

1) HTTP HEADER FILES :-

http header fields provide required information about the request or response or about the object sent in the message body. There are four types of HTTP message headers.

- * General Header :- These header fields have general applicability for both request and response messages.
- * Client Request Header :- These header fields have applicability only for request messages.
- * Server - Response header :- These header fields have applicability only for response messages.
- * Entity - header :- These header fields define meta information about the entity body or if no body is present about resource identified by the request.

General Headers:-

The header fields are:-

- Cache Control.
- Connection.
- Date.
- Pragma
- Trailer.
- Warning.

Client Request Headers:-

The header files are:-

- Accept.
- Accept Encoding.
- Accept Language.
- Authorization.
- Cookie.
- Host.
- If - Match
- Proxy Authorisation.

Server Response Headers:-

The header files are:-

- Accept-Range.
- Age
- Location.
- Vary.
- WWW-Authenticate.
- Server.
- Set-Cookie.

Entity Headers:-

The header files are:-

- Allow
- Content-Encoding
- Content-Language
- Content-Length
- Content-Location
- Content-MD5
- Expires.

FTP COMMANDS AND REPLIES.

HOST - Identify desired virtual host on server by name.

LANG - Language Negotiation.

EPSV - Enter extended passive mode.

DELE - Delete File.

CCC - Clear Command Channel.

APPE - Append.

ABOR - Abort an active file transfer.

ACCT - Account Information.

ALLO - allocate sufficient disk space to receive a file.

AUTH - Authentication/Security Mechanism.

3) RFCs OF HTTP:-

In June 2014, W3C released an updated six-part specification and RFC 2616 was made obsolete.

RFC 7230 - Released as part of HTTP/1.1. It was concerned with message Syntax and Routing.

RFC 7231 - Released as part of HTTP/1.1. It was concerned with semantics and Content.

RFC 7232 - Released as part of HTTP/1.1. Concerned with Conditional Requests.

RFC 7233 - Released as part of HTTP/1.1 and holds Range Requests.

RFC 7234 - Released with HTTP/1.1 and holds Caching.

RFC 7235 - Released with HTTP/1.1 and Controls Authentication.

RFC 7540 - Published in May 2015 as part of HTTP/2.0.

4) VARIOUS OPTIONS OF TCP:-

There are five major options that are associated with TCP and they are the following.

MAXIMUM SEGMENT SIZE:-

It is used to define the max. segment that will be used during a connection between two hosts. As such, the option is used during the SYN and SYN/ACK phase of the 3-way handshake. The MSS TCP uses ~~4 bits of the~~ 32 bits of length.

WINDOW SCALING:- It is essentially an extension of window size flag. It can be maximum of 30 bits in size. It was created for high latency, high bandwidth WAN links where a limited window size can dent performance.

SELECTIVE ACKNOWLEDGEMENT:- If segments arrive out of order, then a hole is created in the receiver's queue. So a robust ACK was introduced and thus came selective ACK (SACK). It is 216-bit fields. The 2 fields are because the receiver has to specify the range it received.

TIMESTAMPS:- For TCP to accurately set the ~~thru~~ threshold value for a Virtual Circuit, it has to measure the round trip delivery times for various segments. Finally it has to monitor additional segments throughout the connection's lifetime to keep up with the changes in the network. This is where the Timestamp option comes into the picture. It has two fields echo and reply both are 4 bytes long.

NOP :-

The nop TCP means "NO Option" and is used to separate the different options used within the TCP option field. The implementation of the nop field depends on the OS used. NOP occupies 1 byte.

5) DNS USING UDP INSTEAD OF TCP :-

The major reason is performance. TCP connections are very expensive with the whole SYN, SYN-ACK and ACK chain of events. A DNS response can fit within a single packet with lots of room to spare, so using TCP for DNS seems unfruitful. UDP is fast, reasonable, reliable and simple so a stateless protocol like UDP is sufficient for DNS.

6) DOMAIN NAMES :-

The common domains are .com, .in, .edu, .ac.

- .org - organisational domain.
- .info - Informational domain.
- .org.in - organisational domain for India.
- .tv - television domain used by broadcasters.
- .mobi - used for mobile websites.
- .net - originally intended for Network companies.
- .co.in - company domain for India.
- .gov.in - India government's websites common domain.
- .nic.in - National Information Centre's domain.
- .ac - academic organisation domain.

7) NSLOOKUP :-

a) nslookup www.google.com

Server: 127.0.1.1

Address: 127.0.1.1#53

Non-Authoritative Server:

Name: www.google.com

Address: 74.125.200.103

Name: www.google.com

Address: 74.125.200.99.

b) nslookup www.facebook.in.

Server: 127.0.1.1

Address: 127.0.1.1#53.

Non-authoritative Server:

Canonical name: www.facebook.com

Name: star-mini.c10r.facebook.com

Address: 157.240.16.35.

c) nslookup www.flipkart.com.

Server: 127.0.1.1

Address: 127.0.1.1#53.

Canonical Name: flipkart.com

Name: flipkart.com

Address: 163.53.78.128.

d) nslookup www.zomato.com

Server:- 127.0.1.1.

Address:- 127.0.1.1#53.

Non-Authoritative Server:

Name = zomato.com.edgekey.net.

Canonical Name = e1156b.b.akamaiedge.net.

Name: e1156b.b.akamaiedge.net.

Address: 104.122.5.136.

e) nslookup www.gmail.com

Server: 127.0.1.1

Address: 127.0.1.1#53.

Non-Authoritative Server: www.gmail.com

Canonical name = www.gmail.com.

Name = googlemail.l.google.com.

Address: 216.58.197.69.

ANALYSIS OF FIVE WEBSITES.

WEB APP	BANDWIDTH SENSITIVITY	TIME SENSITIVITY	LOSS TOLERABILITY	PROTOCOL
Amazon	Moderate. This is due to the images and voluminous data that is displayed	loading time is pretty less as no heavy files are involved.	loss tolerant.	TCP.
Youtube	Requires a lot of Bandwidth owing to the large Video files	loading varies with speed of Network.	loss Intolerant.	TCP.
Gmail.	Bandwidth insensitive	Time insensitive	loss Intolerant	TCP/SMTP.
Dropbox	Bandwidth sensitive	Time insensitive	loss Intolerant	TCP/FTP.
Facebook	Bandwidth sensitive	Time insensitive	loss Tolerant	TCP.