World Wide Web (WWW)

-collection of web resources and application that gives access to use resources via the internet

-greatest information system

Sir Tim Berners-Lee

-invented WWW in 1989

* CERN (Switzerland)
* Text-based resources

Web resource

* Static-already existed
* Dynamic-produced on the flag

Application

* WWW-client-server
* Web servers
  1. Hosting/Storage (electronic)
  2. Request by the client
* Web clients
  + Browser
  + Spiders-application that crawls the web that gets information. (ex. Google spider)

Internet

-shortcut of Internetwork (inter-across boundaries, network-establishing a connection)

-global, network of networks

* hosts- devices

NETWORKS

* interconnection media
* protocols(standards)

File Transfer Protocol(FTP)-allows to access files

http://webtek.org/index.html

scheme

domain name/host name

Fiddler – web debugging proxy

**HTTP**

* Application layer communications protocol used to access resources (hypertext/hypermedia) on the world wide web
* Invented by Sir Tim Berners – Lee at CERN in 1989
* Jointly developed by the W3C and the IETF

Also founded by Berners-Lee

* Used to retrieve resources

Version History

* HTTP 0.9 (1991) (GET)
* HTTP 1.0 (RFC 1945, May 1996) (GET,HEAD,POST)
* HTTP 1.1(RFC Jan 1997, RFC 2016 Jun 1999), RFC 7230-7235 (Jun 2014) (Persistent Connection)
* HTTP 2 (RFC 7540 May 2015)
  + SPDY (From google)
    - Faster
  + Server Push
    - Lessen page load times

Type Lining

* “send an html while still sending other html without getting response” something like that XD
* “pipe lining”

HTTP Fundamentals

* HTTP typically runs on top of TCP/IP, using TCP port(0-65535) 80 by default, or TCP port 443 for HTTPS(HTTP over SSL/TLS)
* HTTP is based on a client-server architecture
  + Servers
    - Origin servers
    - Proxy servers, gateways, tunnels(encryptions)
  + Clients a.k.a user agent(UA)
    - Web browsers, web crawlers/spiders, other end user tools and applications

SSL-encrypts data

* HTTP uses a request- response standard protocol
  + The client sends an HTTP request message on the server
  + The server processes the request and replies with an HTTP response message
* HTTP is a statistics communication protocol
  + Servers do not keep information about clients in-between requests
* HTTP provides support for other functionalities such as:
  + Cache control
  + Content media type (MULTIPURPOSE INTERNET MAIL EXTENTIONS or MIME) specification
  + Language and character set specification
  + Content/transfer codings – compression(rar,zip)
  + Content negotiation
  + Client-server protocol negotiations – switch protocols
  + Persistent connections
  + Request pipelining/multiplexing
  + Authentication/authorization
  + Etc.

HTTP Resource Addressing

* HTTP resources are identified using URIs(RFC (3986),or more, specifically HTTP URLs)
  + Scheme ( http or https)
  + Authority
    - User information or authentication credentials ( usr:pwd)
    - Host
      * Domain name ( resolved to an IP address using DNS) of the server where the resource resides)
    - Port number
  + Path to resource(resolved relative to the document root on the server)
    - May refer to a static or dynamic resource
  + Query
    - Typically provided as key=value pairs, with ampersand(&) separators between key/value pairs
    - May be URL-encoded
  + Fragment identifier
    - Starts w/ hashtag(#)
    - Bookmark

Example

http://usr:pwd@server.org:81/info/profile.php?id=1234#addr

HTTP Request Message

* Request Line (CRLF-terminated line consisting of three space-separated values)
  + Method
  + Request Target
  + Protocol Version
* Message Headers (general,request,and/or entity headers)
  + HTTP 1.1 requires at least the **HOST** request header to be provided
* Empty Line (CRLF) new line
* Message Body (Optional) a.k.a payload

HTTP Response Message

* Status Line
  + Protocol Version
  + Status Code
  + Reason phrase
* Message Headers (general, response, and/or entity headers)
* Empty Line(CRLF)
* Message Body(CRLF)

HTTP Request Methods

STANDARD METHODS

GET, HEAD, POST, PUT, DELETE, OPTIONS, TRACE, CONNECT

GET (w/ payload)

-most commonly used HTTP method

HEAD (w/o payload)

-identical to GET, except that server gives response yon line and headers