



# TRƯỜNG ĐẠI HỌC THỦY LỢI

KHOA CÔNG NGHỆ THÔNG TIN

Bộ môn: Kỹ thuật máy tính và mạng

## MÔN HỌC: MẠNG MÁY TÍNH

Giảng viên: Trần Văn Hội

Email: [hoitv@tlu.edu.vn](mailto:hoitv@tlu.edu.vn)

Điện thoại: 0944.736.007

# NỘI DUNG MÔN HỌC



Chương 1: Tổng quan về mạng máy tính

Chương 2: Tầng ứng dụng

Chương 3: Tầng giao vận

Chương 4: Tầng mạng

Chương 5: Tầng truy nhập mạng

# CHƯƠNG 1: TỔNG QUAN VỀ MẠNG MÁY TÍNH

1

- Các giao thức tầng ứng dụng

2

- Dịch vụ Web và giao thức HTTP

3

- Giao thức truyền tệp FTP

4

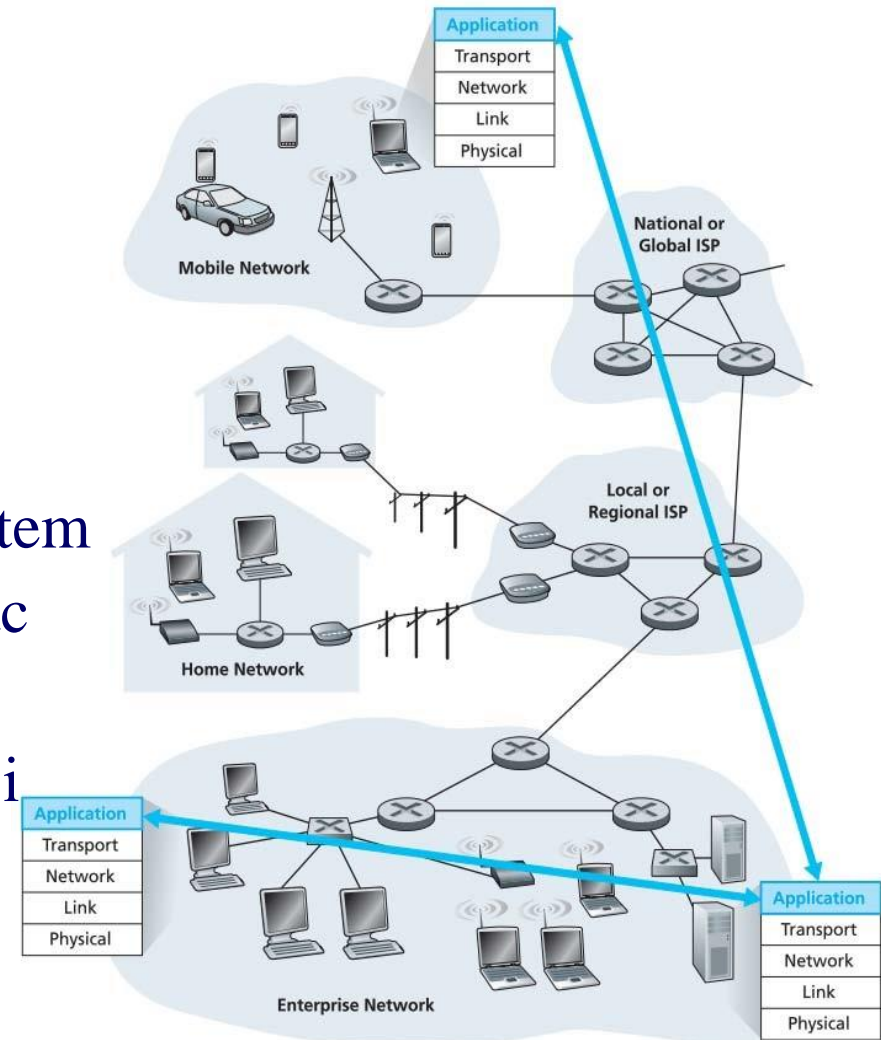
- Giao thức thư điện tử E-mail

5

- Giao thức DNS

# GIAO THỨC ỨNG DỤNG

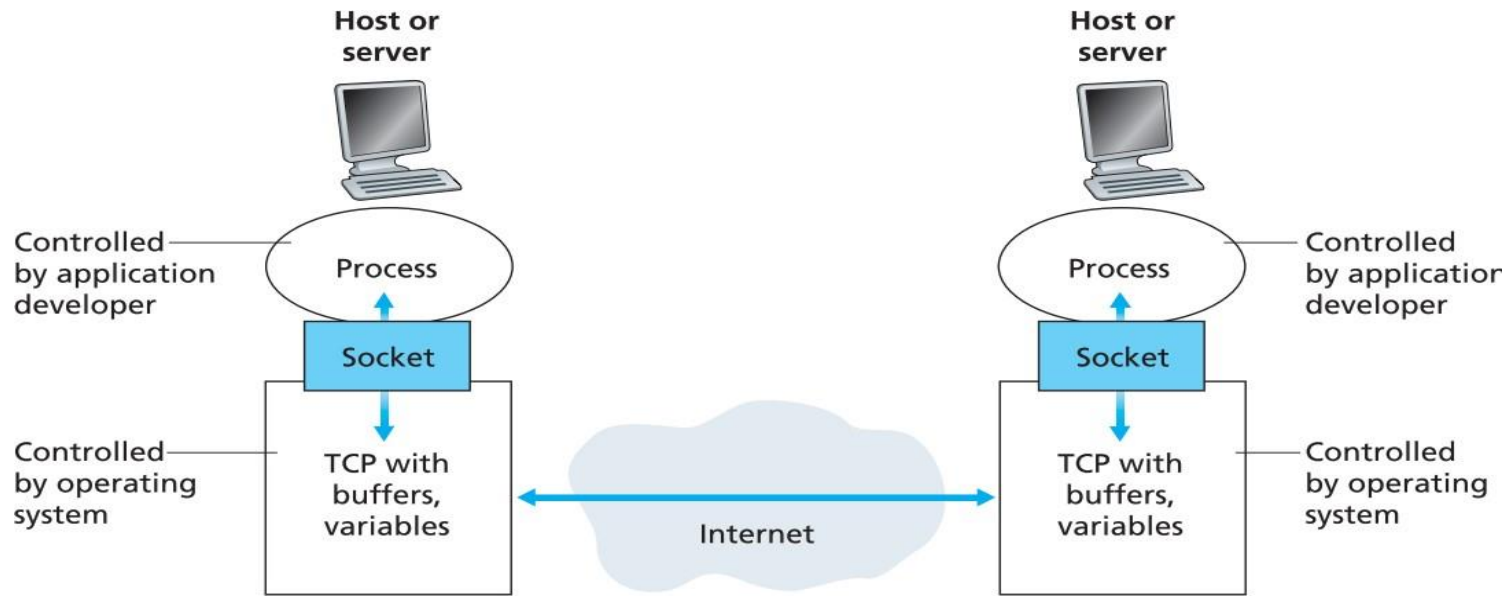
- ❖ Giao thức ứng dụng
  - Định nghĩa các quy tắc giao tiếp
  - Sử dụng các dịch vụ giao vận (TCP/UDP...)
- ❖ Ứng dụng mạng (network application)
  - Chương trình chạy trên End System và giao tiếp với End System khác qua mạng
  - Hoạt động trên các hệ thống cuối
- ❖ Ví dụ ứng dụng/giao thức
  - Web (HTTP)
  - Mail (SMTP/POP/IMAP)



# CÁC THÀNH PHẦN CỦA ỨNG DỤNG

- ❖ **Chương trình người sử dụng (NSD)**
  - ❖ Hỗ trợ NSD để làm việc với ứng dụng
  - ❖ Ví dụ: trình duyệt Web (Firefox, IE), phần mềm thư điện tử (Outlook, ..)
- ❖ **Chương trình máy chủ:** Cung cấp dịch vụ cho người sử dụng
- ❖ **Tiến trình ứng dụng:** Chương trình đang chạy trên hệ điều hành
- ❖ **Các tiến trình giao tiếp?**
  - ❖ Trên một máy: IPC (inter-process communication)
  - ❖ Giữa các máy trên mạng: qua socket

# GIAO TIẾP GIỮA CÁC TIẾN TRÌNH TRÊN INTERNET



- ❖ Tiến trình gửi thông điệp và nhận thông điệp từ mạng qua một giao diện phần mềm gọi là socket
  - ❖ Tiến trình Client: tiến trình khởi tạo giao tiếp
  - ❖ Tiến trình Server: tiến trình chờ để tiến trình khác liên lạc
- ❖ Socket là giao diện giữa tiến trình ứng dụng và tầng giao vận

# GIAO TIẾP GIỮA CÁC TIẾN TRÌNH TRÊN INTERNET

## ❖ Địa chỉ tiến trình

❖ Để một tiến trình chạy trên một host gửi gói tin tới một tiến trình trên một host khác, tiến trình nhận cần có địa chỉ

❖ Địa chỉ của host

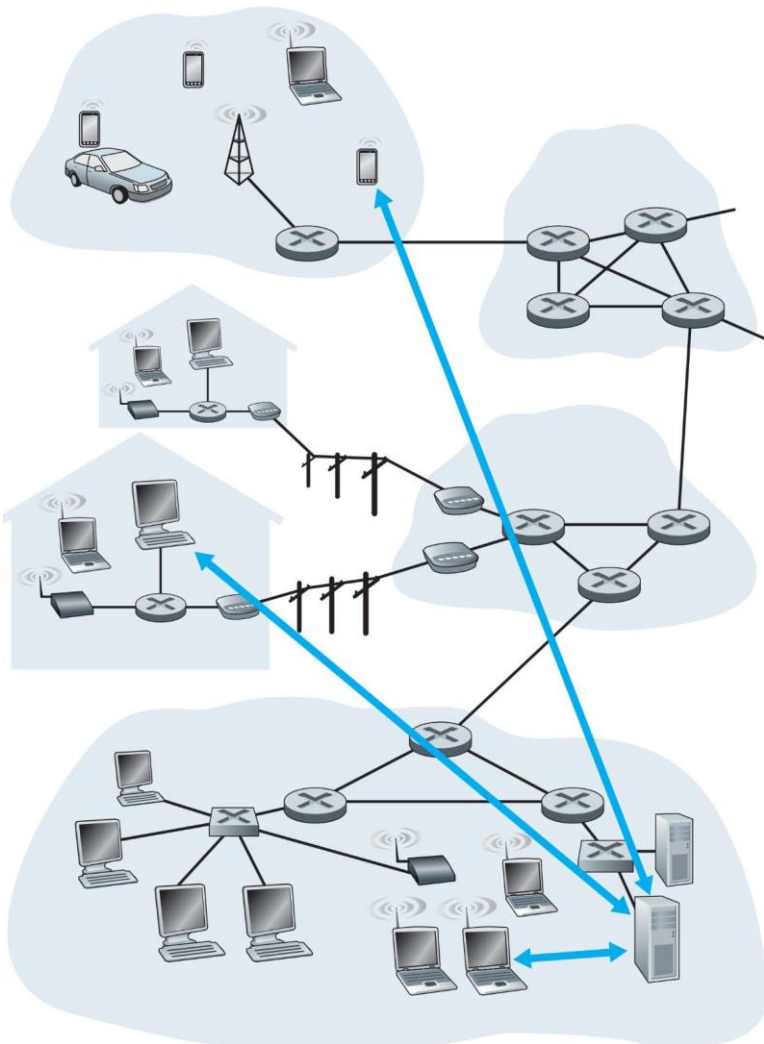
❖ Định danh xác định tiến trình nhận trong host đích

## ❖ Trong Internet

❖ Host được xác định bởi IP address

❖ Tiến trình nhận (hay socket nhận) chạy trên host đích được xác định bởi Port Number. Tại sao lại cần thông tin này?

# MÔ HÌNH KHÁCH – CHỦ



## ❖ Khách

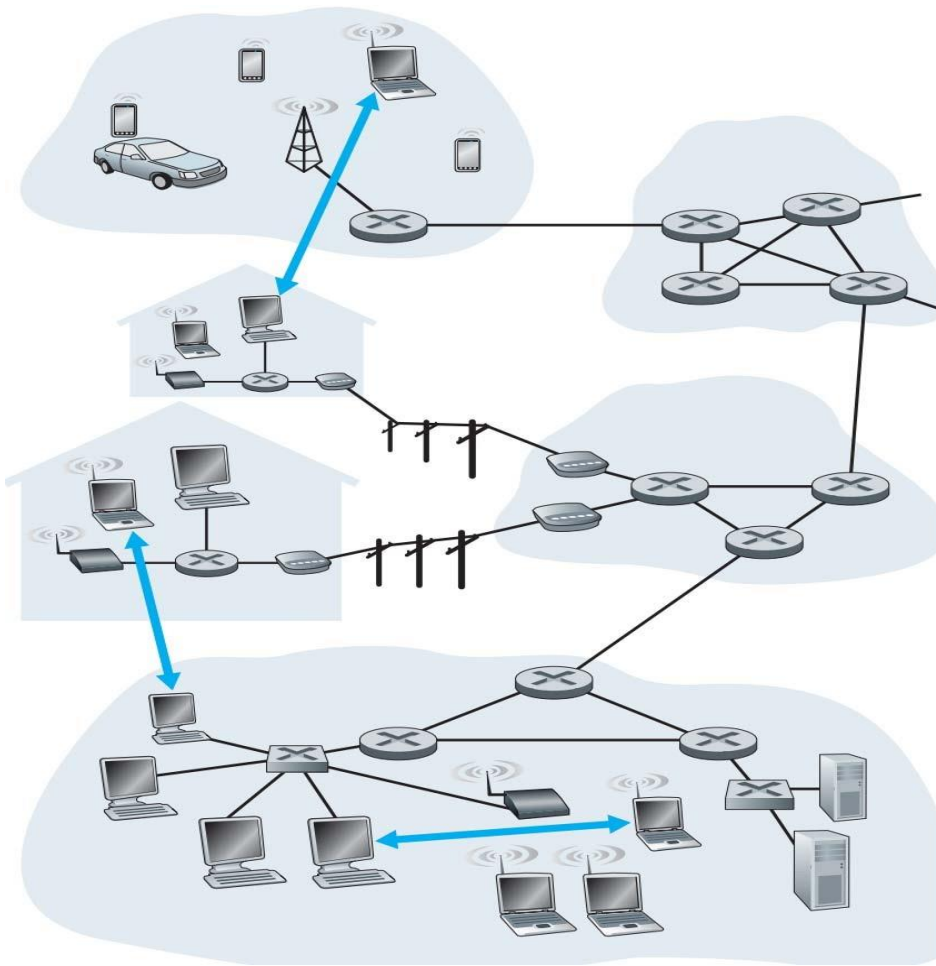
- ❖ Gửi yêu cầu truy cập dịch vụ đến máy chủ
- ❖ các khách không trực tiếp giao tiếp với nhau

## ❖ Chủ

- ❖ Thường xuyên mở để chờ yêu cầu đến từ máy khách
- ❖ Có thể có máy chủ dự phòng để nâng cao hiệu năng, phòng sự cố
- ❖ Ví dụ: Web, e-mail, FTP, ...

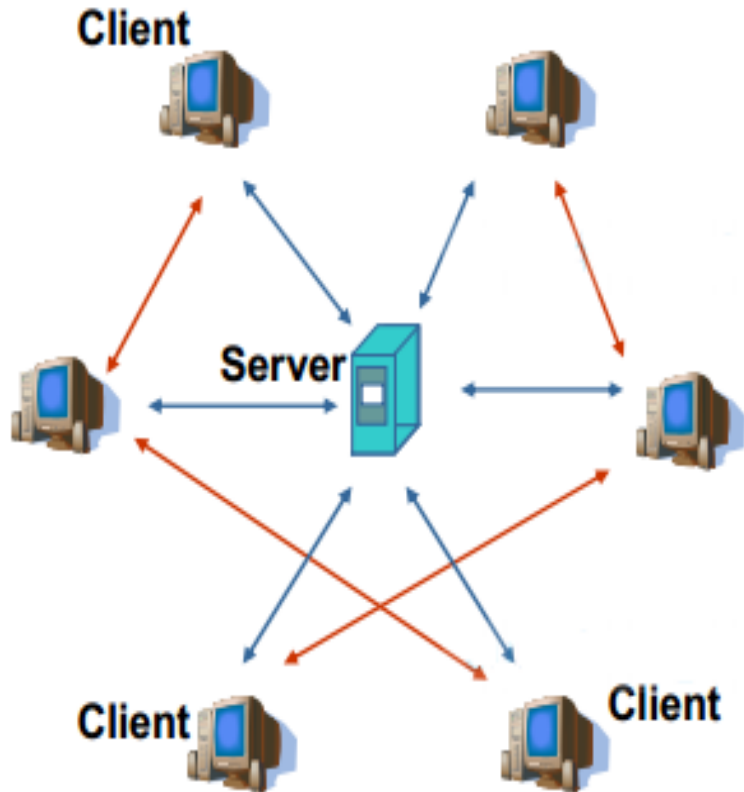


# MÔ HÌNH NGANG HÀNG



- ❖ Không có máy chủ trung tâm
- ❖ Các máy có vai trò ngang nhau
- ❖ Ứng dụng khai thác trao đổi trực tiếp giữa cặp host khi có kết nối, gọi là peer
- ❖ Ví dụ: chia sẻ tập tin (như BitTorrent)...

# MÔ HÌNH LAI



- ❖ Một máy chủ trung tâm để quản lý NSD, thông tin tìm kiếm
- ❖ Các máy khách sẽ trực tiếp giao tiếp với nhau sau khi đăng nhập
- ❖ Ví dụ: Skype
  - ❖ Máy chủ Skype quản lý các phiên đăng nhập, mật khẩu, ..
  - ❖ Sau khi kết nối, các máy sẽ gọi VoIP trực tiếp cho nhau

# CÁC YÊU CẦU CỦA ỨNG DỤNG

- ❖ Ứng dụng yêu cầu các dịch vụ gì từ tầng giao vận?
- ❖ Truyền dữ liệu tin cậy
- ❖ Thông lượng
- ❖ Thời gian
- ❖ An toàn bảo mật

# CÁC YÊU CẦU CỦA ỨNG DỤNG

- ❖ Truyền dữ liệu tin cậy (**reliable data transfer**)
  - ❖ Gói tin có thể bị thất lạc trong mạng máy tính. Khi nào?
  - ❖ Đối với nhiều ứng dụng, thất lạc dữ liệu có thể gây hậu quả không chấp nhận được. Ứng dụng nào? Tại sao?
  - ❖ Dịch vụ truyền dữ liệu đảm bảo (tin cậy): có cơ chế để đảm bảo dữ liệu được chuyển đúng và đầy đủ tới ứng dụng nhận
  - ❖ Khi giao thức tầng giao vận không cung cấp dịch vụ truyền tin cậy, dữ liệu gửi bởi tiến trình gửi có thể không tới tiến trình nhận. Điều này có thể chấp nhận được đối với các ứng dụng chấp nhận mất tin (**loss-tolerant application**). Ví dụ loss-tolerant application?

# CÁC YÊU CẦU CỦA ỨNG DỤNG

## ❖ Thông lượng

- ❖ Thông lượng là tốc độ bit truyền từ bên gửi tới bên nhận.
- ❖ Thông lượng khả dụng có thể thay đổi theo thời gian. Tại sao?
- ❖ Tầng giao vận có thể cung cấp dịch vụ đảm bảo thông lượng khả dụng.
- ❖ Ứng dụng cần đảm bảo yêu cầu thông lượng gọi là ứng dụng tác động bởi băng thông (**bandwidth-sensitive applications**)
  - ❖ Ví dụ, ứng dụng điện thoại Internet cần 32 kbps
- ❖ **Elastic application** có thể sử dụng băng thông ít hoặc nhiều tùy theo thông lượng khả dụng. Ví dụ elastic apps?

# CÁC YÊU CẦU CỦA ỨNG DỤNG

## ❖ Thời gian

- ❖ Đảm bảo thời gian: ví dụ mỗi bit gửi từ socket gửi tới socket nhận không chậm hơn 10msec
- ❖ Dịch vụ kiểu này cần cho các ứng dụng thời gian thực có tính tương tác, ví dụ Internet telephony, virtual environment, teleconferencing, và multiplayer game
- ❖ Trong trò chơi nhiều người chơi hoặc môi trường tương tác ảo, độ trễ lớn giữa hành động và thể hiện được hành động đó trong môi trường (ví dụ, người chơi khác quan sát thấy) sẽ làm giảm tính thực tế của ứng dụng

# CÁC YÊU CẦU CỦA ỨNG DỤNG

- ❖ An toàn bảo mật
  - ❖ Giao thức tầng giao vận có thể cung cấp một hoặc nhiều dịch vụ an toàn bảo mật
  - ❖ Ví dụ, tại nút gửi, giao thức tầng giao vận có thể mã hóa dữ liệu mà tiến trình gửi truyền đi, và tại nút nhận, giao thức tầng giao vận có thể giải mã dữ liệu trước khi chuyển tới tiến trình nhận
  - ❖ Những dịch vụ như vậy có thể cung cấp tính bí mật giữa hai tiến trình, thậm chí nếu dữ liệu bị quan sát khi truyền từ tiến trình gửi tới tiến trình nhận
  - ❖ Giao thức tầng giao vận có thể cung cấp các dịch vụ an toàn bảo mật khác ngoài tính bí mật, ví dụ như tính toàn vẹn dữ liệu, tính xác thực

# YÊU CẦU DỊCH VỤ CỦA MỘT SỐ ỨNG DỤNG

Application	Data Loss	Throughput	Time-Sensitive
File transfer/download	No loss	Elastic	No
E-mail	No loss	Elastic	No
Web documents	No loss	Elastic (few kbps)	No
Internet telephony/ Video conferencing	Loss-tolerant	Audio: few kbps–1 Mbps Video: 10 kbps–5 Mbps	Yes: 100s of msec
Streaming stored audio/video	Loss-tolerant	Same as above	Yes: few seconds
Interactive games	Loss-tolerant	Few kbps–10 kbps	Yes: 100s of msec
Instant messaging	No loss	Elastic	Yes and no



# DỊCH VỤ TẦNG GIAO VẬN CỦA INTERNET

- ❖ Dịch vụ TCP: Mô hình dịch vụ TCP bao gồm dịch vụ hướng kết nối và dịch vụ truyền dữ liệu tin cậy
- ❖ Dịch vụ UDP: UDP là không hướng kết nối. UDP cung cấp dịch vụ truyền dữ liệu không tin cậy.
- ❖ Các dịch vụ không được cung cấp bởi các giao thức tầng giao vận của Internet
  - An toàn bảo mật: TCP cung cấp truyền dữ liệu tin cậy giữa hai nút và có thể dễ dàng bổ sung SSL tại tầng ứng dụng để cung cấp dịch vụ an toàn bảo mật
  - Đảm bảo thông lượng và thời gian là dịch vụ không được cung cấp.
  - Internet ngày nay có thể cung cấp các dịch vụ chấp nhận được cho các ứng dụng bị ảnh hưởng bởi thời gian nhưng không thể đảm bảo về thông lượng hoặc thời gian

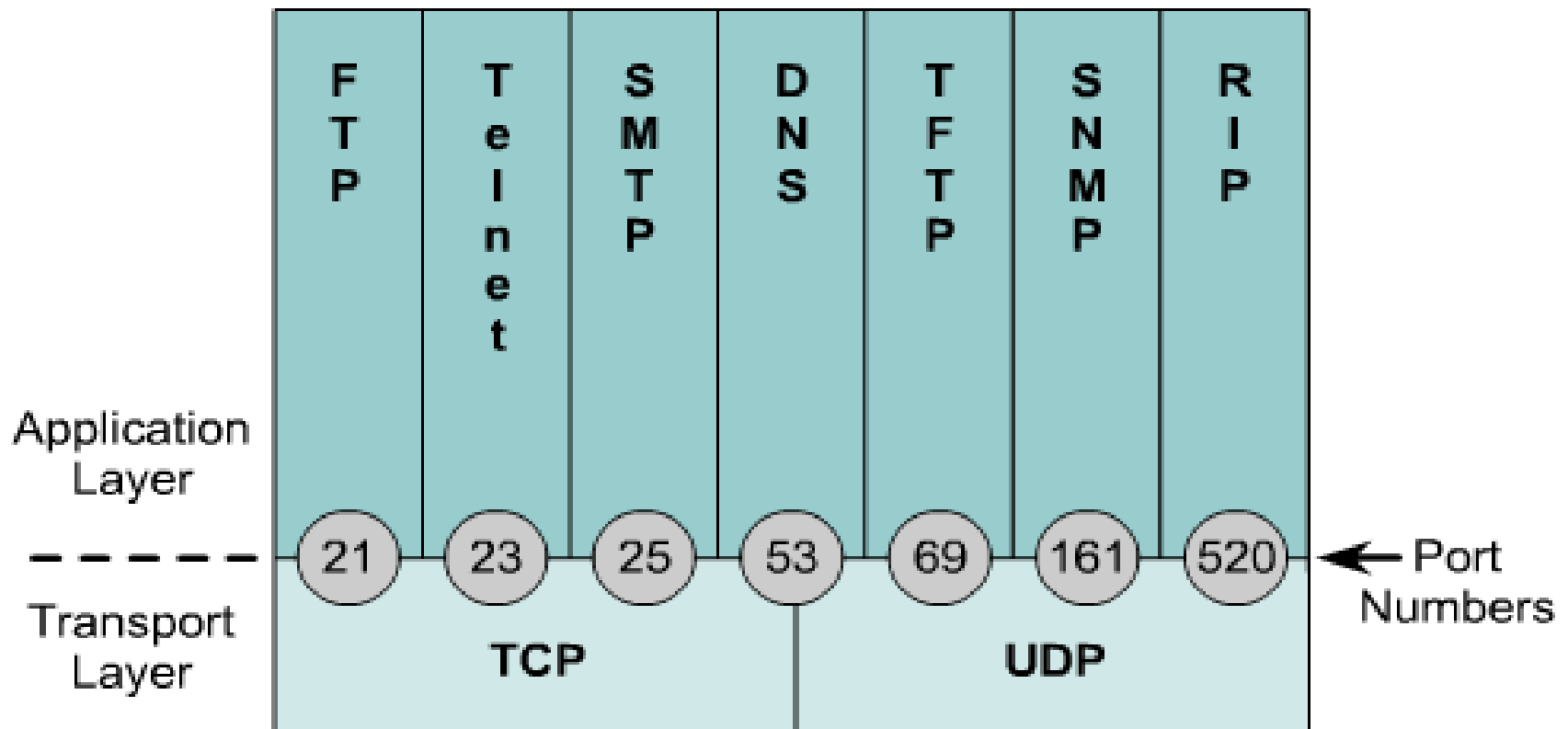
# DỊCH VỤ TẦNG GIAO VẬN CỦA INTERNET

Application	Application-Layer Protocol	Underlying Transport Protocol
Electronic mail	SMTP [RFC 5321]	TCP
Remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
File transfer	FTP [RFC 959]	TCP
Streaming multimedia	HTTP (e.g., YouTube)	TCP
Internet telephony	SIP [RFC 3261], RTP [RFC 3550], or proprietary (e.g., Skype)	UDP or TCP

# GIAO THỨC TẦNG ỨNG DỤNG

- ❖ Giao thức tầng ứng dụng định nghĩa các quy tắc giao tiếp
  - ❖ Kiểu của bản tin, ví dụ bản tin yêu cầu (request message), bản tin trả lời (response message)
  - ❖ Cú pháp của các kiểu bản tin, ví dụ các trường thông tin trong bản tin và cách thức phân chia các trường
  - ❖ Ngữ nghĩa của các trường: thông tin trong trường này dùng để làm gì
  - ❖ Nguyên tắc để xác định một tiến trình gửi và trả lời gói tin khi nào và như thế nào.
- ❖ Một số giao thức tầng ứng dụng được đặc tả trong các RFC:
  - ❖ Giao thức tầng ứng dụng Web: HTTP (HyperText Transfer Protocol [RFC 2616])
- ❖ Nhiều giao thức tầng ứng dụng là có bản quyền sở hữu
  - ❖ Giao thức tầng ứng dụng của Skype

# CÁC CÔNG PHỐ BIẾN DÙNG CHO CÁC GIAO THỨC LỚP ỨNG DỤNG



# CHƯƠNG 1: TỔNG QUAN VỀ MẠNG MÁY TÍNH

1

- Các giao thức tầng ứng dụng

2

- Dịch vụ Web và giao thức HTTP

3

- Giao thức truyền tệp FTP

4

- Giao thức thư điện tử E-mail

5

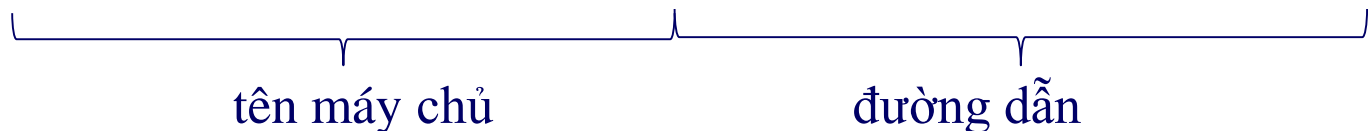
- Giao thức DNS

# DỊCH VỤ WEB VÀ GIAO THỨC HTTP

- ❖ WWW (World Wide Web): Berners-Lee 1994
  - ❖ Trao đổi dữ liệu siêu văn bản HTML (HyperText Markup Language) trên Internet.
- ❖ Đặc điểm mới thu hút người sử dụng là Web hoạt động theo yêu cầu (*on demand*)
- ❖ Mọi người có thể trở thành người xuất bản với chi phí rất thấp

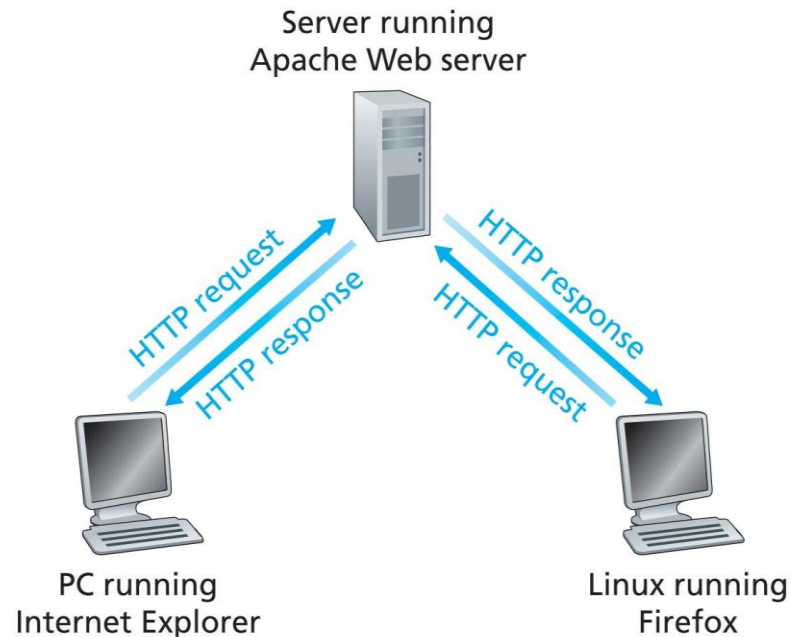
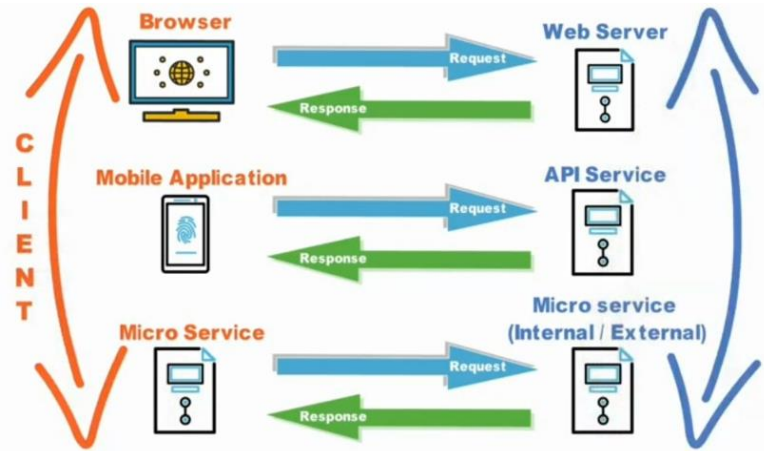
# MỘT SỐ THUẬT NGỮ VỀ WEB

- ❖ **Trang Web (Webpage – hay còn gọi là một tập tin) chứa các đối tượng.**
  - ❖ Đối tượng là một tệp - chẳng hạn như tệp HTML, hình ảnh JPEG, hoặc một video clip, ....
  - ❖ Mỗi đối tượng được xác định qua địa chỉ URL duy nhất
- ❖ **URL (Uniform Resource Locator, nghĩa tiếng Việt: định vị tài nguyên thống nhất)**
  - ❖ Được gọi một cách thông thường là một địa chỉ web)
  - ❖ Là một tham chiếu đến tài nguyên web chỉ định vị trí của nó trên một mạng máy tính để truy xuất nó.
  - ❖ Mỗi URL có hai thành phần: tên của máy chủ (hostname) chứa đối tượng và tên đường dẫn của đối tượng.
  - ❖ Ví dụ: URL <http://www.someSchool.edu/someDepartment/picture.gif>



# GIỚI THIỆU VỀ HTTP

- ❖ HTTP (HyperText Transfer Protocol):  
Giao thức tầng ứng dụng của Web
  - ❖ RFC 1945 và RFC 2616
  - ❖ RFC 7540 (May 2015)
- ❖ HTTP sử dụng TCP làm giao thức tầng giao vận
- ❖ Mô hình Client-Server
  - ❖ Client: Yêu cầu truy nhập tới các trang web (đối tượng web)
  - ❖ Server: Nhận yêu cầu và trả lời cho client
- ❖ HTTP là giao thức không trạng thái (**stateless protocol**)





# HTTP HOẠT ĐỘNG NHƯ THẾ NÀO?

- ❖ Server mở một TCP socket chờ tại cổng 80
- ❖ Client khởi tạo một kết nối TCP tới server
- ❖ Server chấp nhận yêu cầu tạo kết nối
- ❖ Trao đổi thông điệp HTTP
  - ❖ HTTP request
  - ❖ HTTP Response
- ❖ Đóng kết nối TCP

# KẾT NỐI NON-PERSISTENT VÀ PERSISTENT

## ❖ *Kiểu không giữ kết nối (non-persistent HTTP)*

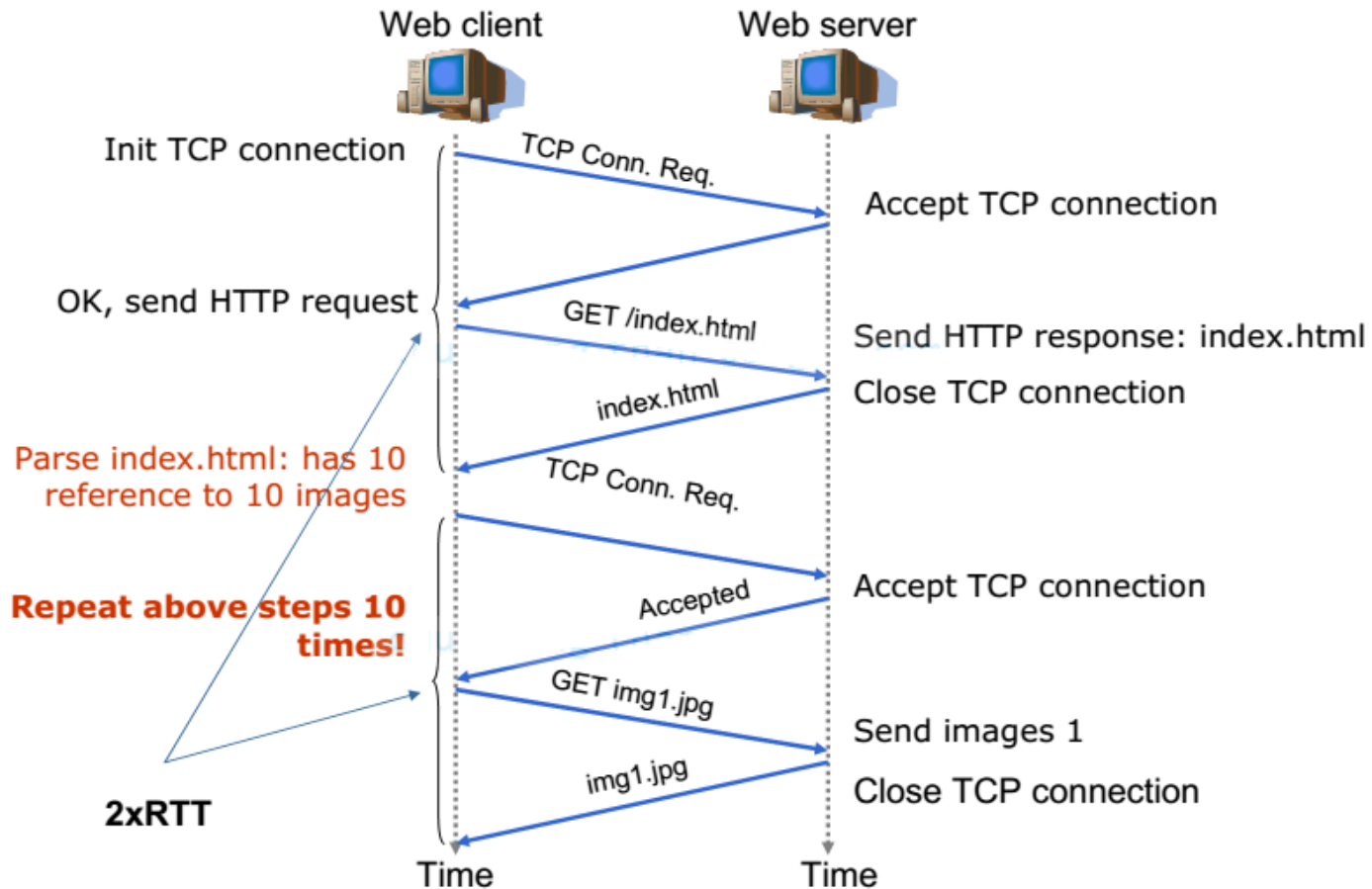
- Một đối tượng (object) gửi qua kết nối TCP
  - ✓ Sau đó kết nối TCP đóng
- Tải nhiều object cần nhiều kết nối

## ❖ *Kiểu giữ kết nối (persistent HTTP)*

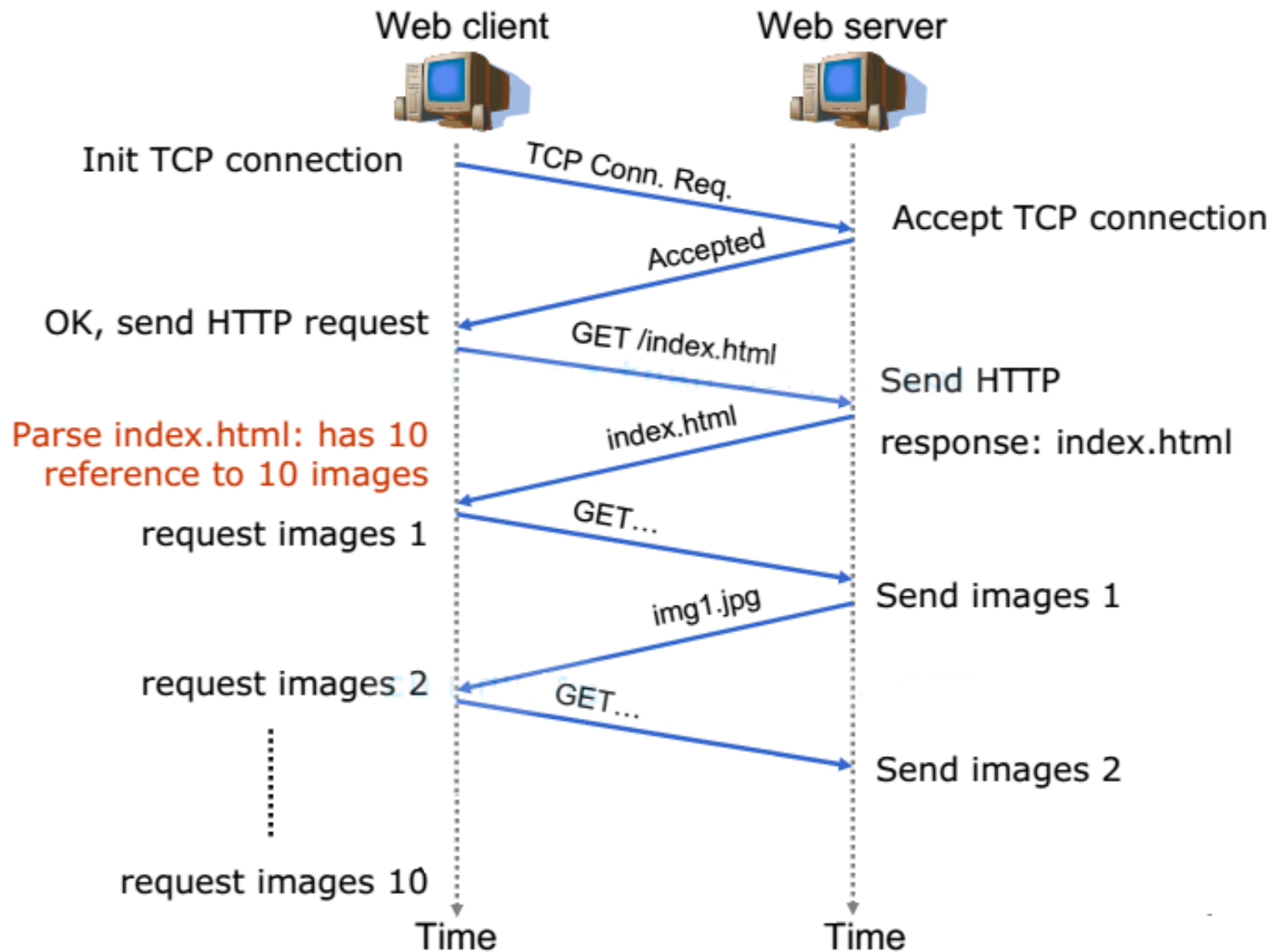
- Nhiều objects có thể gửi qua một kết nối TCP giữa client và server

# KẾT NỐI NON-PERSISTENT

www.someSchool.edu/someDepartment/index.html  
(chứa văn bản, và tham chiếu tới 10 ảnh jpg)



# KẾT NỐI PERSISTENT



# CÁC THÀNH PHẦN HTTP REQUEST

- ❖ Request URL (Endpoint) – Xác định điểm đích mà http request gửi đi:

Định dạng `http/ https ://domain/path ? Quesery String Key = Value`

- ❖ Request Method: Các phương thức hay sử dụng
  - GET: Truy suất dữ liệu từ Server ở tài nguyên đã chỉ định trước
  - HEAD: Trả về nội dung header của tài nguyên cụ thể, ko quan tâm body.
  - POST: Gửi dữ liệu lên Server xử lý tài nguyên nào đó.
  - PUT: Tạo tài nguyên mới hay ghi đè lên tài nguyên cũ.
  - Delete: Xóa tài nguyên nhất định ở Server.

# CÁC THÀNH PHẦN HTTP REQUEST

- ❖ Request Headers: Danh sách các key và value, mỗi cặp key và value mang thông tin gói tin hiện tại.
  - User-agent: Đưa thông tin về thiết bị gửi gói tin (tên của browser, mobile apps ...)
  - Accept: Kiểu định dạng mong muốn có trong http response, có thể là text, javascript, json ...
  - Accept-Encoding: Kiểu mã hóa mà http request mong muốn nhận được trong http response.
  - Content-type: Kiểu định dạng được sử dụng trong body.
  - Content-length: Độ dài body tính bằng byte.
  - Connection: Yêu cầu server có duy trì kết nối không.
  - Cache-Control: Cho phép lưu trữ tạm thông tin để tái sử dụng
- ❖ Request Body: Thân chứa nội dung thông tin mà client gửi lên server xử lý.

# KHUÔN DẠNG HTTP REQUEST MESSAGE

○ ASCII

Request line  
(GET, POST, HEAD  
commands)

header  
Lines

GET /somedir/page.html HTTP/1.1 \r\n

Host: [www.someschool.edu](http://www.someschool.edu) \r\n

Connection: close \r\n

User-agent: Mozilla/5.0 \r\n Accept-

language: Fr \r\n

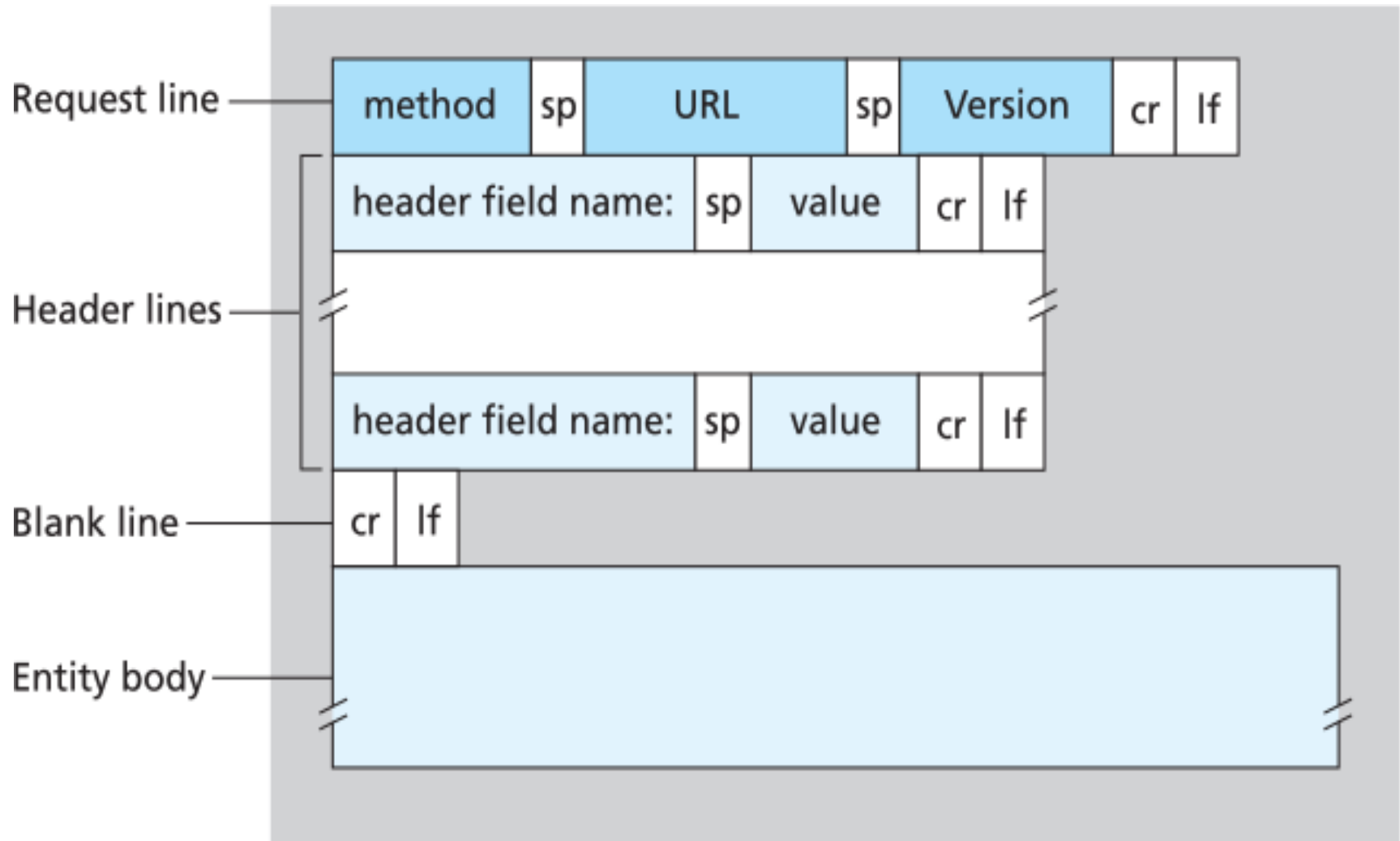
carriage return, —> \r\n

line feed ở đầu dòng chỉ ra kết thúc phần header

carriage return character  
line-feed character

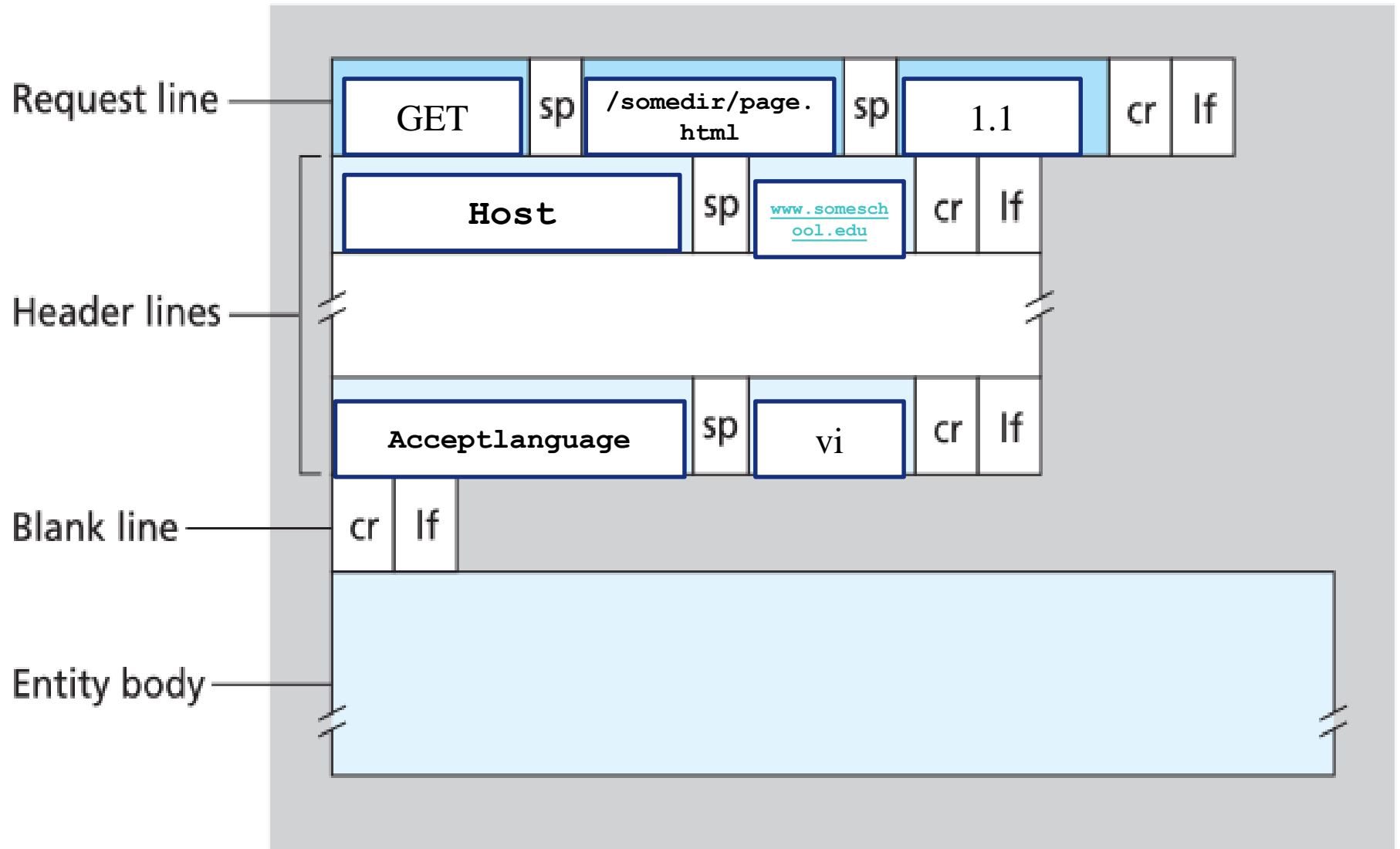
Địa chỉ URL của văn bản mà trình duyệt yêu cầu?

# KHUÔN DẠNG HTTP REQUEST MESSAGE





# KHUÔN DẠNG HTTP REQUEST MESSAGE



# KHUÔN DẠNG HTTP RESPONSE MESSAGE

status line  
(protocol

HTTP/1.1 200 OK \r\n

status code  
status  
phrase)

Connection: close \r\n

Date: Tue, 09 Aug 2011 15:44:04 GMT \r\n

Server: Apache/2.2.3 (CentOS) \r\n

Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT \r\n

Content-Length: 6821 \r\n Content-Type: text/html \r\n

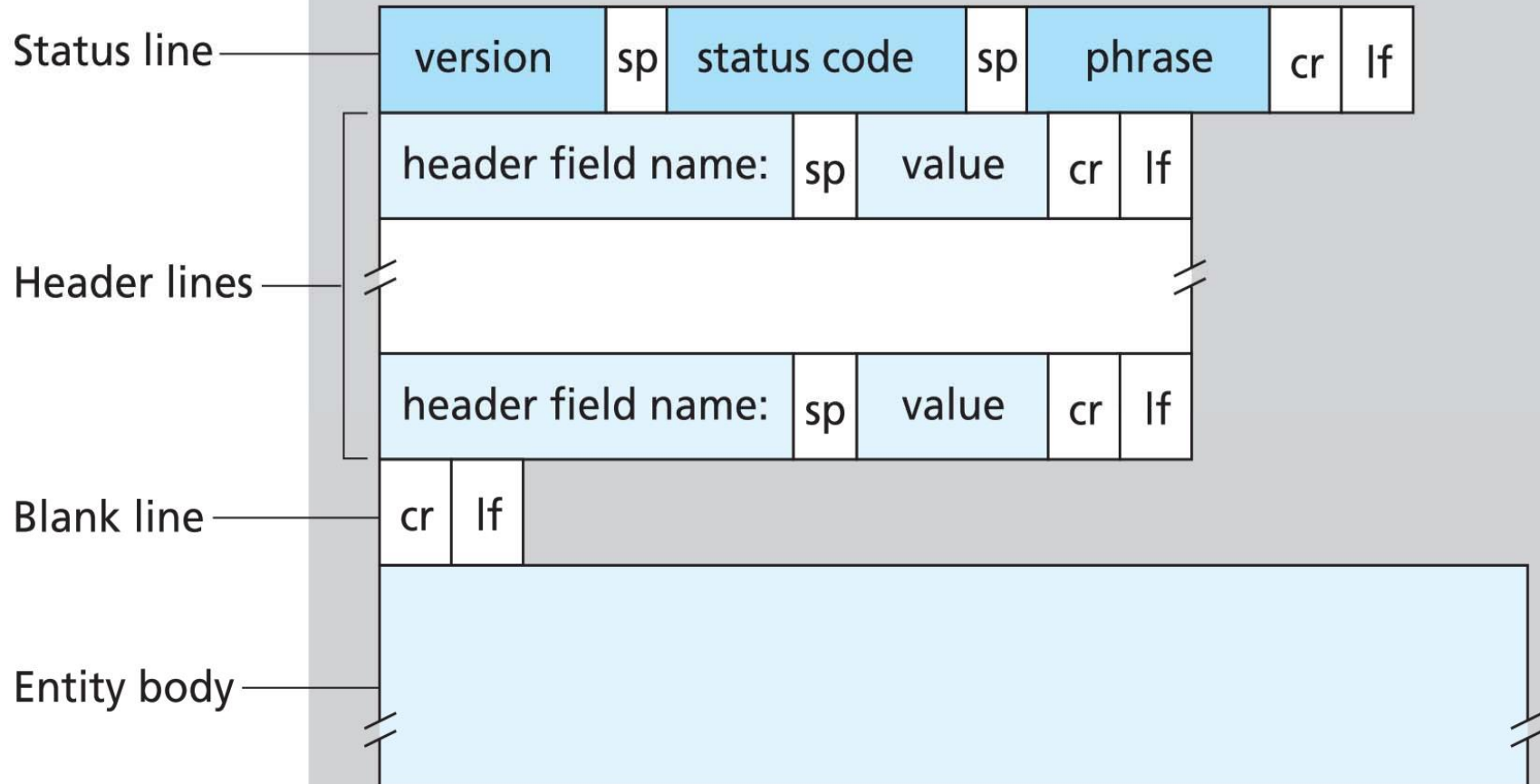
\r\n

header  
lines

data data data data data ...

data, e.g.,  
requested  
HTML file

# KHUÔN DẠNG HTTP RESPONSE MESSAGE



# MỘT SỐ MÃ TRẠNG THÁI PHỔ BIẾN

- ❖ 200 OK: Yêu cầu thành công và thông tin được trả về trong phản hồi.
- ❖ 301 Moved Permanently: Đối tượng được yêu cầu đã bị di chuyển vĩnh viễn; URL mới được chỉ định trong trường Location (vị trí) của thông điệp phản hồi. Phần mềm máy khách sẽ tự động truy xuất URL mới.
- ❖ 400 Bad Request: Server không hiểu được yêu cầu từ client.
- ❖ 404 Not Found: Tài liệu được yêu cầu không tồn tại trên máy chủ
- ❖ 505 HTTP Version Not Supported: Phiên bản giao thức HTTP được yêu cầu không được máy chủ hỗ trợ.

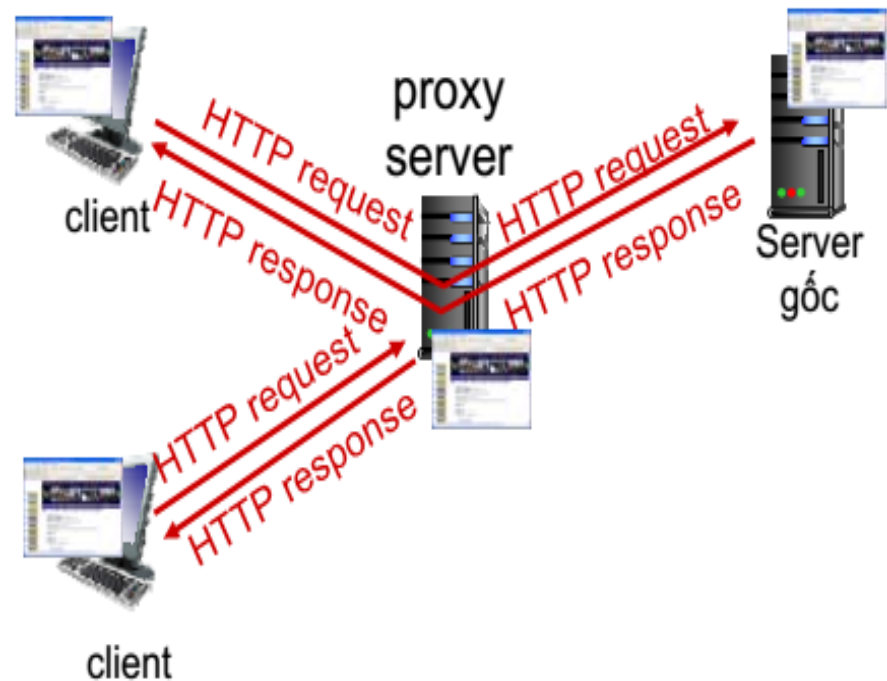
# Quan sát HTTP message:

## Tạo HTTP request dùng trình duyệt

1. Chạy phần mềm Wireshark để ghi lại dữ liệu gửi nhận  
(Chọn Capture -> Start)
2. Mở trình duyệt vào trang web:  
[www.tlu.edu.vn](http://www.tlu.edu.vn)
3. Dừng việc ghi dữ liệu của phần mềm  
Wireshark
4. Gõ Filter: `http.host == " tlu.edu.vn"`
5. Quan sát HTTP message

# BỘ ĐỆM WEB

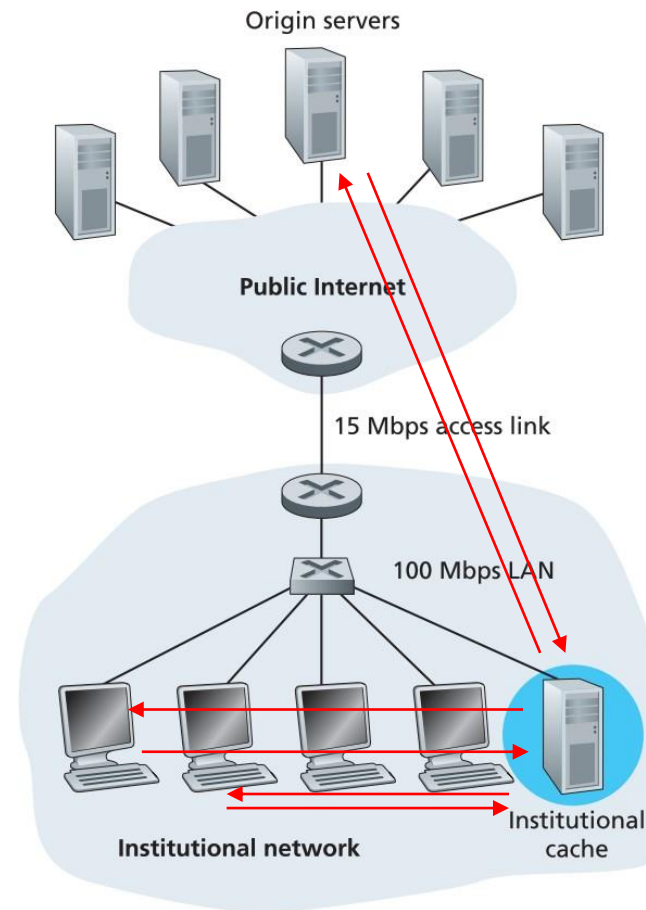
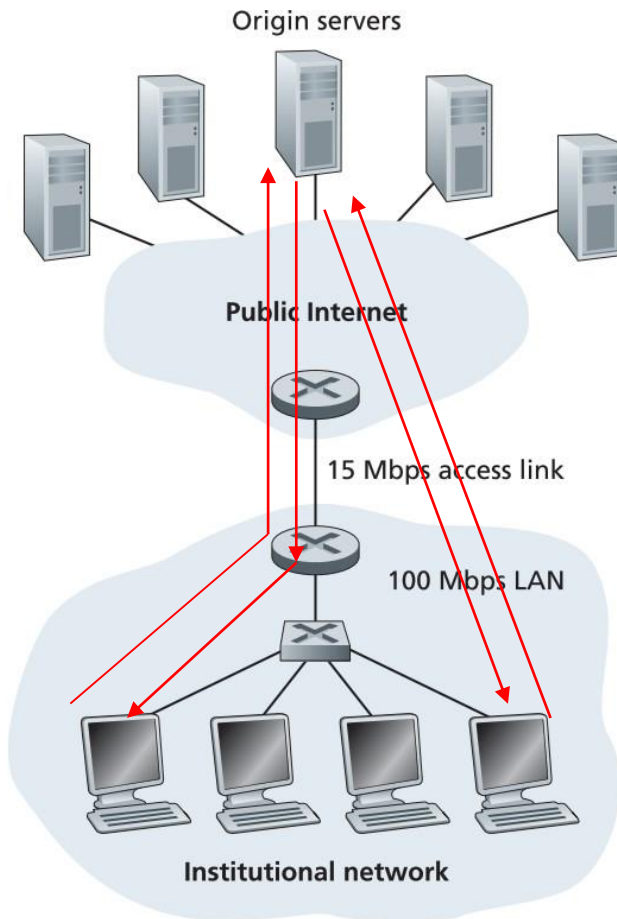
- ❖ Bộ đệm Web (gọi là proxy server): có ổ lưu trữ riêng và giữ các bản sao của các đối tượng được yêu cầu gần đây.



- ❖ Trình duyệt gửi yêu cầu đối tượng web đến proxy server
  - ❖ Hit: Proxy trả đối tượng web cho trình duyệt
  - ❖ Miss: Proxy gửi yêu cầu tới máy chủ web, trả lời trình duyệt và lưu đối tượng web

# BỘ ĐỆM WEB

- ❖ Giảm lưu lượng Web trên đường truy cập ra Internet
- ❖ Giảm thời gian đáp ứng cho yêu cầu của client



# CHƯƠNG 1: TỔNG QUAN VỀ MẠNG MÁY TÍNH

1

- Các giao thức tầng ứng dụng

2

- Dịch vụ Web và giao thức HTTP

3

- Giao thức truyền tệp FTP

4

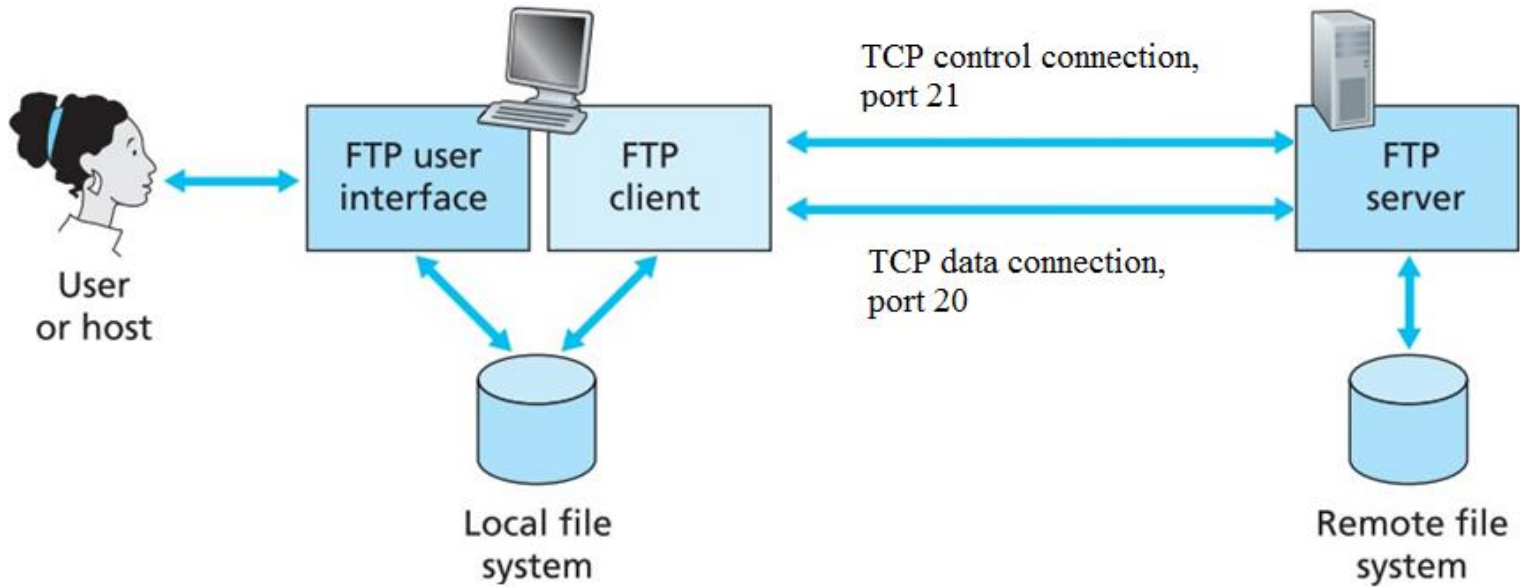
- Giao thức thư điện tử E-mail

5

- Giao thức DNS



# TRUYỀN TẬP TIN: FTP



- ❖ Truyền file đến/ từ máy ở xa
- ❖ RFC 959
- ❖ Mô hình client/server
- ❖ Kết nối điều khiển và kết nối dữ liệu riêng biệt
- ❖ FTP server duy trì trạng thái: thư mục hiện tại, xác thực trước đó
  - ❖ HTTP không duy trì trạng thái

# TRUYỀN TẬP TIN: FTP



- ❖ Kết nối điều khiển (control connection): kiểm soát thông tin điều khiển, duy trì trong suốt phiên làm việc
  - ❖ HTTP connection: dùng chung
- ❖ Kết nối dữ liệu (data connection): Truyền dữ liệu và tự động ngắt ngay sau khi quá trình truyền dữ liệu kết thúc.
  - ❖ HTTP không duy trì trạng thái

# PHƯƠNG THỨC TRUYỀN TẢI DỮ LIỆU FTP

- ❖ Stream Mode: Dữ liệu truyền đi dưới dạng các byte không liên tiếp.
- ❖ Block Mode: Chia dữ liệu thành các khối nhỏ và đóng gói thành các FTP Block và truyền đi.
- ❖ Compressed Mode: Sử dụng phương thức run-length encoding mã hóa để loại bỏ dữ liệu trùng lặp nhằm giảm dung lượng dữ liệu gửi đi.

# CHƯƠNG 1: TỔNG QUAN VỀ MẠNG MÁY TÍNH

1

- Các giao thức tầng ứng dụng

2

- Dịch vụ Web và giao thức HTTP

3

- Giao thức truyền tệp FTP

4

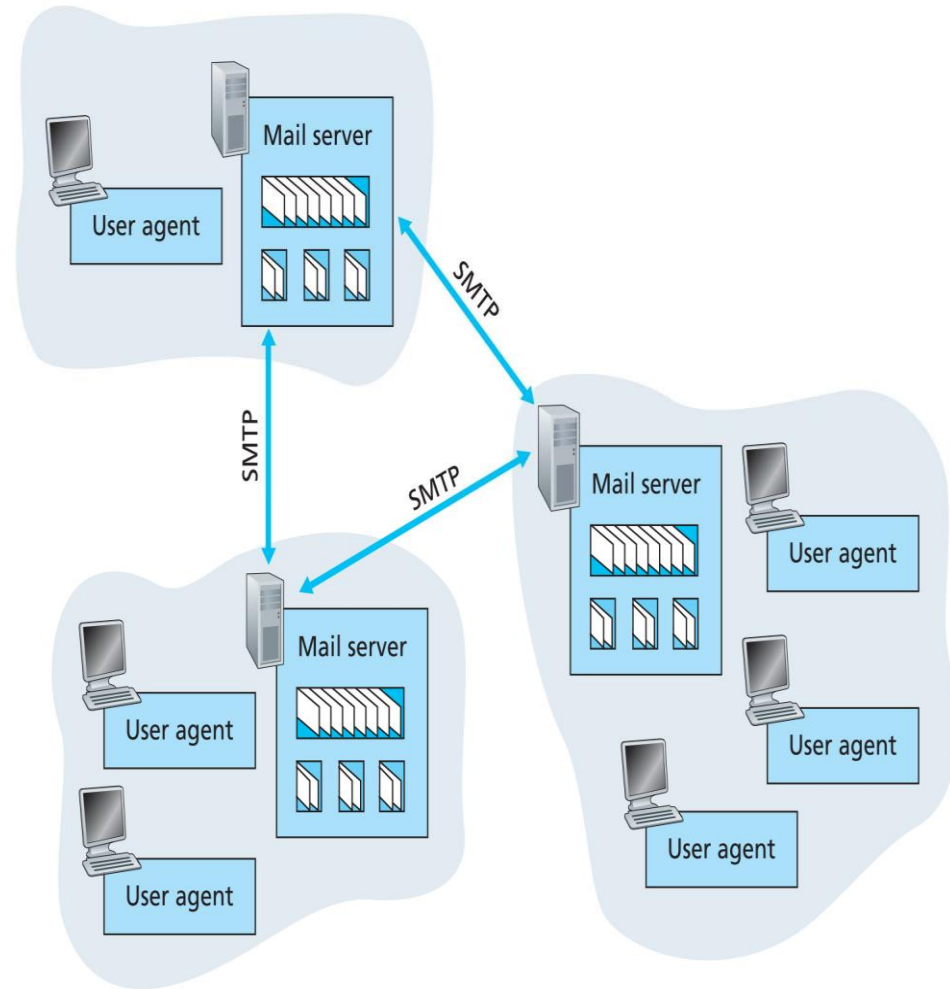
- Giao thức thư điện tử E-mail

5

- Giao thức DNS

# THƯ ĐIỆN TỬ

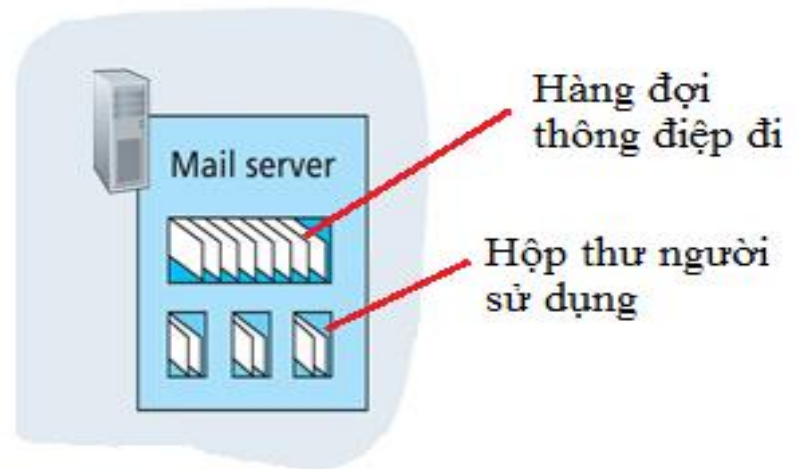
- ❖ Ba thành phần chính
  - ❖ User agent
  - ❖ Mail server
  - ❖ SMTP (simple mail transfer protocol)
- ❖ User agent:
  - ❖ Chương trình giao tiếp người dùng, gọi là mail reader – trình đọc thư
  - ❖ Soạn thảo, gửi, đọc và trả lời thư



# THƯ ĐIỆN TỬ

## ❖ Mail server

- Hộp thư (mail box) chứa thông điệp đến user
- Hàng đợi chứa các thông điệp chuẩn bị gửi



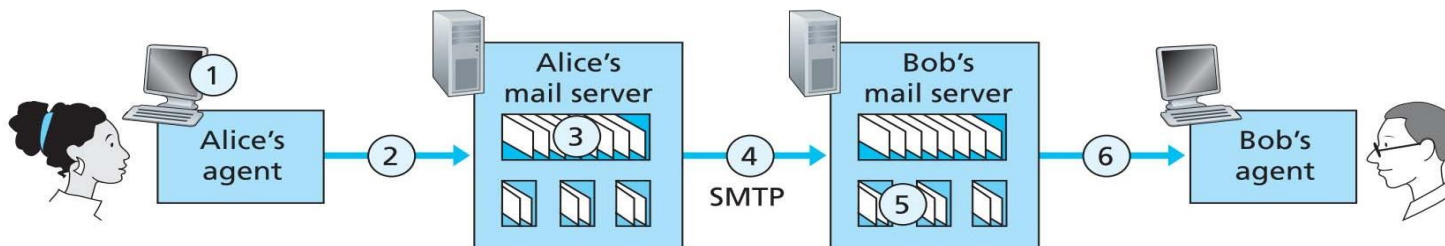
- ## ❖ Giao thức SMTP giữa các mail server để gửi các thông điệp email
- Client: Mail server gửi
  - Server: Mail server nhận

# GIAO THỨC SMTP

- ❖ RFC 2821
- ❖ Sử dụng TCP để truyền tin cậy bản tin email từ client tới cổng 25 server
- ❖ Truyền trực tiếp: server gửi tới server nhận
- ❖ 3 giai đoạn truyền: bắt tay, truyền thông điệp, đóng
- ❖ giao tiếp kiểu lệnh/ phản hồi (giống HTTP, FTP)
  - ❖ Lệnh: văn bản ASCII
  - ❖ Phản hồi: mã trạng thái (status code) và cụm từ (phrase)
- ❖ Thông điệp phải ở dạng mã ASCII 7 bit

# TÌNH HUỐNG: ALICE GỬI THÔNG ĐIỆN ĐẾN BOB

- 1) Alice dùng một UA để soạn thảo thông điệp “gửi đến” [bob@some school.edu](mailto:bob@some school.edu)
- 2) UA của Alice gửi thông điệp đến mail server của cô ta; thông điệp được đặt trong hàng đợi
- 3) Phần client của SMTP server mở kết nối TCP với mail server của Bob
- 4) Phần client của SMTP server gửi thông điệp của Alice trên kết nối TCP
- 5) Mail server của Bob đặt thông điệp đó trong hộp thư của Bob
- 6) Bob kích hoạt user agent của anh ta để đọc thông điệp





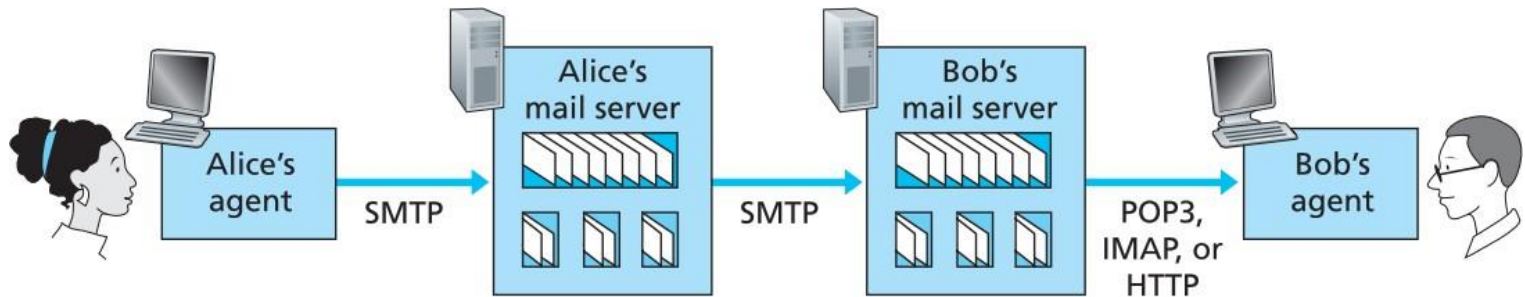
# SMTP

- ❖ SMTP sử dụng persistent connection
- ❖ SMTP yêu cầu bản tin (header & body) là 7-bit ASCII
- ❖ SMTP server sử dụng CRLF.CRLF để xác định kết thúc bản tin

So sánh với HTTP:

- ❖ Cùng giao tiếp dùng lệnh/phản hồi, mã trạng thái dạng ASCII
- ❖ HTTP: mỗi object chứa trong bản tin riêng
- ❖ SMTP: nhiều object được gửi trong một bản tin có nhiều phần (multipart message)

# GIAO THỨC TRUY CẬP THƯ ĐIỆN TỬ



- ❑ **SMTP**: chuyển email tới server của người nhận
- ❑ Giao thức truy cập thư điện tử (mail access protocol): lấy email từ server
  - **POP** (Post Office Protocol [RFC 1939]): đăng nhập và lấy thư về
  - **IMAP** (Internet Mail Access Protocol [RFC 1730]): có nhiều đặc tính hơn, ví dụ thao tác với email trên server
  - **HTTP**: gmail, Hotmail, Yahoo Mail,...

# CHƯƠNG 1: TỔNG QUAN VỀ MẠNG MÁY TÍNH

1

- Các giao thức tầng ứng dụng

2

- Dịch vụ Web và giao thức HTTP

3

- Giao thức truyền tệp FTP

4

- Giao thức thư điện tử E-mail

5

- Giao thức DNS

# HỆ THỐNG TÊN MIỀN: DNS

- ❖ Con người: có nhiều cách để nhận dạng như: Tên, số CMND
- ❖ Host, router trên Internet
  - Địa chỉ IP: ví dụ 203.113.135.55
  - Tên: ví dụ [www.tlu.edu.vn](http://www.tlu.edu.vn)
- ❖ Ánh xạ giữa địa chỉ IP và tên?
- ❖ DNS (domain name system)
  - Cơ sở dữ liệu phân tán được cài đặt trên một hệ thống phân cấp các máy phục vụ tên (name server)
  - Là giao thức tầng ứng dụng cho phép máyính, máy chủ sử dụng để ánh xạ địa chỉ IP-tên

# HỆ THỐNG TÊN MIỀN: DNS

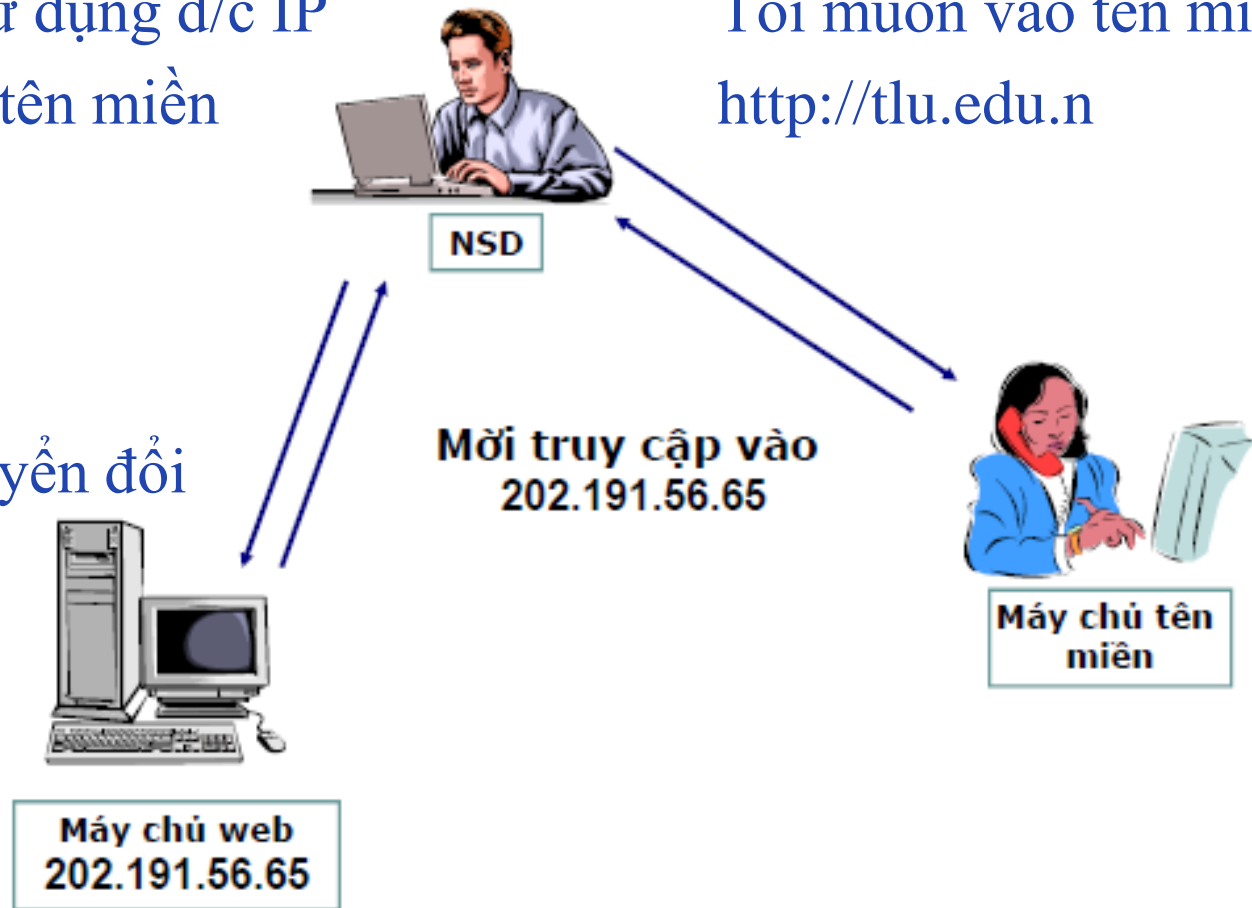
- ❖ Tên miền: Định danh trên tầng ứng dụng cho các nút mạng
  - Trên Internet được quản lý tập trung
  - Quốc tế: ICANN
  - Việt Nam: VNNIC
- ❖ DNS(Domain Name System): hệ thống tên miền
  - Không gian thông tin tên miền
  - Gồm các máy chủ quản lý thông tin tên miền và cung cấp dịch vụ DNS
- ❖ Vấn đề phân giải tên miền sang địa chỉ IP
  - Người sử dụng dùng tên miền để truy cập dịch vụ
  - Máy tính và các thiết bị mạng không sử dụng tên miền mà dùng địa chỉ IP khi trao đổi dữ liệu
- ❖ Làm thế nào để chuyển đổi tên miền sang địa chỉ IP?

# VÍ DỤ CHUYỂN ĐỔI ĐỊA CHỈ

Máy tính sử dụng đ/c IP  
NSD dùng tên miền

Tôi muốn vào tên miền  
<http://tlu.edu.n>

Cần có chuyển đổi

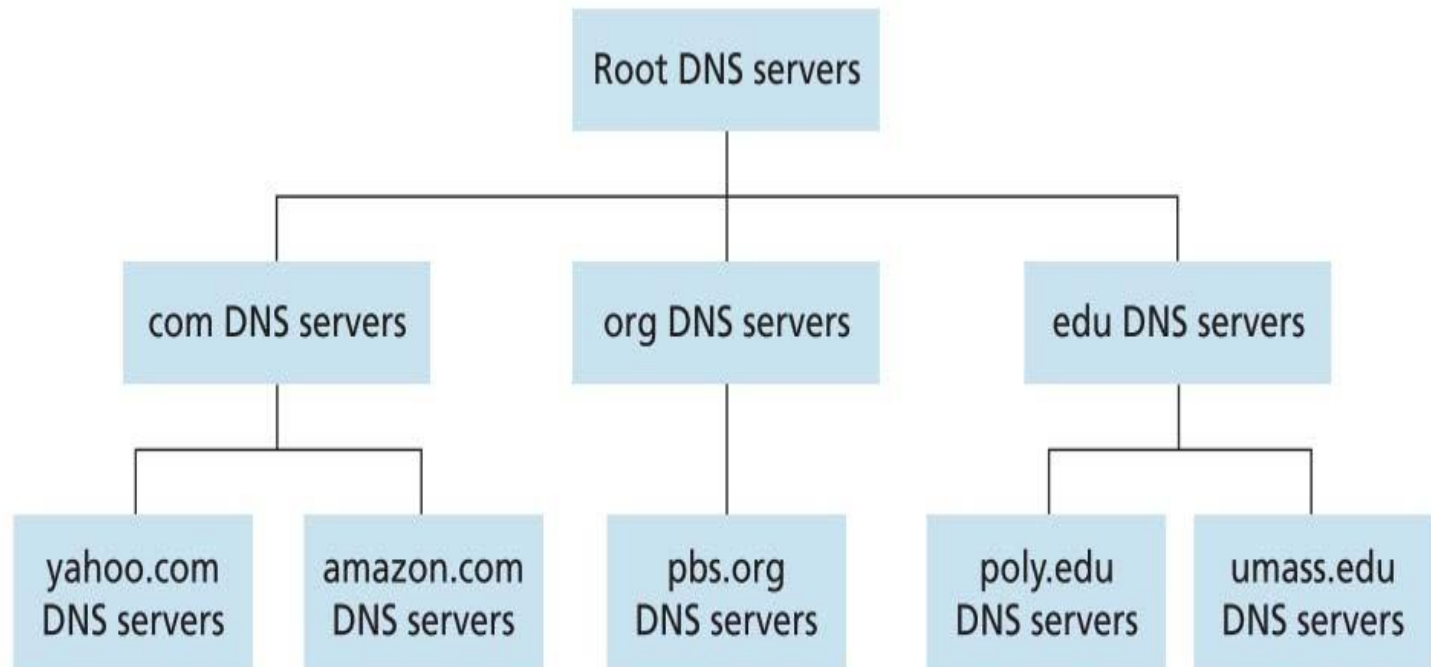


# QUY TẮC ĐẶT TÊN MIỀN

- ❖ Quy tắc đặt tên miền:
- ❖ Độ dài tối đa : 255 ký tự
- ❖ Độ dài tối đa của label : 63 ký tự
- ❖ Label phải bắt đầu bằng số hoặc chữ, chỉ chứa số, chữ, “-“, “.”
- ❖ Phân cấp tên miền : gốc, cấp 1, cấp 2...

# HỆ THỐNG TÊN MIỀN: DNS

- ❖ Tên miền được tổ chức theo cấu trúc phân cấp hình cây
  - Root: Tên miền cấp cao nhất gọi là tên miền gốc (Root name)
  - Zone: Dưới tên miền gốc có hai loại tên miền là tên miền cấp cao dùng chung như .com, .edu và tên miền cấp cao mã quốc gia như .vn, .jp
- ❖ DNS sử dụng nhiều name server tổ chức phân cấp và phân tán trên toàn cầu.
- ❖ Ví dụ:





# HỆ THỐNG TÊN MIỀN: DNS

- ❖ Máy chủ tên miền cấp 1 (Top Level Domain)
  - Quản lý tên miền cấp 1
- ❖ Máy chủ được ủy quyền (Authoritative DNS Servers)
  - Quản lý tên miền cấp dưới
- ❖ Máy chủ của các tổ chức: của ISP
  - Không nằm trong phân cấp của DNS
- ❖ Máy chủ cục bộ: dành cho mạng nội bộ của cơ quan tổ chức
  - Không nằm trong phân cấp của DNS

# MÁY CHỦ TÊN MIỀN GỐC

- ❖ Trả lời truy vấn cho các máy chủ cục bộ. Khi các máy chủ cục bộ không ánh xạ được địa chỉ của một máy tính thì nó sẽ truy cập đến máy chủ tên miền gốc.
- ❖ Quản lý các zone và phân quyền quản lý cho máy chủ cấp dưới
- ❖ Hiện tại trên mạng Internet có 13 hệ thống máy chủ tên miền (tham khảo danh sách tại <https://root-servers.org/> )

# MÁY CHỦ TÊN MIỀN ĐƯỢC ỦY QUYỀN

- ❖ Mỗi máy tính phải đăng ký một máy chủ tên miền có thẩm quyền (authoritative name server).
- ❖ Máy chủ tên miền có thẩm quyền là máy chủ tên miền trong miền ISP của máy tính đó.
- ❖ Máy chủ tên miền có thẩm quyền của một máy tính luôn lưu trữ bản ghi DNS cho phép xác định địa chỉ IP của máy tính đó từ tên miền.

# MÁY CHỦ TÊN MIỀN CỤC BỘ

- ❖ Mỗi ISP như trường học, công ty đều có thể có một local name server, còn được gọi là máy chủ tên miền mặc định (default name server)
- ❖ Địa chỉ IP của local name server phải được cấu hình trên máy tính
- ❖ Khi một host yêu cầu tìm ánh xạ địa chỉ, yêu cầu sẽ được gửi đến DNS cục bộ.
  - Nếu máy tính yêu cầu xác định địa chỉ IP của một máy tính khác trong cùng một ISP, thì local name server xác định được ngay địa chỉ IP cần thiết

# PHÂN GIẢI TÊN MIỀN

- ❖ Tự phân giải
  - File HOST: C:\WINDOWS\system32\drivers\etc\
  - Cache
- ❖ Dịch vụ phân giải tên miền DNS: client/server
- ❖ UDP, Port 53
- ❖ Phân giải đệ quy (Recursive Query)
- ❖ Phân giải tương tác (Interactive Query)

# THÔNG ĐIỆP DNS

- ❖ DNS Query và DNS Reply
  - Chung khuôn dạng
- ❖ QUESTION: tên miền cần truy vấn
  - Số lượng: #Question
- ❖ ANSWER: thông tin tên miền tìm kiếm được
  - Số lượng: #Answer RRs
- ❖ AUTHORITY: địa chỉ server trả lời truy vấn
- ❖ ADDITIONAL: thông tin phân giải tên miền

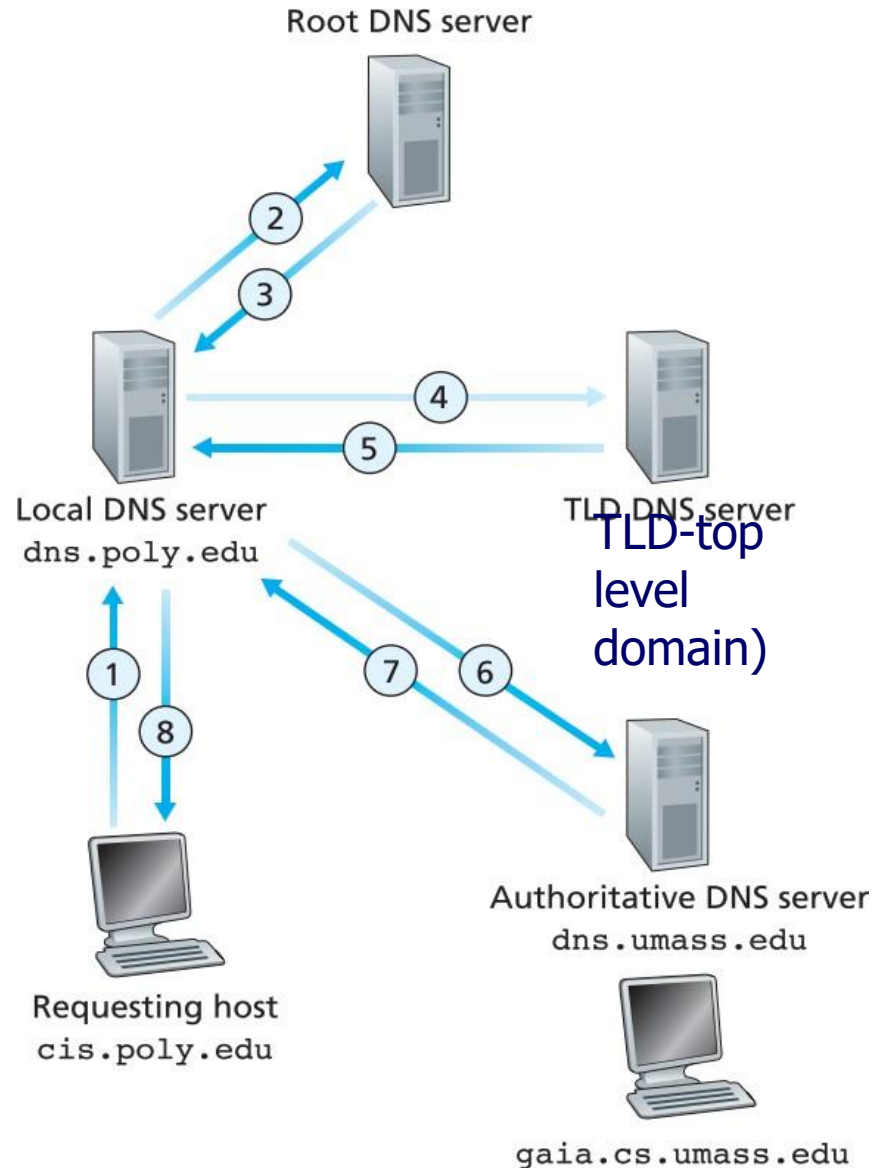
SRC = 53	DST = 53
checksum	length
Identification	Flags
#Question	#Answer RRs
#Authority RRs	#Additional RRs
QUESTION	
ANSWER	
AUTHORITY	
ADDITIONAL	

# VÍ DỤ ÁNH XẠ TÊN MIỀN DÙNG DNS

Máy trạm tại cis.poly.edu muốn địa chỉ IP của gaia.cs.umass.edu

## ❖ Phân giải tương tác

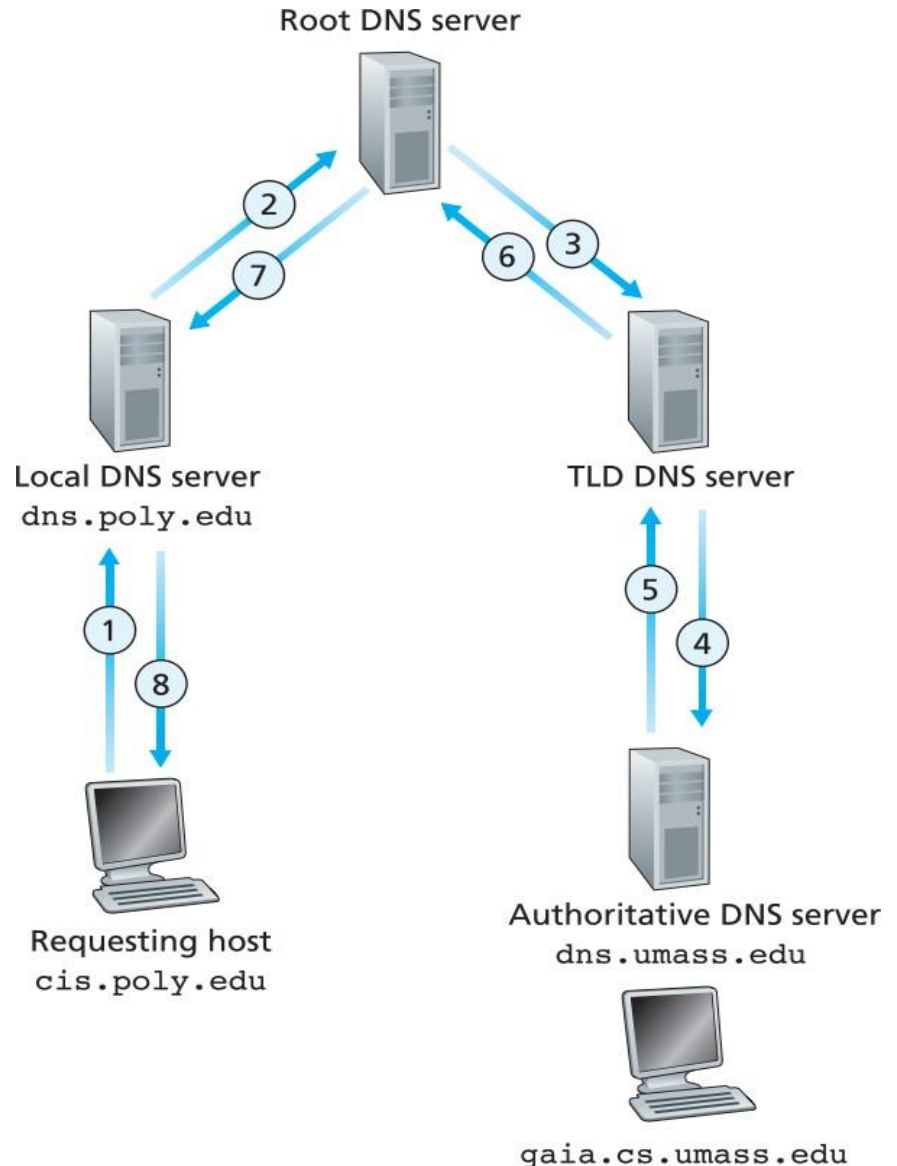
- Server được hỏi sẽ trả lời với tên của server quản lý vùng liên quan
- “Tôi không biết tên này, nhưng hãy hỏi thêm thông tin từ máy chủ này”



# VÍ DỤ ÁNH XẠ TÊN MIỀN DÙNG DNS

Phân giải đệ quy

Tải đặt lên các máy chủ cấp trên rất lớn.





# TÓM TẮT

- ❖ Kiến trúc ứng dụng
  - ❖ Client-server
  - ❖ P2P
- ❖ Yêu cầu dịch vụ ứng dụng:
  - ❖ Truyền tin cậy, độ trễ
- ❖ Mô hình dịch vụ giao vận của Internet
  - ❖ Tin cậy: TCP
  - ❖ Không tin cậy: UDP
- ❖ Giao thức:
  - ❖ HTTP
  - ❖ FTP
  - ❖ SMTP, POP, IMAP
  - ❖ DNS