

## Chương 6. Tầng ứng dụng



1

1

### 1. Tổng quan về tầng ứng dụng

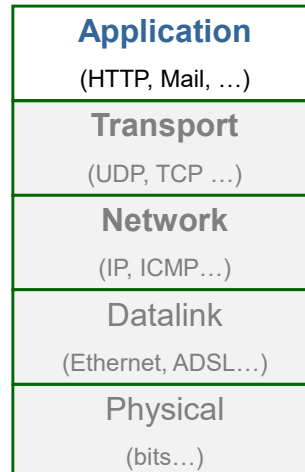


2

2

## Nhắc lại về kiến trúc phân tầng

- Cung cấp các dịch vụ trên mạng.
- Trong mô hình TCP/IP không có 2 tầng trình diễn và tầng phiên, nhưng các giao thức tầng ứng dụng phải cung cấp các chức năng của 2 tầng này (biểu diễn dữ liệu, điều khiển phiên...)

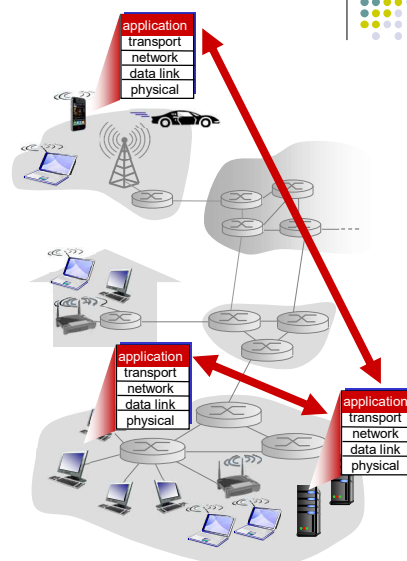


3

3

## Ứng dụng mạng

- Hoạt động trên các hệ thống đầu cuối (end system)
- Cài đặt giao thức ứng dụng để cung cấp dịch vụ
- Gồm có 2 tiến trình giao tiếp với nhau qua môi trường mạng:
  - Client: cung cấp giao diện NSD, gửi thông điệp yêu cầu dịch vụ
  - Server: cung cấp dịch vụ, trả thông điệp đáp ứng
- Ví dụ: Web
  - Web browser (trình duyệt Web): Chrome, Firefox...
  - Web server: Apache, Tomcat...

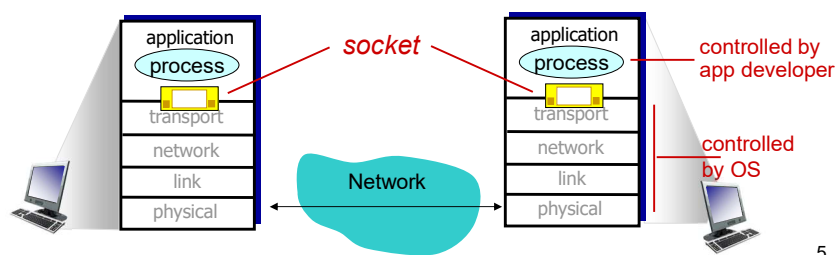


4

4

## Giao tiếp giữa các tiến trình ứng dụng

- Socket: điểm truy cập dịch vụ của tầng giao vận
  - Các tiến trình ứng dụng sử dụng socket gọi dịch vụ của tầng giao vận để trao đổi thông điệp
- Định danh cho tiến trình bởi: Địa chỉ IP, Số hiệu cổng
- Ví dụ: tiến trình web server trên máy chủ của SoICT có định danh 202.191.56.65:80

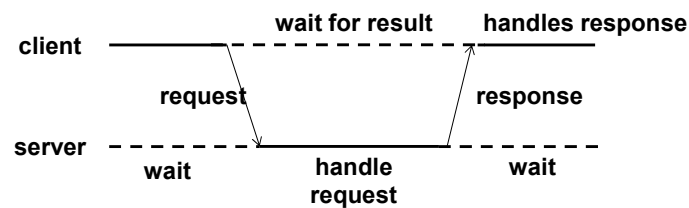


5

5

## Giao tiếp giữa các tiến trình

- Tiến trình client: gửi yêu cầu
- Tiến trình server: trả lời
- Mô hình điển hình: 1 server – nhiều client
- Client cần biết địa chỉ của server: địa chỉ IP, số hiệu cổng



6

6

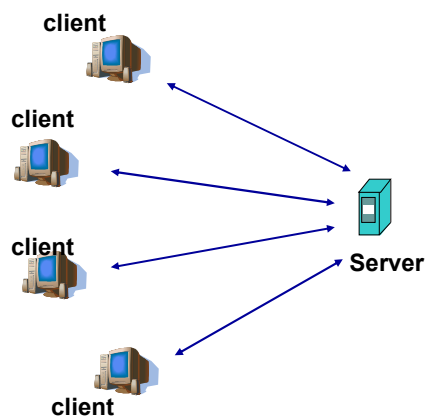
## Các mô hình ứng dụng

- Khách-chủ (Client/Server)
- Ngang hàng (P2P: Peer-to-peer)
- Mô hình lai

7

7

## Mô hình khách chủ

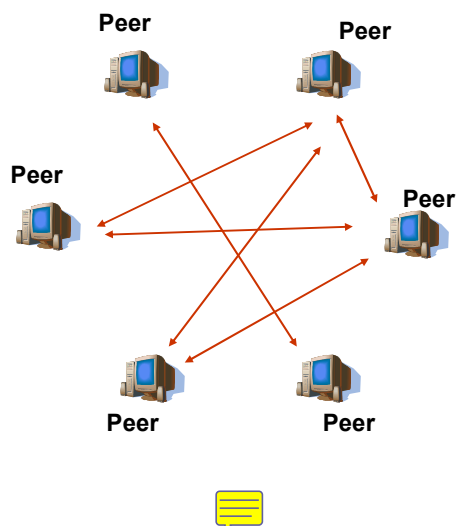


- **Khách**
  - Gửi yêu cầu truy cập dịch vụ đến máy chủ
  - Về nguyên tắc, không liên lạc trực tiếp với các máy khách khác
- **Chủ**
  - Thường xuyên online để chờ y/c đến từ máy trạm
  - Có thể có máy chủ dự phòng để nâng cao hiệu năng, phòng sự cố
- e.g. Web, Mail, ...

8

8

## Mô hình ngang hàng thuần túy

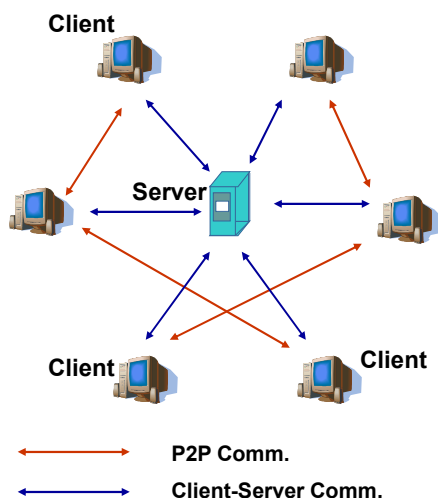


- Không có máy chủ trung tâm
- Các máy có vai trò ngang nhau
- Hai máy bất kỳ có thể liên lạc trực tiếp với nhau
- Không cần vào mạng thường xuyên
- E.g. Gnutella

9

9

## Mô hình lai



- Một máy chủ trung tâm để quản lý NSD, thông tin tìm kiếm...
- Các máy khách sẽ giao tiếp trực tiếp với nhau sau khi đăng nhập
- E.g. 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

10

10

## 2. Tên miền và dịch vụ DNS



11

11

### Giới thiệu chung



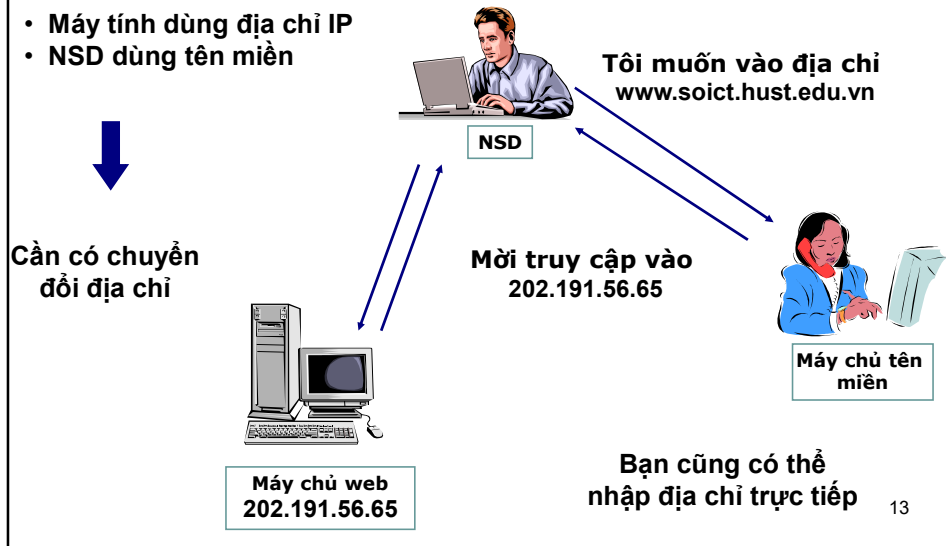
- Tên miền: định danh trên tầng ứng dụng cho các nút mạng
  - Trên Internet được quản lý tập trung
  - Quốc tế: ICANN
  - Việt Nam: VNNIC
- DNS(Domain Name System): hệ thống tên miền
  - Không gian thông tin tên miền
  - Gồm các máy chủ quản lý thông tin tên miền và cung cấp dịch vụ DNS
- Vấn đề phân 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?

12

12

## Chuyển đổi địa chỉ và ví dụ

- Máy tính dùng địa chỉ IP
- NSD dùng tên miền



13

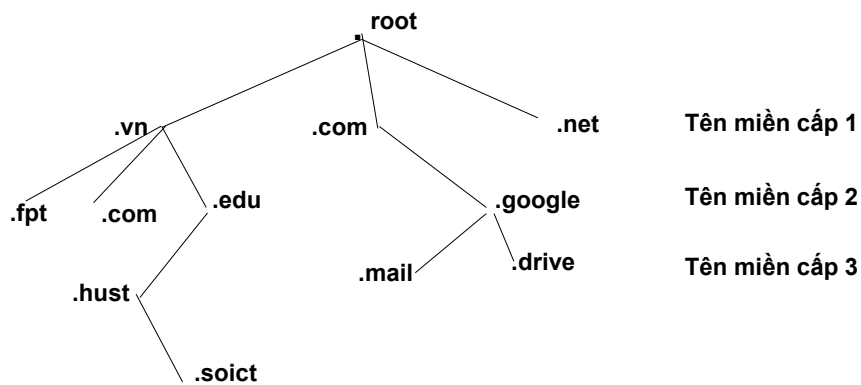
## Quy tắc đặt tên miền

- Tên miền đầy đủ(FQDN): Kết thúc bằng dấu '.' (tên miền gốc). VD: hust.edu.vn.
- Tên miền rút gọn: không thể hiện tên miền gốc. VD: hust.edu.vn
- 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...

14

14

## Phân cấp tên miền

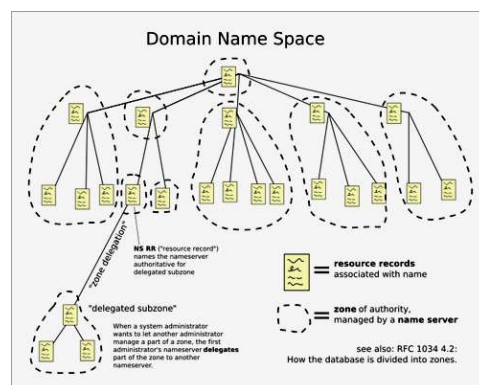


15

15

## Hệ thống DNS

- Không gian tên miền
- Kiến trúc : hình cây
  - Root
  - Zone
- Mỗi nút là một tập hợp các bản ghi (resource record) mô tả tên miền tương ứng với nút đó.
  - SOA
  - NS
  - A
  - PTR
  - CNAME...



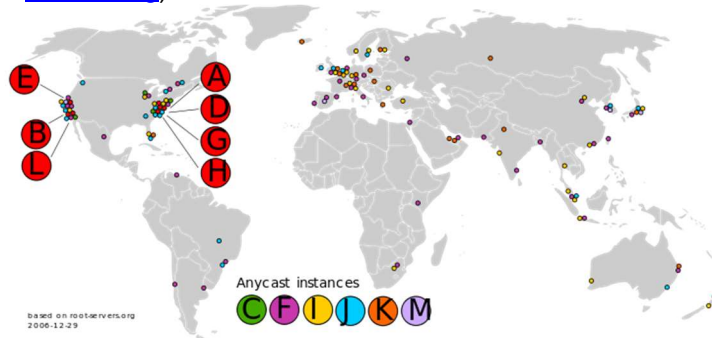
16

16



## Hệ thống máy chủ DNS

- Máy chủ tên miền gốc (Root server)
  - Trả lời truy vấn cho các máy chủ cục bộ
  - Quản lý các zone và phân quyền quản lý cho máy chủ cấp dưới
  - Có 13 hệ thống máy chủ gốc trên mạng Internet (<http://www.root-servers.org>)



17

17

## Hệ thống máy chủ DNS (tiếp)

- 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

18

18

## 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)

19

19

## Thông điệp DNS

- DNS Query và DNS Reply
  - Chung khuôn dạng
- ID: Định danh của truy vấn
- 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 cho các địa chỉ khác

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

20

20

## Ví dụ: dig linux.com



```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 2
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

TTL: thời gian(s) lưu giữ  
trả lời trong cache

21

21

## Ví dụ: dig linux.com



```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 2
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

Tên các máy chủ DNS server trả lời truy vấn.  
Nếu phần ANSWER rỗng, DNS Resolver gửi  
truy vấn tới các máy chủ này

22

22

## Ví dụ: dig linux.com

```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 3
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

Địa chỉ IP của các máy chủ trả lời truy vấn.  
Thông tin này được lưu vào cache

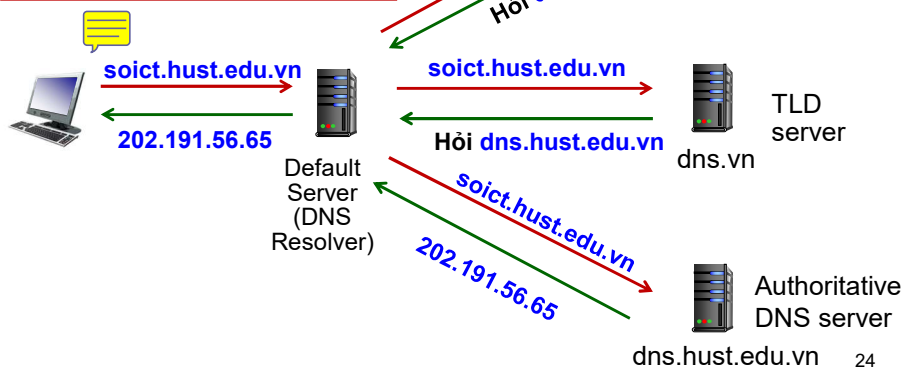
23

23

## Phân giải tương tác

### • Cơ chế mặc định trên các máy chủ DNS

Tải đặt lên máy chủ tên miền gốc rất lớn. Thực tế máy chủ DNS cục bộ thường đã biết các máy chủ TLD (cơ chế cache).

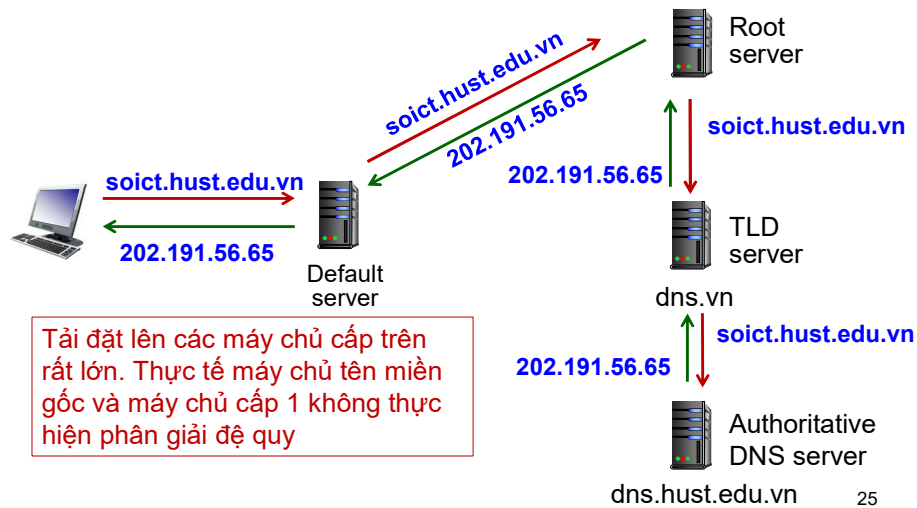


24

24

## Phân giải đệ quy

- Tùy chọn mở rộng



25

## 3. HTTP và WWW

26

26

## HTTP và Web



- Internet trước thập kỷ 1990s:
  - Hầu như chỉ sử dụng hạn chế trong cơ quan chính phủ, phòng nghiên cứu...
  - Các dịch vụ email, FTP không phù hợp cho chia sẻ thông tin đại chúng
  - Không có cơ chế hiệu quả để liên kết các tài nguyên thông tin nằm rải rác trên Internet
- Năm 1990, Tim Berners-Lee giới thiệu World Wide Web:
  - Trao đổi thông tin dưới dạng siêu văn bản (hypertext) sử dụng ngôn ngữ HTML (Hypertext Markup Language)
  - Các đối tượng không cần đóng gói “tất cả trong một” như trên các văn bản trước đó
  - Siêu văn bản chỉ chứa chứa liên kết (hyperlink) tới các đối tượng khác (định vị bằng URL).

27

27

## Uniform Resource Locator



- Định vị một tài nguyên bất kỳ trên mạng và cách thức để truy cập tài nguyên đó

`protocol://hostname[:port]/directory-path/resource`

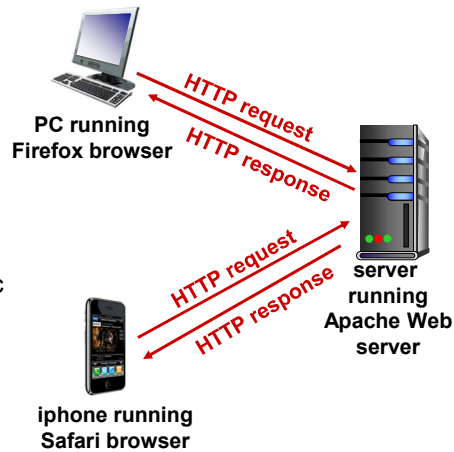
- *protocol*: Giao thức (http, ftp, https, smtp, rtsp...)
- *hostname*: tên miền, địa chỉ IP
- *port*: cổng ứng dụng (có thể không cần)
- *directory path*: đường dẫn tới tài nguyên
- *resource*: định danh của tài nguyên

28

28

## HTTP và Web

- WWW: World Wide Web
  - trao đổi dữ liệu siêu văn bản HTML (HyperText Markup Language) trên mạng
- HTTP: HyperText Transfer Protocol
  - Mô hình Client/Server
  - Client yêu cầu truy nhập tới các trang web (chứa các đối tượng web) và hiển thị chúng trên trình duyệt
  - Server: Nhận yêu cầu và trả lời cho client



29

29

## HTTP hoạt động ntn?

- Server mở một TCP socket chờ yêu cầu kết nối tại cổng 80 (default)
- Client khởi tạo một liên kết TCP tới server
- Server chấp nhận yêu cầu, tạo liên kết
- Trao đổi thông điệp HTTP (giao thức ứng dụng)
  - HTTP Request
  - HTTP Response
- Đóng liên kết TCP

30

30

## Khuôn dạng HTTP request

- Mã ASCII (dễ dàng đọc được dưới dạng văn bản)

request line  
(GET, POST,  
HEAD commands)

header  
lines

carriage return,  
line feed at start  
of line indicates  
end of header lines

```
GET /~tungbt/index.htm HTTP/1.1\r\n
Host: soict.hust.edu.vn\r\n
User-Agent: Mozilla/5.0
Accept: text/html,application/xhtml+xml\r\n
Accept-Language: en-us,en;q=0.5\r\n
Accept-Encoding: gzip,deflate\r\n
Accept-Charset: ISO-8859-1,utf-8;q=0.7\r\n
Keep-Alive: 115\r\n
Connection: keep-alive\r\n
\r\n
```

31

31

## Các phương thức trong thông điệp yêu cầu

### HTTP/1.0

- GET
- POST
- HEAD
  - yêu cầu máy chủ loại một số đối tượng ra khỏi thông điệp trả lời

### HTTP/1.1

- GET, POST, HEAD
- PUT
  - tải file lên máy chủ, đường dẫn chỉ ra trong URL, file để trong body
- DELETE
  - Xóa file chỉ ra bởi đường dẫn

Lưu ý: Có 2 cách để gửi tham số đến server: POST hoặc GET

<http://www.google.com/search?q=computer+network&flags=68&num=10>

32

32



## Khuôn dạng HTTP response



status line  
(protocol  
status code  
status phrase)

header  
lines

data, e.g.,  
requested  
HTML file

```
HTTP/1.1 200 OK\r\n
Date: Thu, 31 Jul 2014 00:00:14 GMT\r\n
Server: Apache/2.2.15 (CentOS)\r\n
Last-Modified: Wed, 30 Jul 2014 23:59:50 GMT\r\n
ETag: "17dc6-a5c-bf716880"\r\n
Accept-Ranges: bytes\r\n
Content-Length: 2652\r\n
Connection: close\r\n
Content-Type: text/html; charset=UTF-8\r\n
\r\n
data data data data data ...
```

33

33

## Mã trạng thái trả lời



Trong dòng đầu tiên của thông điệp trả lời, ví dụ

### 200 OK

- request succeeded, requested object later in this message

### 301 Moved Permanently

- requested object moved, new location specified later in this message (Location:)

### 400 Bad Request

- request message not understood by server

### 404 Not Found

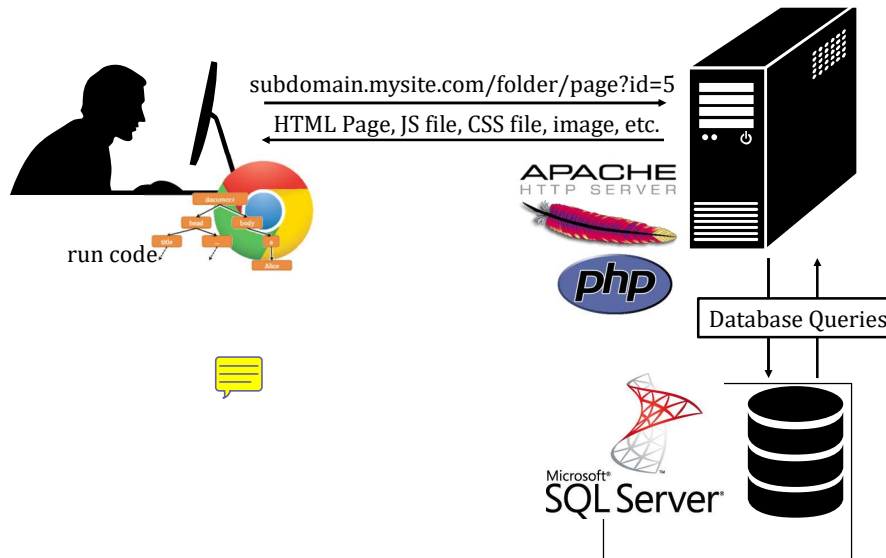
- requested document not found on this server

### 505 HTTP Version Not Supported

34

34

## Kiến trúc chung của các dịch vụ web



35

## Hiển thị (rendering) nội dung trang web

- Mô hình xử lý cơ bản tại trình duyệt: mỗi cửa sổ hoặc 1 frame:
  - Nhận thông điệp HTTP Response
  - Hiển thị:
    - Xử lý mã HTML, CSS, Javascripts
    - Gửi thông điệp HTTP Request yêu cầu các đối tượng khác (nếu có)
    - Bắt và xử lý sự kiện
- Các sự kiện có thể xảy ra:
  - Sự kiện của người dùng: `OnClick`, `OnMouseOver`...
  - Sự kiện khi hiển thị: `OnLoad`, `OnBeforeUnload`...
  - Theo thời gian: `setTimeout()`, `clearTimeout()`...

36

36

## Các liên kết HTTP

### HTTP không duy trì

- Chỉ một đối tượng web được gửi qua liên kết TCP
- Sử dụng mặc định trong HTTP/1.0
- HTTP 1.0: RFC 1945

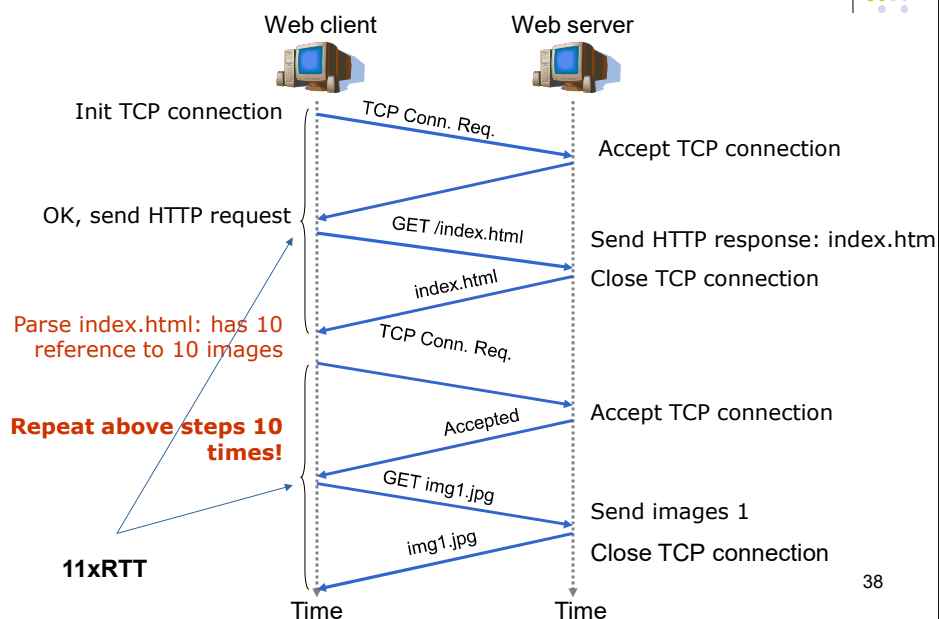
### HTTP có duy trì

- Nhiều đối tượng có thể được gửi qua một liên kết TCP.
- Sử dụng mặc định trong HTTP/1.1
- HTTP 1.1: RFC 2068

37

37

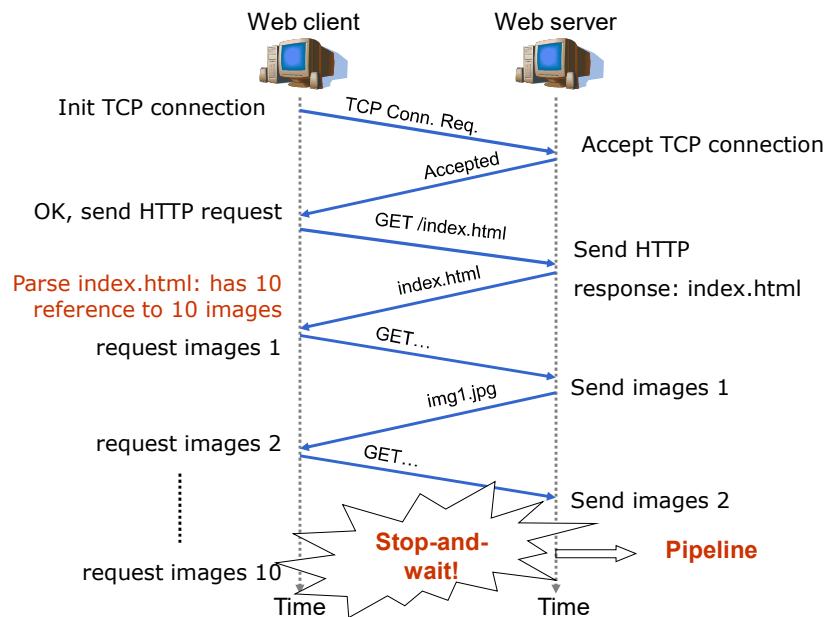
## Hoạt động của HTTP/1.0



38

38

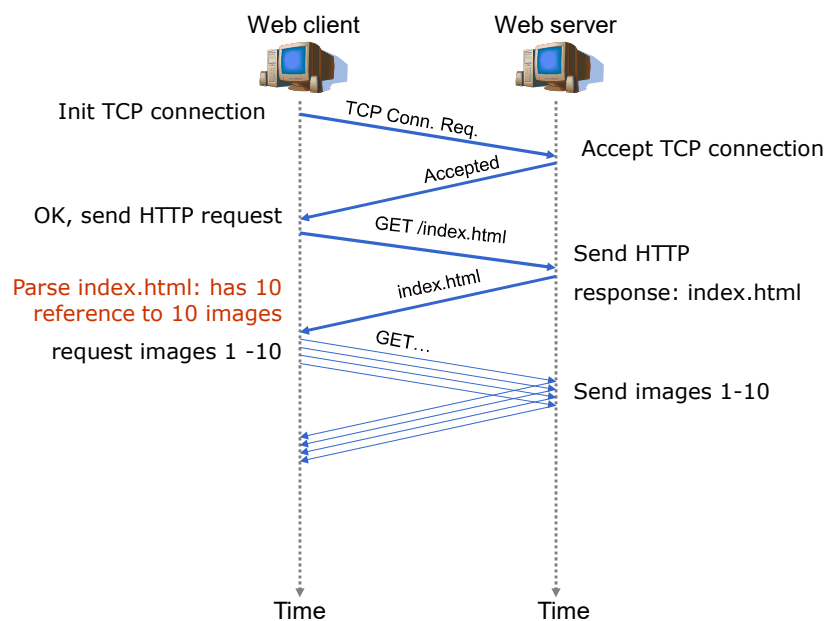
## Hoạt động của HTTP/1.1



39

39

## HTTP/1.1 với pipeline



40

40

## HTTP là giao thức stateless



- Một phiên hoạt động của HTTP:
  - Trình duyệt kết nối với Web server
  - Trình duyệt gửi thông điệp yêu cầu HTTP Request
  - Web server đáp ứng với một thông điệp HTTP Response
  - ...lặp lại...
  - Trình duyệt ngắt kết nối
- Các thông điệp HTTP Request được xử lý độc lập
- Web server không ghi nhớ trạng thái của phiên HTTP
  - Nếu dịch vụ Web cần xác thực người dùng thì người dùng sẽ phải đăng nhập lại cho mỗi thông điệp HTTP Request gửi đi ☹

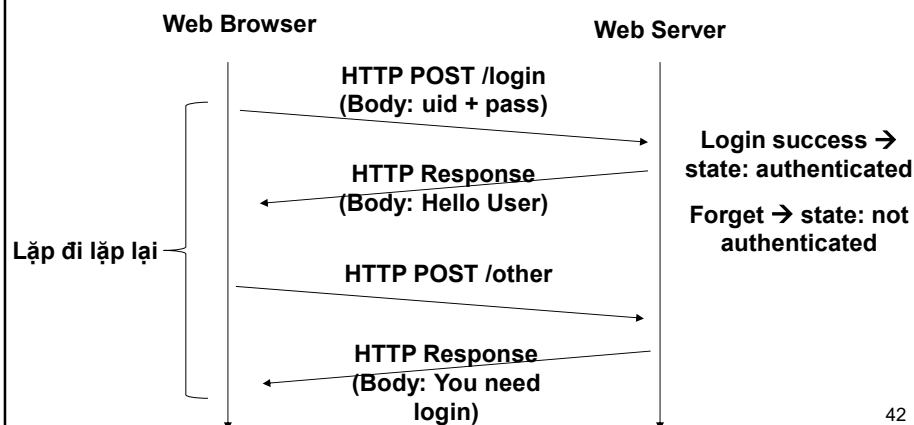
41

41

## HTTP là stateless – Ví dụ



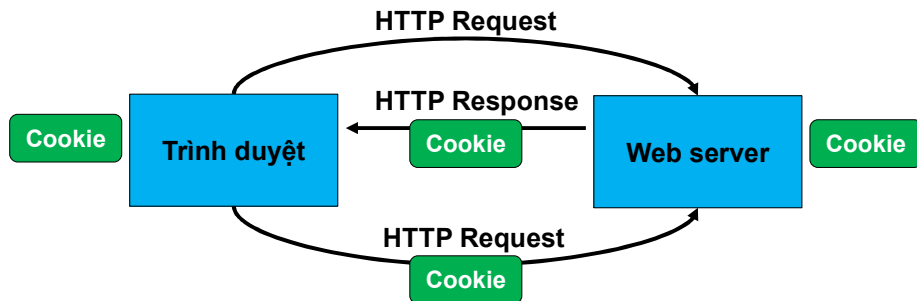
- Quản lý phiên truy cập: Nếu HTTP Request chỉ được xử lý bởi phần mềm Web Server



42

42

# HTTP Cookie



- Cookie: dữ liệu do ứng dụng Web tạo ra, chứa thông tin trạng thái của phiên làm việc
  - Server có thể lưu lại cookie (một phần hoặc toàn bộ)
- Sau khi xử lý yêu cầu, Web server trả lại thông điệp HTTP Response với cookie đính kèm
  - Set-Cookie: key = value; options;
- Trình duyệt lưu cookie
- Trình duyệt gửi HTTP Request tiếp theo với cookie được đính kèm

43

43

# HTTP Cookie - Ví dụ

## HTTP Response

```
HTTP/1.1 200 OK
Server: nginx/1.10.1
Date: Mon, 14 Nov 2016 09:19:19 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/5.4.45
Set-Cookie: vflastvisit=1479115159; expires=Tue, 14-Nov-2017 09:19:19 GMT; path=/; domain=vozforums.com; secure
Set-Cookie: vflastactivity=0; expires=Tue, 14-Nov-2017 09:19:19 GMT; path=/; domain=vozforums.com; secure
Expires: 0
Cache-Control: private, post-check=0, pre-check=0, max-age=0
Pragma: no-cache
Content-Encoding: gzip
```

44

44

## HTTP Cookie - Ví dụ



- HTTP Request

```
GET /clientscript/vbulletin_important.css?v=380 HTTP/1.1
Host: vozforums.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:49.0) Gecko/20100101 Firefox/49.0
Accept: text/css,*/*;q=0.1
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://vozforums.com/
Cookie: vflastvisit=1479115159; vflastactivity=0
Connection: keep-alive
```

45

45

## Web caching



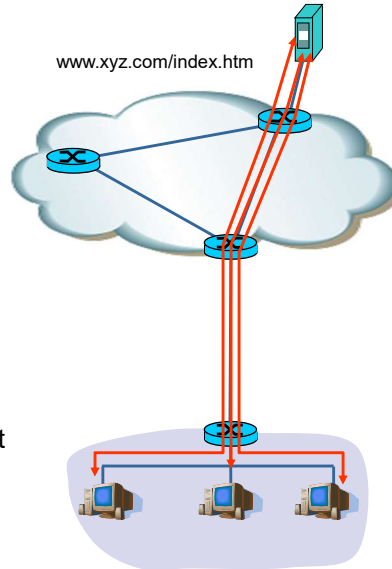
- Các trang web có thể được lưu trên máy trạm cục bộ
- Sử dụng local cache để
  - Duyệt web offline
  - Duyệt các trang web hiệu quả hơn

46

46

## Web caching

- “Cache”: Bộ nhớ đệm
- Khái niệm bộ nhớ cache trong máy tính
  - L1 cache, L2 cache
  - “cache miss”, “cache hit”
- Xem xét trường hợp sau:
  - Một tổ chức có một đường nối tới Internet
  - Tất cả lưu lượng truy cập web đều đi qua liên kết này
  - Nhiều NSD web có thể cùng truy nhập tới cùng một nội dung
  - Giải pháp cải tiến?

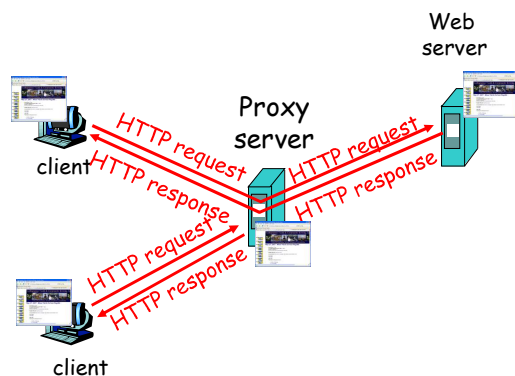


47

47

## Sử dụng bộ đệm - web proxy

- NSD đặt tham số kết nối truy cập web của trình duyệt qua một máy chủ proxy
- trình duyệt gửi yêu cầu đến proxy
  - Miss: Proxy gửi yêu cầu tới máy chủ web, trả lời trình duyệt và lưu đệm đối tượng web
  - Hit: Proxy trả đối tượng web cho trình duyệt



48

48



## Web proxy

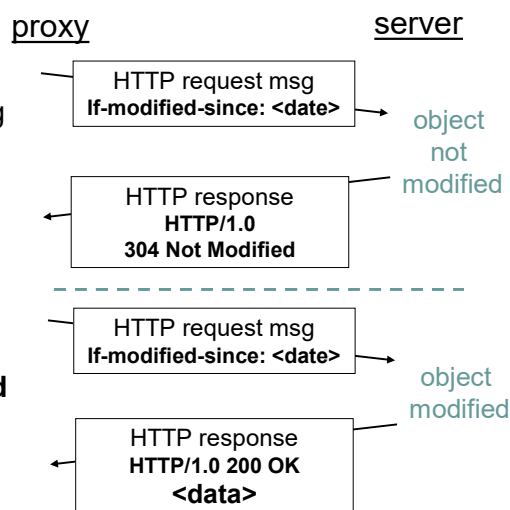
- Proxy: Vừa là client, vừa là server
- Sử dụng bởi các ISP nhỏ, các tổ chức như trường học, công ty...
- Ảnh hưởng của proxy
  - Làm giảm lưu lượng web trên đường ra Internet
  - Có thể làm giảm thời gian đáp ứng
  - Thử phân tích vài trường hợp
    - cache hit
    - cache miss
    - proxy bị quá tải
    - Trang web thay đổi/trang web động?

49

49

## Phương thức GET có điều kiện

- **Mục đích:** Máy chủ sẽ không gửi đối tượng web nếu proxy còn lưu giữ thông tin cập nhật
- Proxy: chỉ ra thời gian cũ của đối tượng  
**If-modified-since: <date>**
- server: Xác nhận lại có thay đổi hay không:  
**HTTP/1.0 304 Not Modified**



50

50

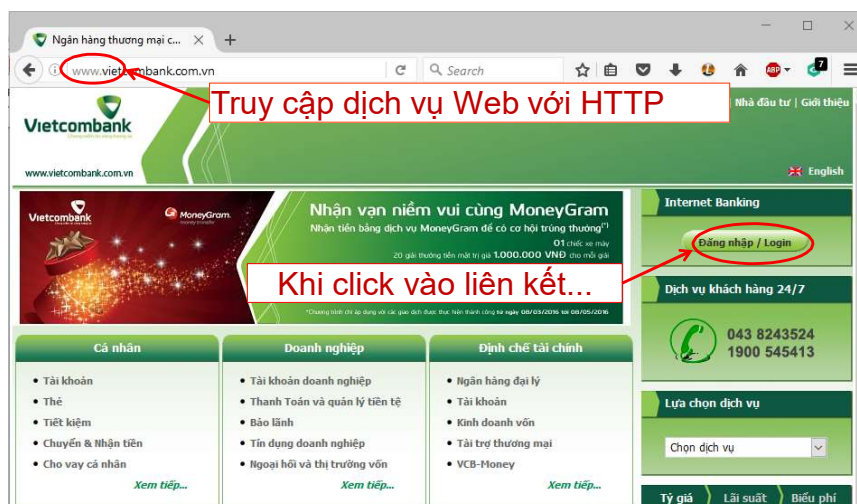
# HTTPS

- Hạn chế của HTTP:
  - Không có cơ chế để người dùng kiểm tra tính tin cậy của Web server → lỗ hổng để kẻ tấn công giả mạo dịch vụ hoặc chèn mã độc vào trang web HTML
  - Không có cơ chế mã hóa giữ mật → lỗ hổng để kẻ tấn công nghe lén đánh cắp thông tin nhạy cảm
- Secure HTTP: sử dụng liên kết SSL/TLS thay cho TCP để truyền các thông điệp HTTP
  - Xác thực:
    - Người dùng truy cập vào đúng Website mong muốn
    - Dữ liệu trong quá trình truyền không bị thay đổi
  - Bảo mật: dữ liệu được giữ bí mật trong quá trình truyền
- Số hiệu cổng ứng dụng: 443

51

51

# HTTP trên trình duyệt Web



52

52

## HTTPS trên trình duyệt Web

Truy cập Web với HTTPS

Đăng nhập

JOINT STOCK COMMERCIAL BANK... (VN) https://www.vie

Chấp cánh ước mơ cùng Vietcombank

Chương trình khuyến mại hấp dẫn từ 06/10/2016 đến hết 31/12/2016

Dành cho Khách hàng mua mới Bảo An Thành Tài, Bảo An Tài Trĩ của Vietcombank


**Vietcombank**  
Chặng mới kỷ niệm 40 năm  
NGÂN HÀNG TRỰC TUYẾN  
VCB - iB@nking

Đăng nhập

53

53

## HTTPS

- Tuy nhiên, HTTPS có thể gây hiểu nhầm cho người dùng rằng trang web là an toàn:
  - Người dùng bất cẩn vì chỉ chú ý biểu tượng 
  - HTTPS không chống lại được các dạng tấn công khai thác điểm yếu của Website

54

54

## 4. Email



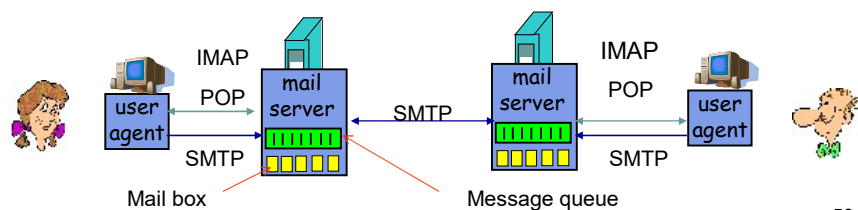
55

55

## Thư điện tử



- MUA (Mail User Agent)
  - Lấy thư từ máy chủ, gửi thư đến máy chủ
  - e.g. Outlook, Thunderbird...
- MTA (Mail Transfer Agent):
  - Chứa hộp thư đến của NSD (mail box)
  - Hàng đợi để gửi thư đi
  - e.g. Sendmail, MS Exchange...
- Giao thức:
  - Chuyển thư: SMTP-Simple Mail Transfer Protocol
  - nhận thư
    - POP – Post Office Protocol
    - IMAP – Internet Mail Access Protocol



56

56

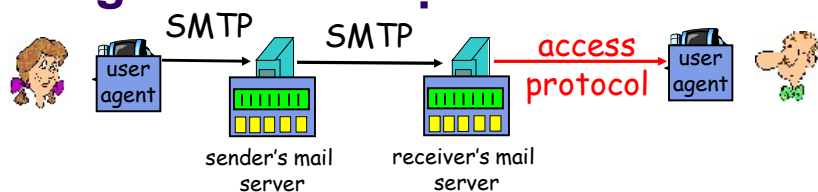
## Giao thức SMTP

- RFC 2821
- TCP, port 25: Chuyển thư từ client đến server và giữa các server với nhau
- Tương tác yêu cầu/trả lời
  - Yêu cầu: Lệnh với mã ASCII
  - Trả lời: mã trạng thái và dữ liệu

57

57

## Các giao thức nhận thư



- POP: Post Office Protocol [RFC 1939]
  - Đăng nhập và lấy hết thư về
- IMAP: Internet Mail Access Protocol [RFC 1730]
  - Phức tạp hơn POP
  - Cho phép lưu trữ và xử lý thư trên máy chủ

58

58

## Web Mail

- Sử dụng Web browser như một MUA
- MUA và MTA giao tiếp thông qua HTTP
- Mails được lưu trữ trên máy chủ
- E.g.
  - Gmail,
  - Hotmail,
  - Yahoo! Mail, etc.
- Ngày nay, rất nhiều các MTA cho phép truy cập thông qua giao diện web
  - <http://mail.hust.edu.vn>
  - <http://mail.soict.hust.edu.vn>

59

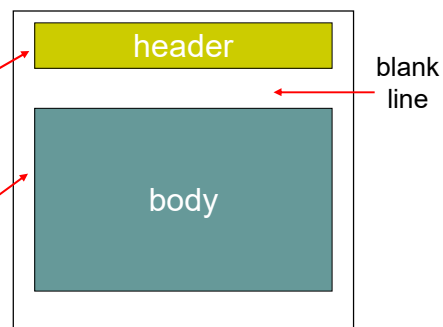
59

## Khuôn dạng thông điệp thư điện tử

SMTP: Giao thức để truyền thư

RFC 822: Định nghĩa khuôn dạng

- Phần đầu
  - To:
  - From:
  - Subject:
- Phần thân
  - mã hóa dưới dạng mã ASCII



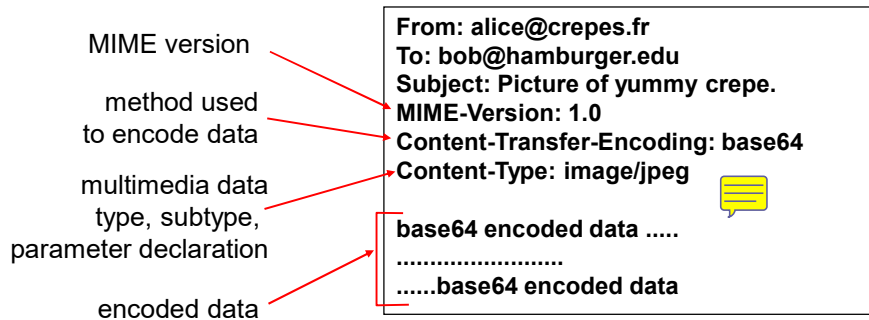
60

60

## Để chuyển dữ liệu đa phương tiện: multimedia extensions



- MIME: multimedia mail extension, RFC 2045, 2056
- Thêm một dòng trong phần đầu chỉ rõ khuôn dạng dữ liệu gửi đi



61

61

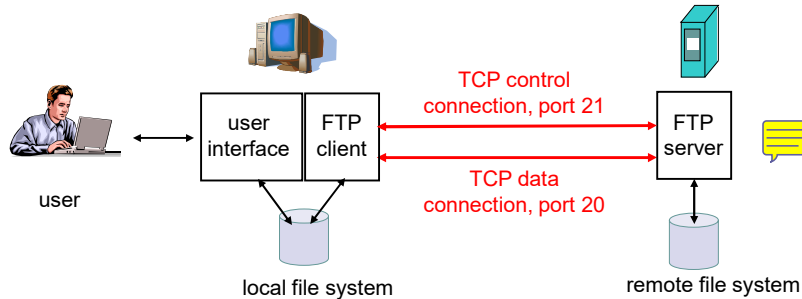
## 5. Ứng dụng truyền tệp



62

62

# FTP: File Transfer Protocol



- Mô hình Client-server
- Trao đổi file giữa các máy
- RFC 959
- Sử dụng TCP, cổng 20, 21
- Điều khiển **Out-of-band** :
  - Lệnh của FTP : cổng 21
  - Dữ liệu: cổng 20
- NSD phải đăng nhập trước khi truyền file
- Một số server cho phép NSD với tên là anonymous

63

63

## Lệnh và mã trả lời

### Một số ví dụ

- **USER *username***
- **PASS *password***
- **LIST** : trả về danh sách file
- **RETR filename** Lấy file
- **STOR filename** Đặt file lên máy chủ

### Ví dụ về mã trả lời

- **331 Username OK, password required**
- **125 data connection already open; transfer starting**
- **425 Can't open data connection**
- **452 Error writing file**

64

64



## Ví dụ về ftp client

### Command line

C:\Documents and Settings\hongson>ftp  
ftp> ?

Commands may be abbreviated. Commands are:

!	<b>delete</b>	literal	prompt	send
?	debug	ls	<b>put</b>	status
append	dir	mdelete	pwd	trace
ascii	disconnect	mdir	quit	type
bell	<b>get</b>	mget	quote	user
binary	glob	mkdir	recv	verbose
<b>bye</b>	hash	mls	remotehelp	
cd	help	mput	rename	
close	lcd	<b>open</b>	rmdir	

GUI FTP clients: IE, Firefox, GFTP, ....

65

65

## Tóm tắt

- Mô hình ứng dụng
  - Client-server vs. P2P
- Một số ứng dụng và giao thức
  - HTTP
  - Mail
  - FTP
- Về nhà, hãy tìm hiểu thêm
  - P2P
  - .....
- Giao diện lập trình Socket

66

66



## Tài liệu tham khảo

- Keio University
- “Computer Networking: A Top Down Approach”, J.Kurose
- “Computer Network”, Berkeley University