Introduction to WWW

Introduction to Internet and Web

부산대학교 정보·의생명 공학대학
정보컴퓨터공학부

Table of Contents

PRINCIPLES OF APPLICATION

Network Applications

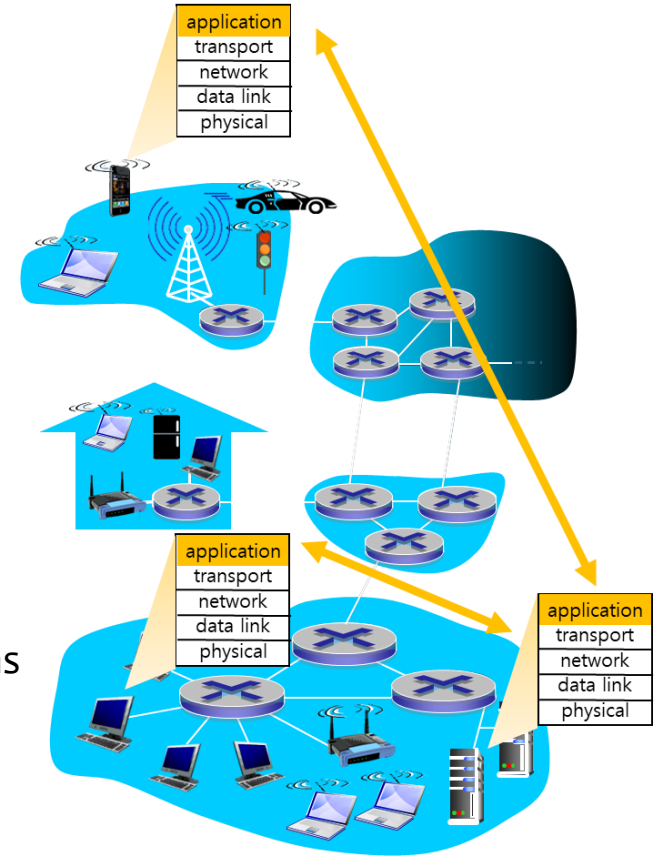
❖ Types

- email
- web (server software, browser)
- P2P file sharing
- SNS (Social Network Service)
- messenger program
- online-game
- streaming stored video (YouTube, Netflix)

❖ Run on (different) end systems

- network-core devices do not run user applications

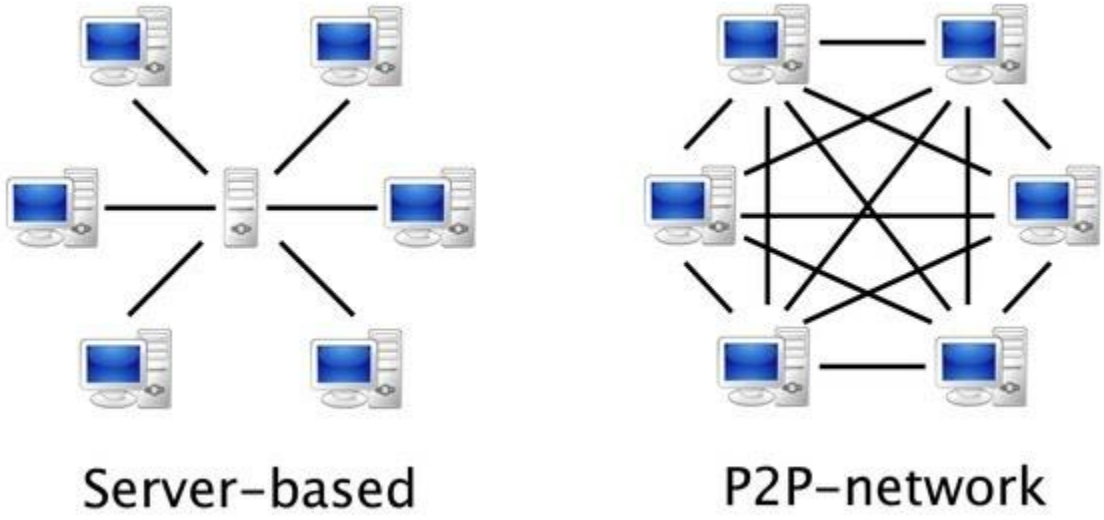
❖ Communicate over network



Application Architectures (1/3)

❖ Two kinds of application structures:

- Client-server model
- Peer-to-peer (P2P) model

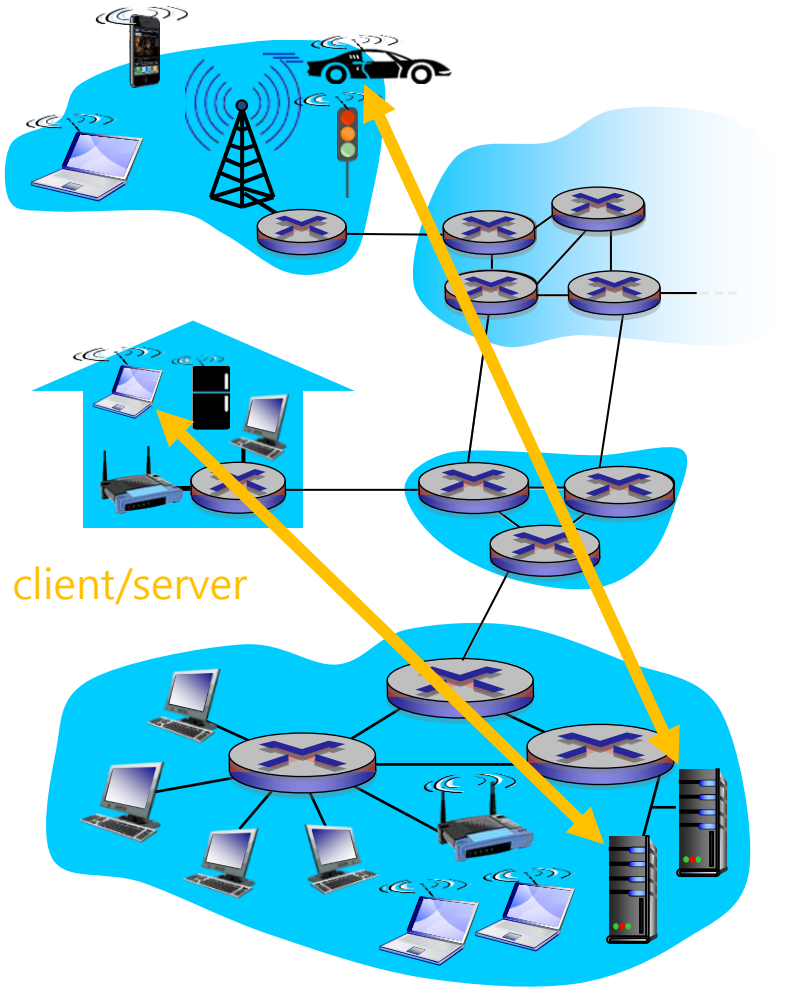


출처 - https://www.quora.com/Whats-difference-between-p2p-and-cdn/

Application Architectures (2/3)

❖ Client-server model

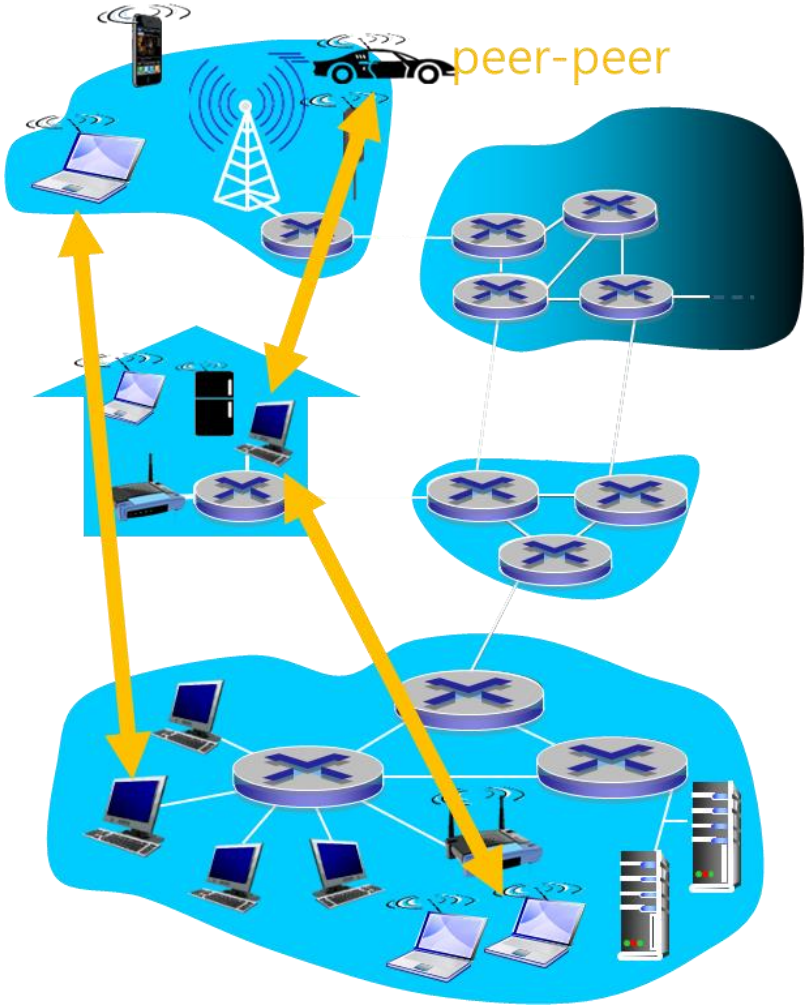
- server:
  - always-on
  - permanent IP address
  - data centers for scaling
- client:
  - communicate with server
  - may be intermittently connected
  - may have dynamic IP addresses
  - do not communicate directly with each other



Application Architectures (3/3)

❖ Peer-to-peer (P2P) model

- no always-on server
- arbitrary end systems directly communicate
- self scalability – new peers bring new service capacity, as well as new service demands
- peers are intermittently connected and change IP addresses
  - complex management



Requirements of Network Applications

application	data loss	throughput	time sensitive
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video:10kbps-5Mbps	yes, 100's msec
stored audio/video	loss-tolerant	same as above	yes, few secs
interactive games	loss-tolerant	few kbps up	yes, 100's msec
text messaging	no loss	elastic	yes and no

❖ Who does meet these requirements?

- Transport layer protocols - TCP (reliable) / UDP (unreliable)!

Application & Transport Protocol Pairs

❖ Application-layer protocol

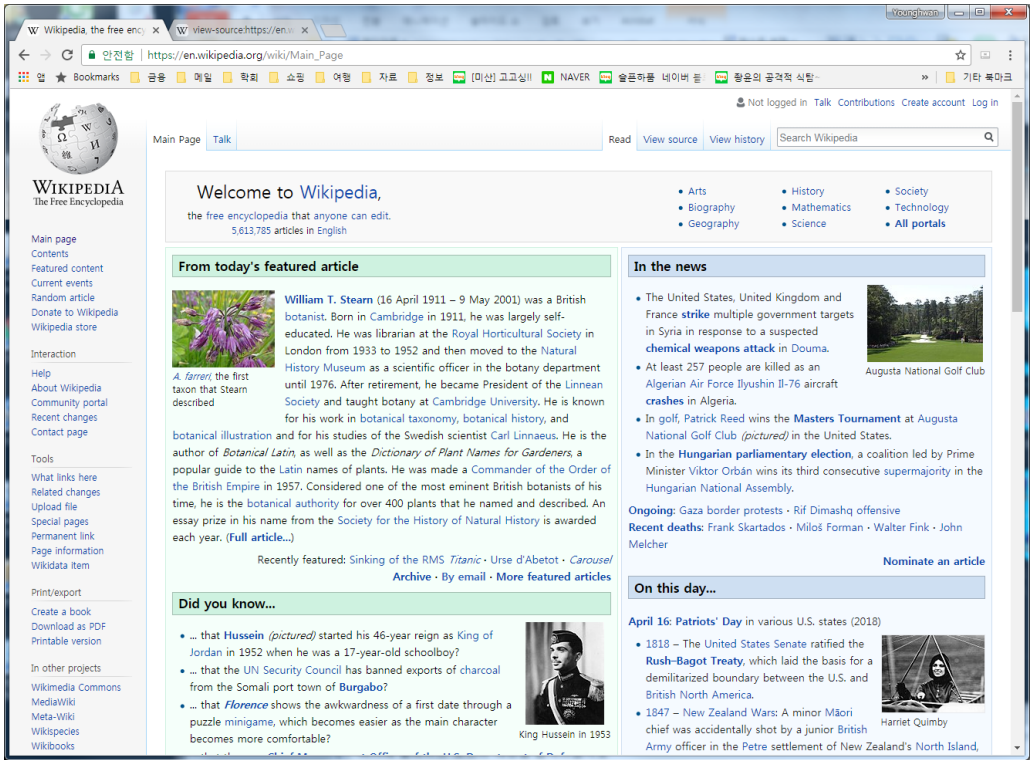
- The types of messages exchanged, for example, request messages and response messages
- The syntax of the various message types, such as the fields in the message and how the fields are delineated
- The semantics of the fields, that is, the meaning of the information in the fields
- Rules for determining when and how a process sends messages and responds to messages

Application	application layer protocol	underlying transport protocol
e-mail	SMTP [RFC 2821]	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), RTP [RFC 1889]	TCP or UDP
Internet telephony	SIP, RTP, proprietary (e.g., Skype)	TCP or UDP

WEB AND HTTP

World Wide Web (WWW) (1/2)

- Web page consists of objects
- Object can be HTML file, JPEG image, Java applet, audio file,…



World Wide Web (WWW) (2/2)

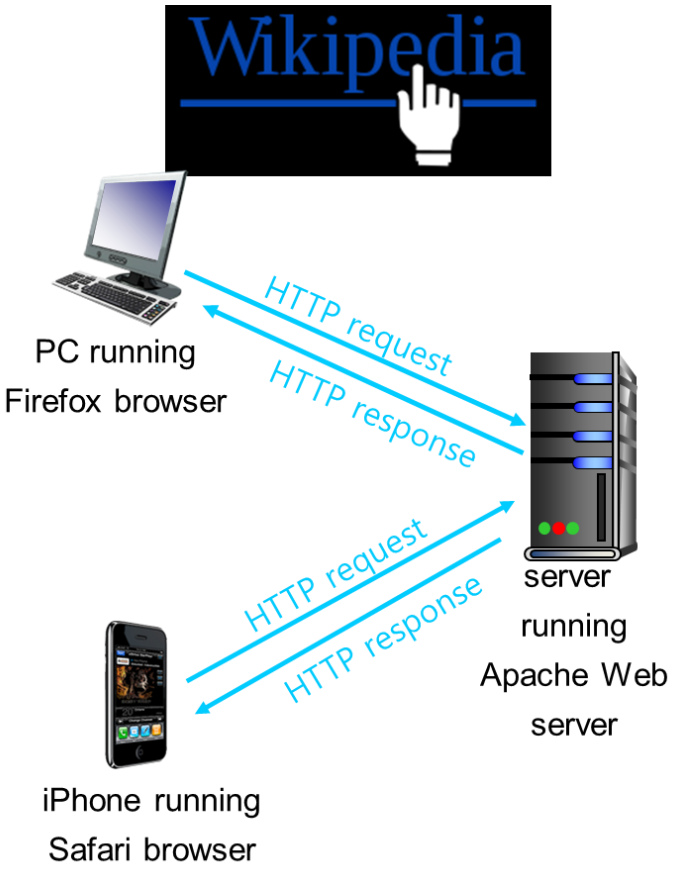
- Web page is described by HTML-file which includes several referenced objects
- Each object is addressable by a URL, e.g.,

`www.someschool.edu/someDept/pic.gif`

host name | path name

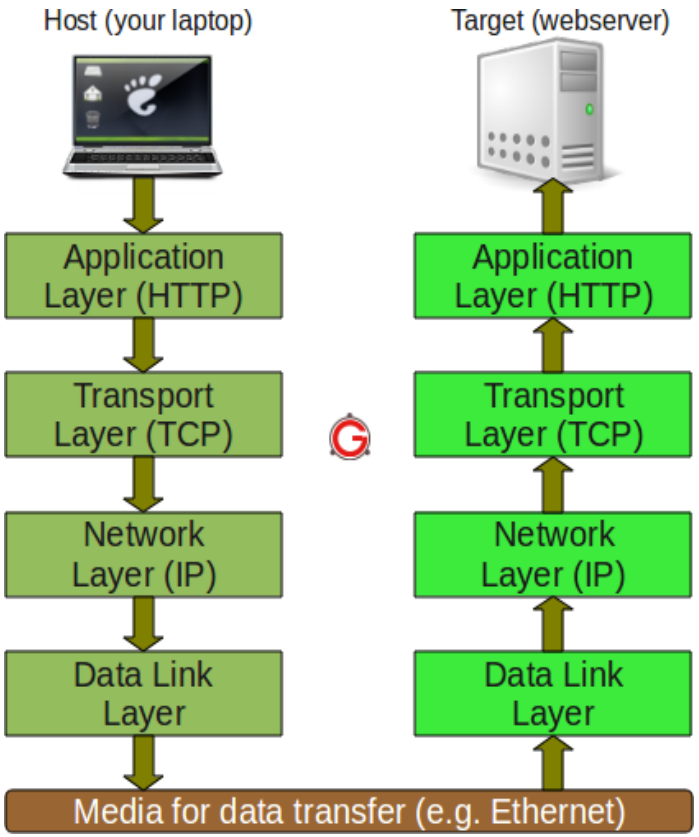
HTTP Overview (1/2)

- HTTP (HyperText Transfer Protocol)
- Web's application layer protocol
- Hyperlink: a reference to data the reader can directly follow by clicking
- Client/server model
  - client: browser that requests, receives, and "displays" Web objects
  - server: web server sends objects in response to requests



HTTP Overview (2/2)

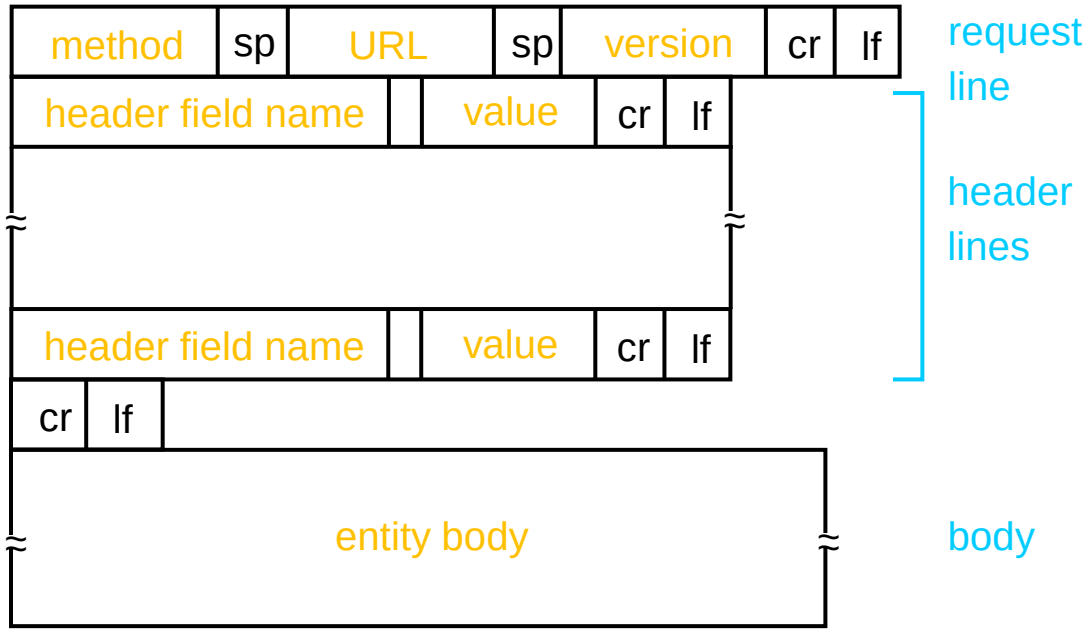
- Based on TCP
- Client initiates TCP connection (creates socket) to server
- Server accepts TCP connection from client
- HTTP messages (application-layer protocol messages) exchanged between browser (HTTP client) and Web server (HTTP server)
- TCP connection closed



출처 - http://tamil-it-guru.blogspot.com/2017/02/tcpip-protocol-fundamentals-explained.html/

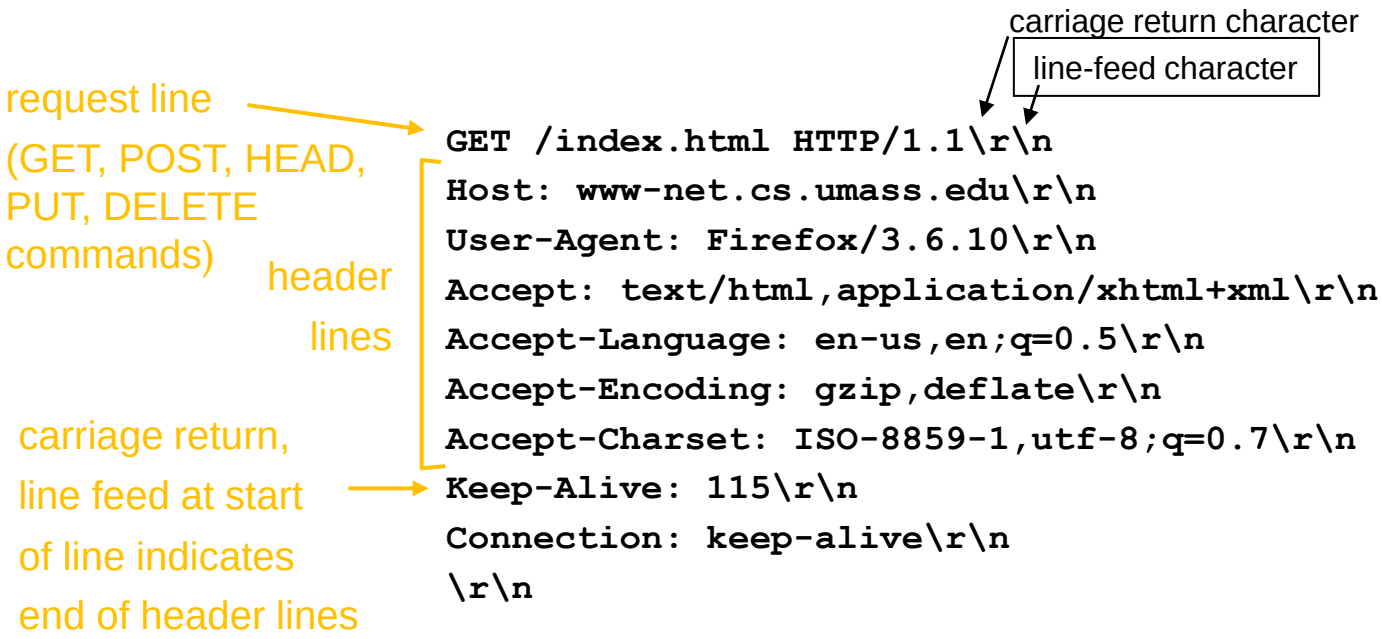
HTTP Message

- Two types of messages: *request*, *response*
- Message format



HTTP Request Message

- **ASCII (human-readable format)**



HTTP Response Message

status line (protocol status code status phrase)

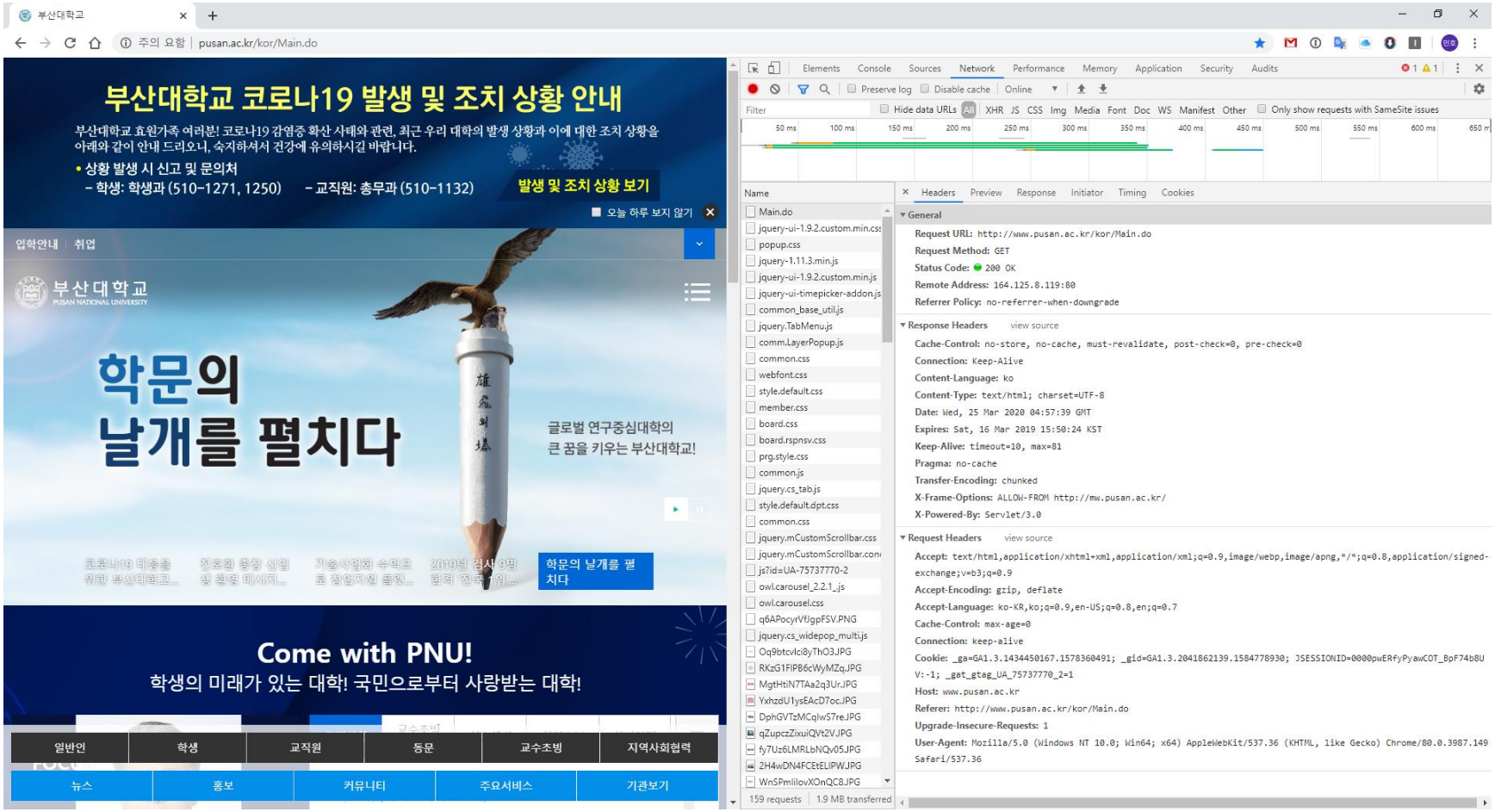
header lines

data, e.g., requested HTML file

```
HTTP/1.1 200 OK\r\n
Date: Sun, 26 Sep 2010 20:09:20 GMT\r\n
Server: Apache/2.0.52 (CentOS)\r\n
Last-Modified: Tue, 30 Oct 2007
   17:00:02 GMT\r\n
ETag: "17dc6-a5c-bf716880"\r\n
Accept-Ranges: bytes\r\n
Content-Length: 2652\r\n
Keep-Alive: timeout=10, max=100\r\n
Connection: Keep-Alive\r\n
Content-Type: text/html; charset=ISO-
   8859-1\r\n
\r\n
data data data data data ...
```

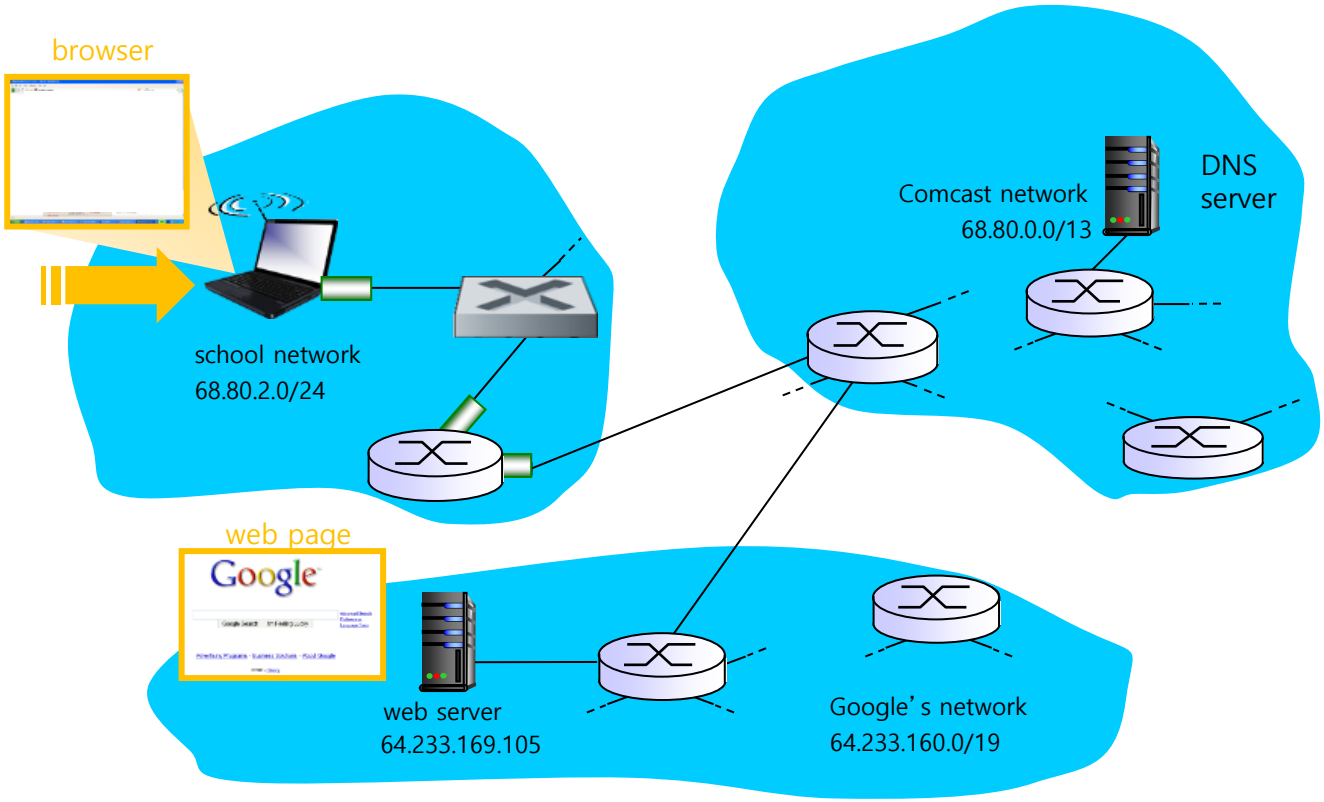
- **Response codes**
  - 200 OK
  - 301 Moved Permanently
  - 400 Bad Request
  - 404 Not Found
  - 505 HTTP Version Not Supported

HTTP Message

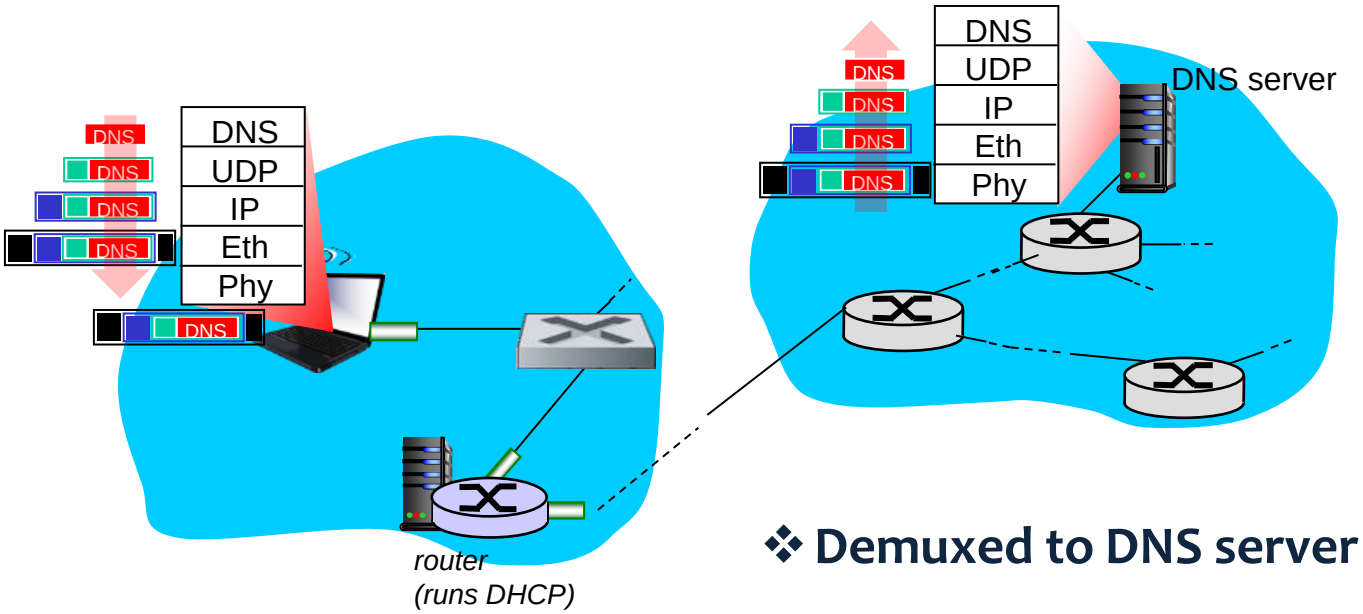


HOW TO DELIVER WEB REQUEST

Scenario in Real Life



DNS Query/Reply

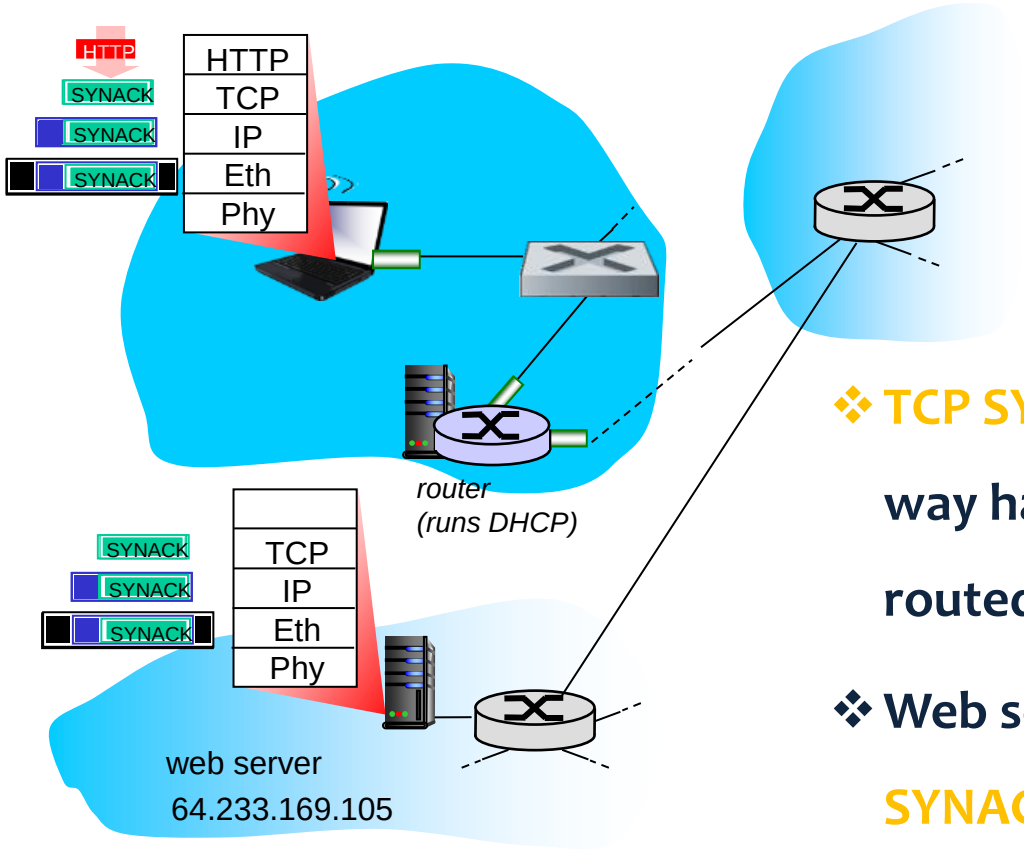


- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router

❖ Demuxed to DNS server

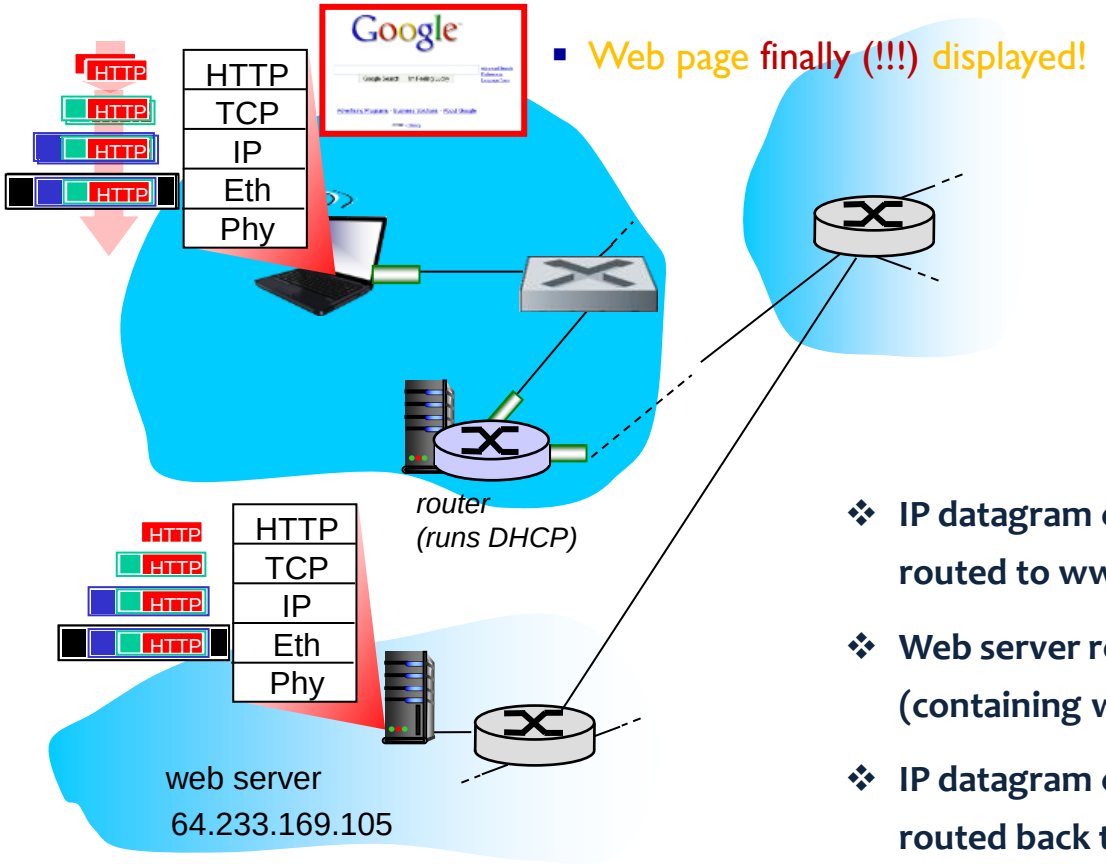
❖ DNS server replies to client with IP address of www.google.com

TCP Connection Carrying HTTP



- TCP SYN segment (step 1 in 3-way handshake) inter-domain routed to web server
- Web server responds with TCP SYNACK (step 2 in 3-way handshake)
- TCP connection established!

HTTP Request/Reply



- Web page finally (!!!) displayed!

- ❖ IP datagram containing HTTP request routed to www.google.com
- ❖ Web server responds with HTTP reply (containing web page)
- ❖ IP datagram containing HTTP reply routed back to client

NETWORK SECURITY

Types of Cryptography

Symmetric key cryptosystem

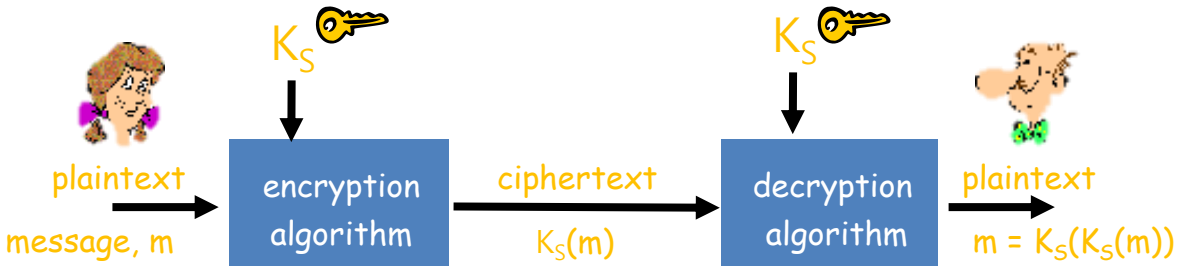
- the same key is used for encryption and decryption
- the key must be kept secret
- secret key system

Asymmetric key cryptosystem

- different keys are used for encryption and decryption
- one of the two keys is exposed to other users
- public key system

Symmetric Key Cryptography

- Bob and Alice share same (symmetric) key: K_S
- Q: How do Bob and Alice agree on key value?



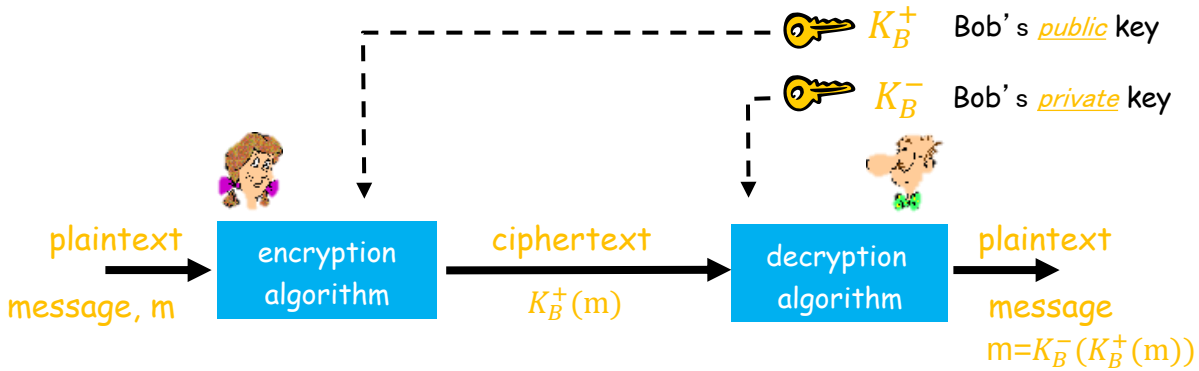
Asymmetric (Public) Key Cryptography

❖ Challenge of symmetric key cryptography

- "How to agree on key in first place?" (particularly, if never meet each other?)

❖ Asymmetric key cryptography

- sender, receiver do not share a secret key
- public encryption key known to all
- private decryption key known only to receiver



SSL/TLS

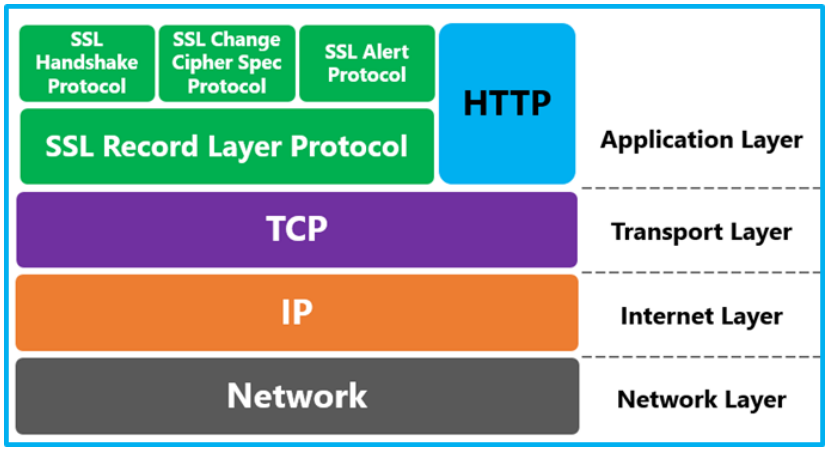
Securing TCP

❖ Transport layer protocols: TCP, UDP

- no encryption
- even passwords traversed Internet in cleartext

❖ SSL/TLS

- provides encrypted TCP connection at the application layer
- data integrity
- end-point authentication



출처 - https://linuxacademy.com/community/posts/show/topic/14103-lecture-elb-amp-ec2-logging-puzzled-about-http-vs-tcp/

SSL/TLS

❖ SSL (Secured Socket Layer)

- SSL v2.0 and v3.0: released in 1995 and 1996

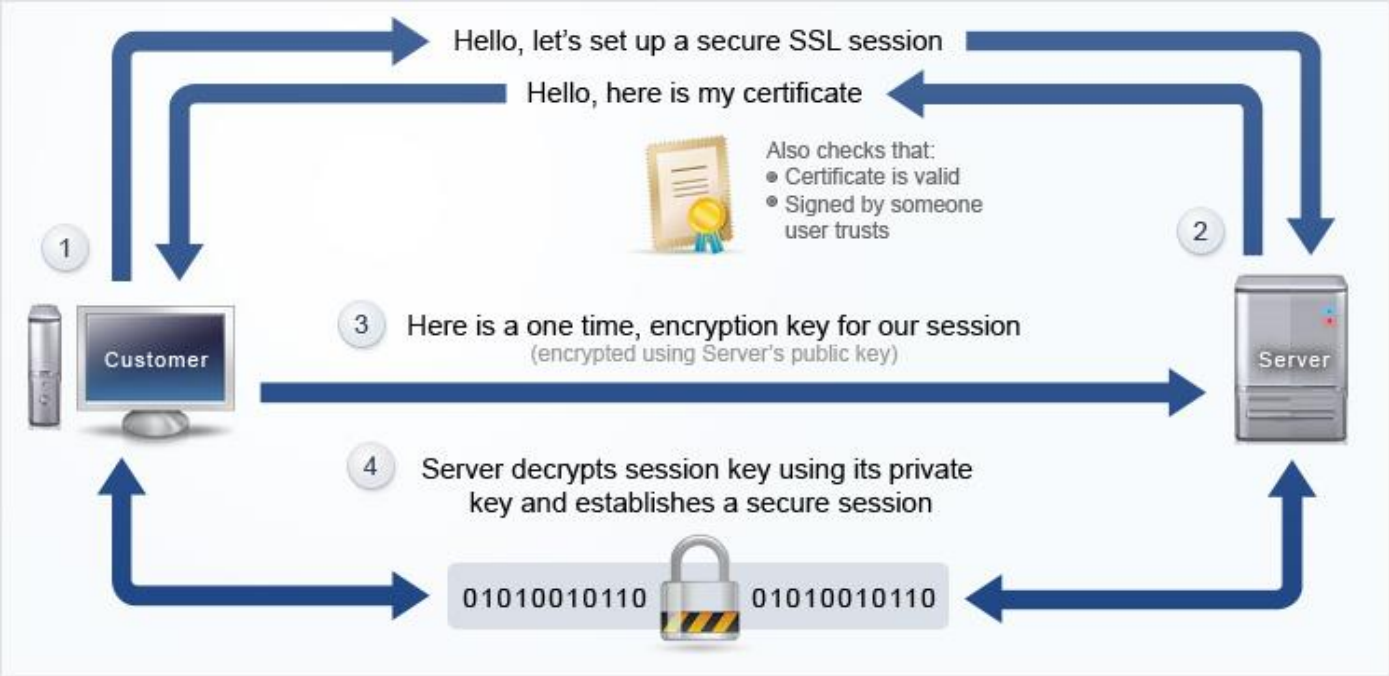
❖ TLS (Transport Layer Security)

- the improved version of SSL v3.0
- more secure but little slower due to the two-step communication processes, i.e., server authentication and actual data transfer

출처 - https://www.quora.com/What-is-SSL-TLS-SSH-Secure-DNS-and-HTTPS

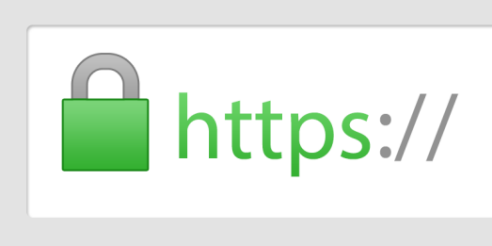
SSL/TLS Principle

❖ Usage of the public-private (asymmetric) key pair system

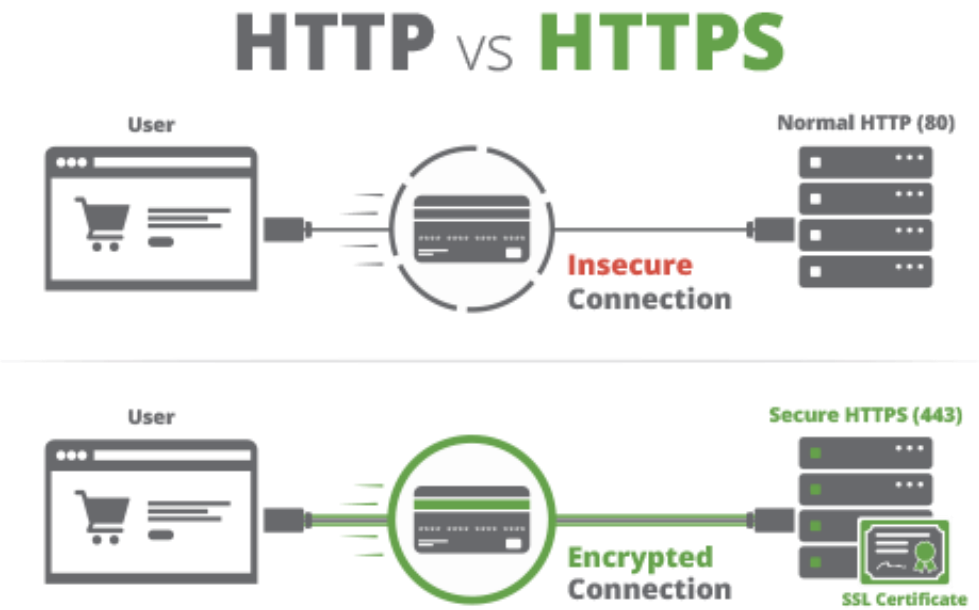


출처 - https://www.msctrustgate.com/ssl_id.php

HTTPS = HTTP + SSL/TLS



출처 - http://blog.getpostman.com/2017/12/05/set-and-view-ssl-certificates-with-postman/



출처 - https://sucuri.net/guides/how-to-install-ssl-certificate

ELECTRONIC MAIL

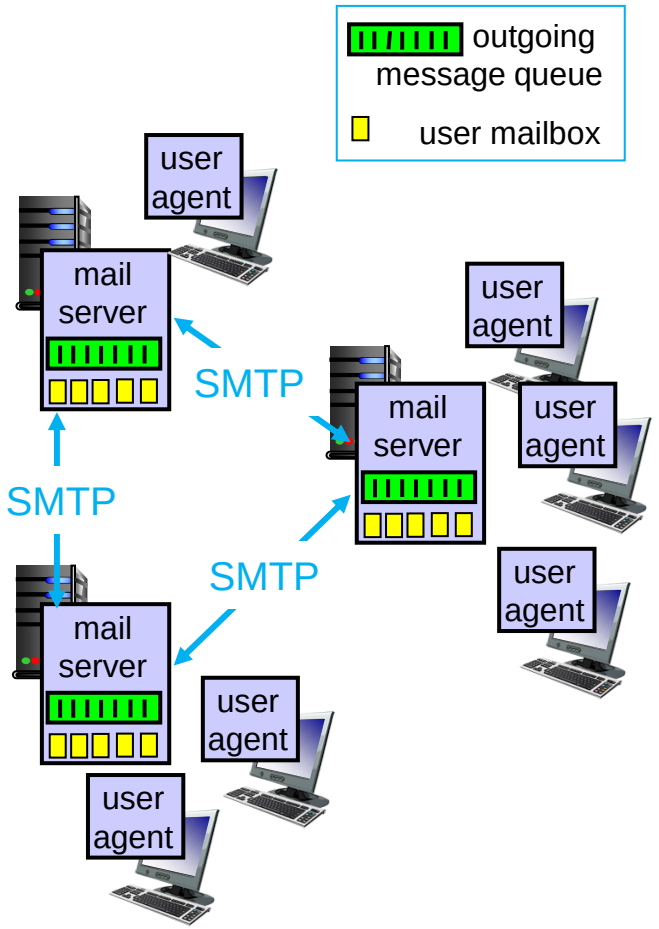
Electronic Mail

❖ Three components

- user agents (clients): editing, reading
- mail servers
- protocols: SMTP, POP3, IMAP, …

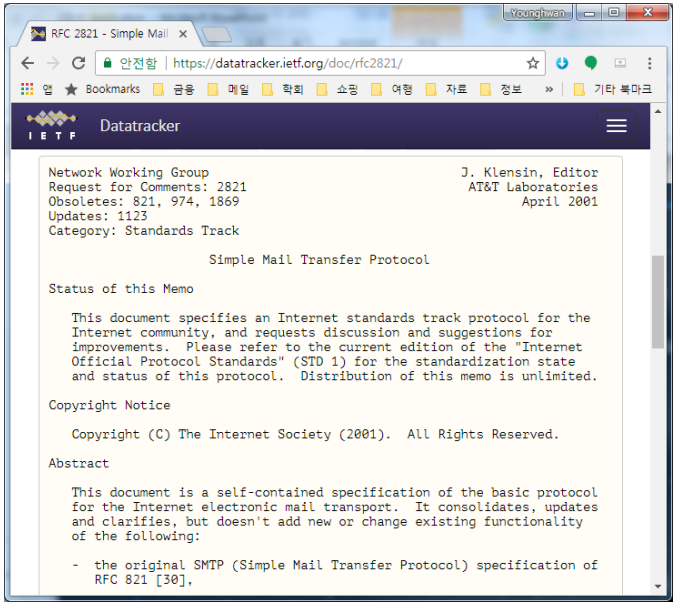
❖ Components of mail servers

- mailbox for incoming message
- message queue for outgoing message
- SMTP (Simple Mail Transfer Protocol)
  - Sending out email from a user
  - Exchanging between mail servers

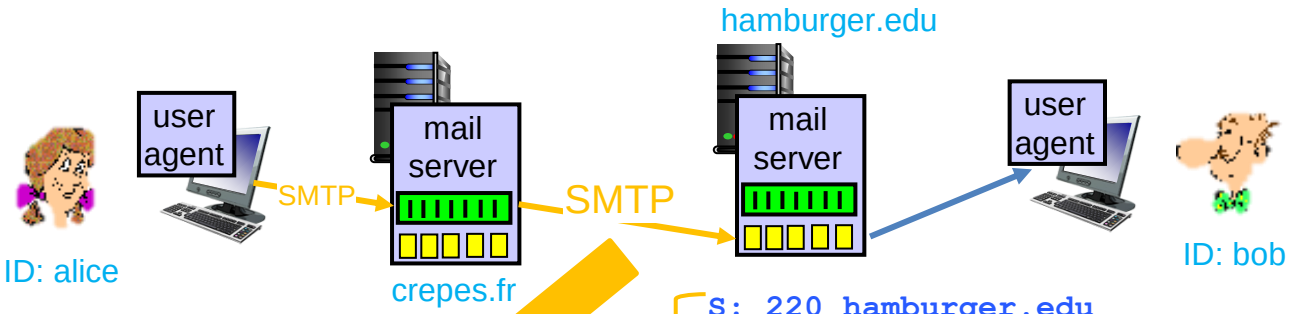


SMTP [RFC 2821]

- Uses TCP as the transport layer protocol for reliable email delivery from sending server to receiving server
- Three phases of transfer
  - handshaking (greeting)
  - transfer of messages
  - closure
- Command/response interaction (like HTTP)
  - commands: ASCII text
  - response: status code and phrase



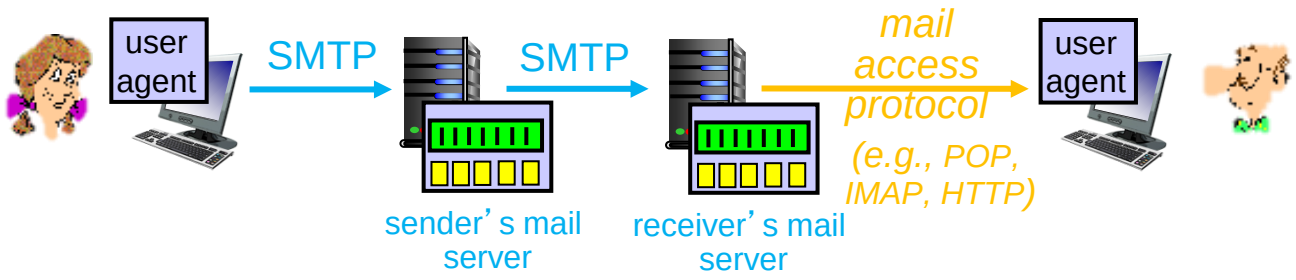
SMTP Example



After the TCP connection is established,

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250  Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Mail Access Protocols



POP3

- Post Office Protocol 3
- By default, deletes messages from the server after retrieving
- Disconnects from the server after download

IMAP

- Internet Mail Access Protocol
- Keeps all messages at server and allows user to organize message folders => synchronization across devices
- Stays connected until the mail client app is closed and downloads messages on demand

HTTP

- Web-based email
- Used between browser and server (user-to-server, server-to-user)
- Hotmail in the mid 1990s
- Google, Yahoo!, etc.

관련 영상: https://www.youtube.com/watch?v=SBaARws0hy4