-The Internet & World Wide Web

- WWW Technologies

* HTTP, HTML, CSS, Client side scripting
* Server Side Scripting
* Web Servers, Web Clients
* Web Applications

Technical Documents – HTTP (RFC 2616, RFC 7230-7237)

* DL, Install & Config a local web server
* DL, Install & Config experiment with fiddler

Internet- Worldwide system of computer networks

* Connection of devices
* International Network
* LAN(Local Area Network)

IP- INTERNET PROTOCOL

ISP – INTERNET SERVICE PROVIDER

TCP/IP- TRANSMISSION CONTROL PROTOCOL/ INTERNET PROTOCOL

Nodes- Interconnection Technology (Wired & Wireless); protocol (Standard)

IoT – Internet of things

WWW- World Wide Web

* Information System on the Internet that allows documents to be connected to other documents by hypertext links.
* Play game, send email, share files

Tim-Berners Lee- Father of the web

Info.cern.ch (first website)

Birth of web

* Tim Berners Lee, British scientist at CERN (European Organization for Nuclear Research) invented w3 in 1989, to meet the demand for automatic info- sharing between scientists in universities and institutes around the world.
* Hosted by BL’s next computer
  + Basic features of the web
  + Access other people’s documents
  + Set up your own server

WAIS (Wide Area Information System)

* Internet system in which specialized subject database are created at a multiple server locations

GOPHER (protocol)

* TCP/IP application layer protocol assigned for distributing, searching, and retrieving documents over the Internet

USENET

* Worldwide system distributed discussion system available on computers

HTTP

* HYPERTEXT TRANSFER PROTOCOL
* Foundation of data communication
* Application layer communications protocol used to access resource (hypertext/ hypermedia) on the world wide web
* Developed by the W3C and the IETF

History

* HTTP 0.9: The One-Line Protocol (1991)
* HTTP/1.0: Rapid Growth and Informational RFC (RFC 1945, May 1996)
* HTTP/1.0: Rapid Growth and Informational RFC (RFC 2068 Jan 1997, RFC 2616 June 1999, RFC 7230-7235 June 2014)
* HTTP/2: Improving Transport Performance (RFC 7540 May 2015)

HTTP FUNDAMENTALS

* HTTP runs on top of TCP/IP, TCP port 80 by default or TCP port 443 for HTTPS (HTTP over SSL/TLS)
* HTTP based on a client-server architecture
  + Clients aka user agents (UA)

HTTP uses a request-response standard protocol

* + The client sends an HTTP request protocol

HTTP is a stateless communications protocol

* Servers do not keep ionformation about clients in between requests

HTTP provides support for other functionalities such as:

* Cache controls
* Content media type (MIME) specification
* Language and character set specifications
* Content or transfer codings
* Content negotiation
* Client-server protocol negotiation
* Persistent connections
* Request pipelining
* Authentications or authorizations

HTTP resource addressing

* Resources are identified using URLs (RFC 3986), or more specifically, HTTP URLs
  + Scheme (https or http)
  + Authority
    - User information or authentication credentials (depreciated)
    - Host
      * domain name (resolved to an IP address using DNS of the server using the resource)
    - Port number
  + Path to the resource
    - Resolved relative to the content not the server

DNS – Domain name system

* + - * May refer to a static dynamic resource
  + Query
    - Typically provided as key=value pairs, with & ampersand separator between key/ value pairs
    - May be URL encoded
  + Fragment identifier
    - bookmark

HTTP request messages

* Request line
  + CRLF- terminated line consisting of three space separated values
    - Method
    - Request URL
    - HTTP Protocol version
* Message headers
  + General, request and/or entity headers
    - Status line, status code 200- successful
    - HTTPs 1.1 requires at least the HOST request header to be provided
* Empty line (CRLF)
* Message Body aka Payload(opt)

HTTP response message

* Status line (CRF- Terminated line consisting of three space- separated values)
  + HTTP protocol version
  + Status code
  + Reason phrase
* Message headers (general, response and/or entity headers)
* Empty line (CRLF)
* Message body(opt)

HTTP request methods

* Get
  + Most commonly used
  + Used to request from the server the retrieval of the source identified by the request URI; the retrieved resource is returned in the message body as an entity
* Head
  + The entity doesn’t response
* Post
* Put
  + store the enclosed entity in the message under the specified request URI
* Delete
  + delete the resource identified by the request URI
* Options
* Trace
  + “echo” back to the client the received request
* Connect

HTTP message headers

* General headers
  + Cache-control
  + Connect
  + Date
  + Pragma
  + Trailer
  + Transfer-Encoding
  + Upgrade
  + Via
  + Warning
* Request header field

o Accept

o Accept-charset

o Accept-encoding

o Accept-Language

o Authorization

o Expect

o From

o Host

o If-Match

o If-Modified-Since

o If-None-Match

o If-Range

o If-Unmodified-Since

o Max-forward

o Proxy-Authorization

o Range

o Referrer

o User-Agent

* Response Header Fields

o Accept-Range (bytes, fetch portions)

o Age( age of the resource, last modified/ETag)

o E-Tag

o Location(where to find the redirect)

o Proxy-Authenticate

o Retry-After

o Server

o Vary

o WWW-Authenticate

* Entity Header Fields

o Allow

o Content-encoding

o Content-language

o Content-length

o Content-location

o Content-MD5

o Content-range

o Content-type

o Expires

o Last Modified

HTTP Status Codes

* Informational (1xx)

o 100 Continue

o 101 Switching Protocols

* Success (2xx)

o 200 OK

o 202 Accepted

o 203 Non-Authoritative Information

o 204 No Content

o 205 Reset Content

o 206 Partial Content

* Redirection (3xx)

o 300 Multiple Choices

o 301 Moved Permanently

o 302 Found

o 303 See Other

o 304 Not Modified

o 305 Use Proxy

o 307 Temporary Redirect

* Client Error (4xx)

o 400 Bad Request

o 401 Unauthorized

o 402 Payment Required

o 403 Forbidden

o 404 Not Found

o 405 Method not Allowed

o 406 Not Acceptable

o 407 Proxy Authentication Required

o 408 Request Timeout

o 409 Conflict

o 410 Gone

o 411 Length Required

o 412 Precondition Failed

o 413 Request Entity Too Large

o 414 Request-URI Too Long

o 415 Unsupported Media Type

o 416 Request Range Not Satisfiable

o 417 Expectation Failed

o 426 Upgrade Required

* Server Error (5xx)

o 500 Internal Server Error

o 501 Not Implemented

o 502 Bad Gateway

o 503 Service Unavailable

o 504 Gateway Timeout

o 505 HTTP Version Not Supported