: none"> - Không hiển thị đường đi - Không biết bị chặn ở đâu nếu không có phản hồi 	<ul style="list-style-type: none"> - Chậm hơn - Một số router có thể chặn gói ICMP/UDP, gây kết quả không chính xác - Phức tạp hơn một chút

- Trên hệ điều hành Ubuntu, thực hành cài đặt và cấu hình dịch vụ DNS;

Bước 1: Cài đặt BIND9

```
root@e6c043718e13:/# apt install -y bind9 bind9utils bind9-doc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Diễn giải: Dùng để cài đặt toàn bộ phần mềm và công cụ cần thiết của BIND9, nhằm thiết lập một máy chủ DNS trên hệ thống Ubuntu.

Máy chủ DNS (BIND9) để hệ thống có thể phân giải tên miền thành địa chỉ IP và ngược lại.

Bước 2: Cấu hình zone trong BIND9

```
root@e6c043718e13:/# nano /etc/bind/named.conf.local
```

```
//
// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "zone.local" {
    type master;
    file "/etc/bind/db.zone.local";
};
```

Diễn giải: Khai báo tên zone, đường dẫn đến file dữ liệu chứa các bản ghi DNS của zone đó và thông báo cho BIND biết về domain (zone) mà nó phải quản lý.

Bước 3: Tạo file zone dữ liệu db.zone.local

1. Copy file mẫu:

```
root@e6c043718e13:/# cp /etc/bind/db.local /etc/bind/db.zone.local
```

2. Chỉnh sửa file zone:

```
root@e6c043718e13:/# nano /etc/bind/db.zone.local
```

```
$TTL      604800
@         IN      SOA      ns1.zone.local. admin.zone.local. (
                                2           ; Serial
                                604800      ; Refresh
                                86400       ; Retry
                                2419200     ; Expire
                                604800 )    ; Negative Cache TTL
;
@         IN      NS       ns1.zone.local.
ns1       IN      A        127.0.0.1
www       IN      A        127.0.0.1
```

Diễn giải: Đây là file chứa các bản ghi DNS cụ thể cho zone đã khai báo.

Đây chính là nơi BIND trả lời các truy vấn DNS

Bước 4: Kiểm tra cấu hình zone

```
root@e6c043718e13:/# named-checkconf
root@e6c043718e13:/# named-checkzone zone.local /etc/bind/db.zone.local
zone zone.local/IN: loaded serial 2
OK
```

Diễn giải: Kiểm tra file cấu hình tổng thể và kiểm tra file dữ liệu zone không gặp lỗi cú pháp và giúp tránh lỗi khi khởi động dịch vụ DNS

Bước 5: Khởi động lại BIND9

```
root@e6c043718e13:/# named -g -c /etc/bind/named.conf
```

Diễn giải: Áp dụng các thay đổi đã cấu hình. Khởi động lại dịch vụ DNS để đọc các cấu hình mới.

Bước 6: Kiểm tra hoạt động DNS

```
root@e6c043718e13:/# apt install dnsutils -y
```

Kết quả:

```
;; ANSWER SECTION:
www.zone.local.      604800  IN      A       127.0.0.1
```

Phần 2: Câu hỏi lấy điểm bài tập tự luận

1) So sánh giao thức TCP và UDP;

TCP là một giao thức tầng vận tải cung cấp dịch vụ truyền dữ liệu tin cậy, nó sử dụng nhiều nguyên lý để đảm bảo truyền tính tin cậy như: phát hiện lỗi, đánh số thứ tự các đoạn tin, cơ chế xác nhận. Giao thức TCP thuộc loại kết nối có hướng vì trước khi gửi dữ liệu của lớp ứng dụng phải có thủ tục “Bắt tay”, nghĩa là chúng phải gửi một số đoạn tin đặc biệt để xác định các tham số đảm bảo cho quá trình truyền dữ liệu. Giao thức TCP có các đặc điểm sau:

- Định hướng kết nối: Trước khi truyền dữ liệu phải thực hiện thủ tục thiết lập liên kết, sau khi truyền dữ liệu xong thì một trong hai bên hoặc cả hai bên gửi tín hiệu yêu cầu hủy bỏ liên kết.
- Đánh số tuần tự: Mỗi đoạn tin trước khi gửi đi phải được đánh số tuần tự, dựa vào cơ chế này mà bên nhận sẽ sắp xếp lại các đoạn tin chính xác và đồng thời phát hiện được những đoạn tin bị thất lạc.
- Đảm bảo tính tin cậy: Lỗi có thể xảy ra khi một đoạn tin nào đó bị thất lạc hoặc nội dung đoạn tin bị thay đổi. Trong cả hai trường hợp trên, giao thức TCP sẽ yêu cầu gửi lại.
- Điều khiển lưu lượng: năng lực xử lý của mỗi máy tính chỉ có hạn nhất định (thường phụ thuộc ba tài nguyên cơ bản trong mỗi máy tính: tốc độ CPU, dung

lượng bộ nhớ, tốc độ đọc/ghi thiết bị lưu trữ, ngoài ra có thể xét thêm các yếu tố khác như: số lượng người dùng, băng thông mạng...). Nếu không có cơ chế điều khiển lưu lượng thì bên nhận có thể không kịp xử lý các đoạn tin gửi đến và dẫn đến tình trạng lỗi như: tràn bộ nhớ, treo hệ thống.

UDP là giao thức thuộc tầng vận tải, ngoài chức năng ghép kênh/phân kênh, UDP có thêm cơ chế phát hiện lỗi đơn giản (chức năng phát hiện và sửa lỗi sẽ được chuyển lên tầng ứng dụng). Có thể nói, nếu sử dụng UDP thì gần như ứng dụng làm việc trực tiếp với tầng mạng IP. UDP lấy bản tin từ tiến trình ứng dụng, chèn thêm một số trường điều khiển, trong đó có hai trường địa chỉ cổng nguồn và đích cho chức năng Ghép kênh/phân kênh để tạo nên dữ liệu (gọi là datagram) và được chuyển xuống tầng mạng. Tầng mạng đặt datagram này trong gói tin và cố gắng gửi gói IP tới máy tính nhận. UDP không đòi hỏi bên gửi và bên nhận phải thiết lập liên kết trước khi trao đổi dữ liệu. Vì vậy UDP được xem là giao thức kết nối vô hướng hay không liên kết. Giao thức UDP dường như không có có nhiều ưu điểm như giao thức TCP: truyền dữ liệu tin cậy, kiểm soát lưu lượng..., tuy nhiên trên thực tế giao thức UDP được sử dụng nhiều hơn vì những đặc điểm sau:

- **Không cần thiết lập liên kết:** Giao thức TCP sử dụng cơ chế “bắt tay” ba bước trước khi truyền dữ liệu nhưng giao thức UDP không cần cơ chế. Vì vậy UDP sẽ không phải mất thời gian để thiết lập đường truyền. Đây chính là nguyên nhân dịch vụ DNS chạy trên UDP chứ không phải là TCP, DNS sẽ chạy chậm nếu sử dụng TCP. HTTP sử dụng TCP vì các đối tượng Web cần được tải về chính xác, do đó yêu cầu một đường truyền tin cậy.

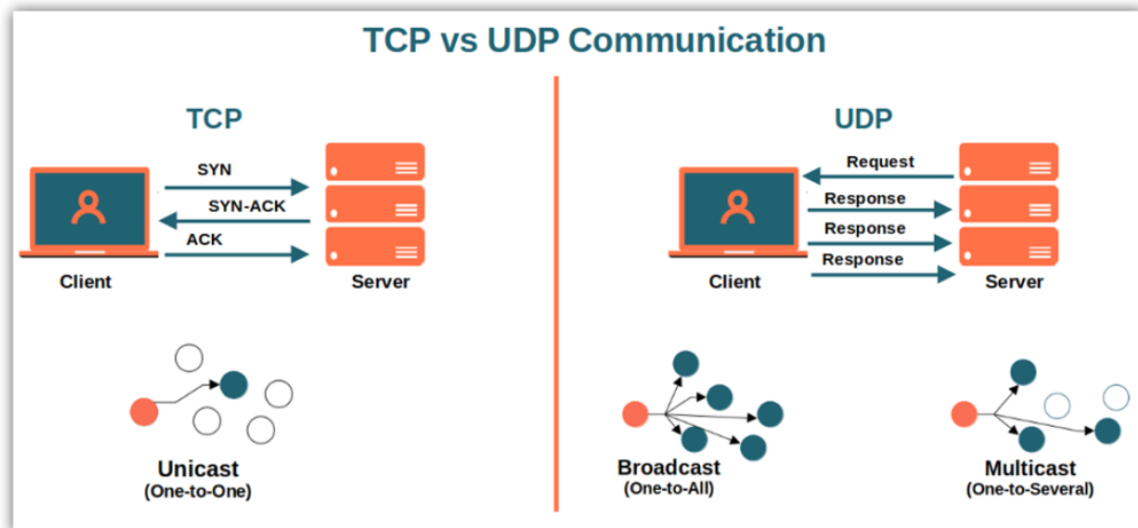
- **Không duy trì trạng thái kết nối.** TCP ghi nhớ trạng thái kết nối trên hệ thống đầu cuối. Trạng thái kết nối bao gồm vùng đệm của bên nhận và bên gửi, các tham số kiểm soát tắc nghẽn, số tuần tự phát và số biên nhận. Các thông số đó giúp TCP triển khai dịch vụ truyền tin tin cậy và cơ chế kiểm soát tắc nghẽn.

UDP không phải lưu giữ những thông tin như vậy, do đó nếu phía máy chủ sử dụng UDP thì có khả năng phục vụ đồng thời nhiều máy khách hơn.

- **Thông tin điều khiển ít hơn:** Thông tin điều khiển của đoạn tin TCP tối thiểu là 20 bytes trong khi UDP chỉ có 8 bytes.

- **Không kiểm soát tốc độ gửi.** TCP có cơ chế kiểm soát tắc nghẽn, điều chỉnh tốc độ gửi khi xảy ra tắc nghẽn. Cơ chế điều chỉnh này có thể ảnh hưởng tới những ứng dụng thời gian thực, đó là những ứng dụng chấp nhận mất mát dữ liệu nhưng lại đòi hỏi phải có một tốc độ truyền tối thiểu. Tốc độ truyền dữ liệu của UDP chỉ bị giới hạn bởi tốc độ sinh dữ liệu của tầng ứng dụng, khả năng máy tính nguồn (CPU, tốc độ đồng hồ) và tốc độ truy nhập mạng. Bên nhận không nhất thiết phải nhận toàn bộ dữ liệu, khi nghẽn mạng, một phần dữ liệu có thể bị mất do tràn vùng đệm ở các thiết bị mạng, tốc độ nhận có thể bị giới hạn do tắc nghẽn ngay cả khi tốc độ gửi không bị hạn chế.

Các ứng dụng phổ biến như: Email, truy cập từ xa, Web và truyền tập tin chạy trên nền TCP vì chúng cần đến dịch vụ truyền dữ liệu tin cậy. Tuy nhiên có một số ứng dụng khác thích hợp với UDP hơn TCP như: Giao thức cập nhật bảng định tuyến RIP sử dụng UDP, bởi vì việc cập nhật được thực hiện định kỳ, cho nên dù cập nhật bị mất nhưng sẽ có cập nhật mới sau một khoảng thời gian ngắn. UDP được sử dụng để gửi dữ liệu quản trị mạng SNMP, trong trường hợp này UDP thích hợp hơn TCP vì các tiến trình quản trị mạng thường hoạt động khi mạng có sự cố không thể truyền dữ liệu chính xác hay cơ chế kiểm soát tắc nghẽn không làm việc. DNS sử dụng UDP, do đó có thể tránh được thời gian trễ của giai đoạn thiết lập kết nối.



2) Định dạng địa chỉ IPv4/ IPv6;

Định dạng địa chỉ IPv4 là Địa chỉ IP có độ dài 32 bit (4 byte) và như vậy không gian địa chỉ có 232 địa chỉ IP. Địa chỉ IP được viết theo ký pháp dấu chấm thập phân. Mỗi byte của địa chỉ được viết dưới dạng thập phân và phân cách với các byte khác bằng ký tự chấm (.). Xét địa chỉ IP 193.32.216.9, giá trị 193 là số thập phân ứng với nhóm 8 bit đầu của địa chỉ, 32 là số thập phân ứng với nhóm 8 bit thứ hai của địa chỉ, bởi vậy địa chỉ 193.32.216.9 trong ký pháp nhị phân sẽ là 110000001 00100000 11011000 00001001.

Định dạng địa chỉ IPv6 là IP v6 tăng kích thước địa chỉ IP từ 32 bit lên 128 bit. Với độ dài 128 bit có thể tạo được 2128 địa chỉ sẽ đảm bảo khả năng không bị thiếu địa chỉ IP. Bên cạnh địa chỉ duy nhất (unicast) và địa chỉ nhóm (multicast), IPv6 còn có một dạng địa chỉ mới gọi là “anycast”, cho phép một gói tin với địa chỉ đích thuộc kiểu “anycast” có thể được chuyển từ một nhóm các máy tính (đặc điểm này sẽ được sử dụng ví dụ khi gửi thông điệp HTTP GET tới nhiều trang phụ (mirror site) chứa cùng một tài liệu nào đấy).

3) Các giao thức định tuyến trên Internet (RIP, OSPF, BGP)

Hiện nay có 3 giao thức định tuyến nội miền được sử dụng rộng rãi: RIP (Routing Information Protocol), OSPF (Open Shortest Path First) và EIGRP

(Cisco propriety Enhanced Interior Gateway Routing Protocol). Giao thức định tuyến liên miền thường được sử dụng là giao thức BGP.

Giao thức RIP: là một trong những giao thức định tuyến nội miền đầu tiên, nó có một số đặc điểm sau:

- Định tuyến nội miền: Cho phép trao đổi thông tin giữa các thiết bị định tuyến trong một miền.
- Đo khoảng cách bằng chặng đường đi: Giá đường đi giữa hai thiết bị đầu cuối được xác định bằng số lượng các thiết bị định tuyến trung gian trên đường đi đó. Độ dài tối đa của một tuyến đường là 15, nghĩa là đường kính tối đa của một miền là 15 thiết bị định tuyến.
- Truyền thông không tin cậy: RIP sử dụng UDP để chuyển bản tin về các tuyến đường.
- Gửi tin dạng quảng bá và nhóm: RIP v1 sử dụng cách truyền quảng bá khi truyền giữa hai thiết bị định tuyến, RIP v2 cho phép truyền theo chế độ nhóm.
- Thuật toán vector khoảng cách: RIP sử dụng thuật toán vector khoảng cách. Các thiết bị định tuyến hàng xóm trao đổi bảng định tuyến cho nhau 90s một lần (có thể thay đổi chu kỳ này) trong các bản tin RIP (RIP response message, RIP advertisement), mỗi bản tin chứa tối đa 25 địa chỉ đích.
- Các máy tính có thể thụ động nhận thông tin từ các thiết bị định tuyến: RIP cho phép các thiết bị đầu cuối (chủ yếu là máy tính) lắng nghe và cập nhật bảng định tuyến. Điều này đặc biệt hữu dụng với các mạng có nhiều thiết bị định tuyến. Khi đó máy tính trong mạng có thể dễ dàng xác định được thiết bị định tuyến cần chuyển tới.

Giao thức OSPF: giao thức RIP chỉ phù hợp với các mạng có kích cỡ nhỏ. Khi một tổ chức mạng tương đối lớn, người ta cần phải đưa ra giao thức phù hợp hơn. IETF đã đưa ra giao thức OSPF với các đặc điểm sau:

- Định tuyến nội miền: Cho phép trao đổi thông tin giữa các thiết bị định tuyến trong một miền.

- Hỗ trợ CIDR: hỗ trợ việc phân mạng, tạo lập mạng con.
- Trao đổi các thông tin đã được kiểm chứng. Hai thiết bị định tuyến trao đổi bản tin OSPF với nhau có thể tiến hành thủ tục kiểm tra để xác định mình nhận được bản tin từ đúng đối tượng, điều này ngăn ngừa các cuộc tấn công bằng phương pháp giả mạo.
- Sử dụng thuật toán tìm đường dựa trên trạng thái kênh truyền.
- Hỗ trợ phân cấp trong miền. Ưu điểm chính của OSPF là cho phép tiếp tục phân một miền thành nhiều miền con.

Giao thức BGP được mô tả trong các khuyến nghị RFC 1771, 1772, 1773, được xem là một chuẩn hiển nhiên trong định tuyến giữa các vùng trên mạng Internet. Nhiệm vụ của nó là định tuyến giữa các vùng được quản trị độc lập với nhau. BGP có những đặc điểm sau:

- Định tuyến liên vùng: BGP cho phép cung cấp các thông tin định tuyến giữa các vùng, mỗi tuyến đường được xem là một chuỗi các AS liên tiếp nhau.
- Hỗ trợ việc thiết lập chính sách: Người quản trị có thể áp dụng những chính sách nào đó, ví dụ hạn chế việc quảng cáo ra bên ngoài.
- Truyền thông tin cậy: Hai thực thể BGP sử dụng kết nối TCP để trao đổi bản tin.

4) Giao thức đa truy nhập

Thực hành: (Mang tính khuyến khích, không bắt buộc)

(Yêu cầu chụp hình, diễn giải câu lệnh, làm rõ các bước thực hành)

- Cấu hình và cài đặt dịch vụ Email SMTP trên máy tính hệ điều hành Ubuntu;
- Thực hiện việc gửi và nhận thư;

Bước 1: Cập nhật, cài đặt công cụ cần thiết và cấu hình postfix cơ bản

```
root@5d6a6b4f6d59:/# apt update && apt install -y postfix mailutils vim net-tools
```

Cập nhật các gói phần mềm và cài đặt các công cụ cần thiết cho việc cấu hình và cài đặt dịch vụ Email SMTP bao gồm:

- postfix: Phần mềm máy chủ gửi mail chính (MTA – Mail Transfer Agent).
- mailutils: Bộ công cụ gửi/nhận mail từ dòng lệnh, giúp kiểm tra hoạt động của Postfix
- Trình soạn thảo văn bản để chỉnh sửa file cấu hình
- net-tools: Bộ công cụ mạng cổ điển (bao gồm lệnh ifconfig, netstat...), hữu ích trong việc kiểm tra kết nối và IP máy chủ.

```
1. No configuration 2. Internet Site 3. Internet with smarthost 4. Satellite system 5. Local only
General mail configuration type: 5
```

Chọn kiểu cấu hình gửi mail. Ở đây chỉ xử lý mail nội bộ trong hệ thống hoặc container.

```
System mail name: mailserver.local
```

Đặt tên cho mail server

```
Please select the geographic area in which you live. Subsequent configuration questions will narrow this down by
presenting a list of cities, representing the time zones in which they are located.
```

```
1. Africa 3. Antarctica 5. Asia 7. Australia 9. Indian 11. Etc
2. America 4. Arctic 6. Atlantic 8. Europe 10. Pacific 12. Legacy
Geographic area: 5
```

```
Please select the city or region corresponding to your time zone.
```

```
1. Aden 16. Brunei 31. Hong_Kong 46. Kuala_Lumpur 61. Pyongyang 76. Tel_Aviv
2. Almaty 17. Chita 32. Hovd 47. Kuching 62. Qatar 77. Thimphu
3. Amman 18. Choibalsan 33. Irkutsk 48. Kuwait 63. Qostanay 78. Tokyo
4. Anadyr 19. Chongqing 34. Istanbul 49. Macau 64. Qyzylorda 79. Toms
5. Aqtau 20. Colombo 35. Jakarta 50. Magadan 65. Riyadh 80. Ulaanbaatar
6. Aqtobe 21. Damascus 36. Jayapura 51. Makassar 66. Sakhalin 81. Urumqi
7. Ashgabat 22. Dhaka 37. Jerusalem 52. Manila 67. Samarkand 82. Ust-Nera
8. Atyrau 23. Dili 38. Kabul 53. Muscat 68. Seoul 83. Vientiane
9. Baghdad 24. Dubai 39. Kamchatka 54. Nicosia 69. Shanghai 84. Vladivostok
10. Bahrain 25. Dushanbe 40. Karachi 55. Novokuznetsk 70. Singapore 85. Yakutsk
11. Baku 26. Famagusta 41. Kashgar 56. Novosibirsk 71. Srednekolymsk 86. Yagon
12. Bangkok 27. Gaza 42. Kathmandu 57. Omsk 72. Taipei 87. Yekaterinburg
13. Barnaul 28. Harbin 43. Khandyga 58. Oral 73. Tashkent 88. Yerevan
14. Beirut 29. Hebron 44. Kolkata 59. Phnom_Penh 74. Tbilisi
15. Bishkek 30. Ho_Chi_Minh 45. Krasnoyarsk 60. Pontianak 75. Tehran
Time zone: 30
```

Cài đặt múi giờ cho hệ thống theo múi giờ Việt Nam (Asia/Ho_Chi_Minh – UTC+7).

Mục đích: Log hệ thống, email hiển thị đúng thời gian và đồng bộ thời gian giữa các dịch vụ liên quan đến mail.

Bước 2: Kiểm tra và cấu hình Postfix

```
root@5d6a6b4f6d59:/# vim /etc/postfix/main.cf
```

```

myhostname = mailserver.local
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = $myhostname, localhost.localdomain, localhost
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = ipv4

```

Bước 3: Khởi động dịch vụ Postfix trong Docker

```

root@5d6a6b4f6d59:/# postfix start
postfix/postfix-script: starting the Postfix mail system

```

Bước 4: Tạo người dùng để gửi/nhận email nội bộ

```

root@5d6a6b4f6d59:/# adduser hhnguyen1196
info: Adding user `hhnguyen1196' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `hhnguyen1196' (1001) ...
info: Adding new user `hhnguyen1196' (1001) with group `hhnguyen1196 (1001)' ...
info: Creating home directory `/home/hhnguyen1196' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for hhnguyen1196
Enter the new value, or press ENTER for the default
    Full Name []: Hoang Hai
    Room Number []: 1
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
info: Adding new user `hhnguyen1196' to supplemental / extra groups `users' ...
info: Adding user `hhnguyen1196' to group `users' ...

```

```

root@5d6a6b4f6d59:/# adduser hhnguyen3011
info: Adding user `hhnguyen3011' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `hhnguyen3011' (1002) ...
info: Adding new user `hhnguyen3011' (1002) with group `hhnguyen3011 (1002)' ...
info: Creating home directory `/home/hhnguyen3011' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for hhnguyen3011
Enter the new value, or press ENTER for the default
    Full Name []: Hoang Hai 3011
    Room Number []: 1
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
info: Adding new user `hhnguyen3011' to supplemental / extra groups `users' ...
info: Adding user `hhnguyen3011' to group `users' ...

```

Bước 5: Gửi email từ root đến user hhnguyen1196 và thực hiện kiểm tra thư đến với user hhnguyen1196

```
root@5d6a6b4f6d59:/# echo "Hello there. Welcome to the mail server" | mail -s "Try sending mail from root" hhnguyen1196
```

Chuyển sang user hhnguyen1196 và xem email

```
root@5d6a6b4f6d59:/# su - hhnguyen1196
```

```
hhnguyen1196@5d6a6b4f6d59:~$ mail
"/var/mail/hhnguyen1196": 1 message 1 new
>N 1 root Sun Oct 12 07:21 13/485 Try sending mail from root
?
Return-Path: <root@5d6a6b4f6d59>
X-Original-To: hhnguyen1196
Delivered-To: hhnguyen1196@mailserver.local
Received: by mailserver.local (Postfix, from userid 0)
        id 5CD29E3D; Sun, 12 Oct 2025 00:21:07 +0000 (UTC)
Subject: Try sending mail from root
To: hhnguyen1196@mailserver.local
User-Agent: mail (GNU Mailutils 3.17)
Date: Sun, 12 Oct 2025 07:21:07 +0700
Message-Id: <20251012002107.5CD29E3D@mailserver.local>
From: root <root@5d6a6b4f6d59>

Hello there. Welcome to the mail server
? |
```

Tương tự thực hiện gửi thư từ hhnguyen1196 đến user hhnguyen3011 và kiểm tra hộp thư

```
hhnguyen1196@5d6a6b4f6d59:~$ echo "Chao ban! Toi la hhnguyen1196" | mail -s "Email from hhnguyen1196" hhnguyen3011
hhnguyen1196@5d6a6b4f6d59:~$ su - hhnguyen3011
Password:
hhnguyen3011@5d6a6b4f6d59:~$ mail
"/var/mail/hhnguyen3011": 1 message 1 new
>N 1 Hoang Hai Sun Oct 12 07:26 13/496 Email from hhnguyen1196
?
Return-Path: <hhnguyen1196@5d6a6b4f6d59>
X-Original-To: hhnguyen3011
Delivered-To: hhnguyen3011@mailserver.local
Received: by mailserver.local (Postfix, from userid 1001)
        id 268D1E40; Sun, 12 Oct 2025 00:26:46 +0000 (UTC)
Subject: Email from hhnguyen1196
To: hhnguyen3011@mailserver.local
User-Agent: mail (GNU Mailutils 3.17)
Date: Sun, 12 Oct 2025 07:26:46 +0700
Message-Id: <20251012002646.268D1E40@mailserver.local>
From: Hoang Hai <hhnguyen1196@5d6a6b4f6d59>

Chao ban! Toi la hhnguyen1196
```