Computer Networking Notes 🔥

By Garvit Singh, IT Department

Protocols

Protocols are nothing but how data is transferred in a computer network.

- **TCP/IP**(Transmission Control Protocol/Internet Protocol) - Ensures data will reach its destination and will not be corrupted in the way. Connection-oriented protocol.
- **UDP**(User Datagram Protocol) - Useful in cases where you do not care if 100% data is reaching the destination or not. Stateless and Connection-less protocol. Data may be lost. Used widely in Gaming, Video Conferencing etc.
- **HTTP**(Hyper Text Transfer Protocol) - Used by World Wide Web(WWW) and web browsers.
- **HTTPS**(Hyper Text Transfer Protocol Secure) - Extension of HTTP. Used for secure communication over a computer network. Uses HSTS and TLS for encryption and authentication.
- **DHCP**(Dynamic Host Configuration Protocol) - Network management protocol used on Internet Protocol networks for automatically assigning IP addresses and other communication parameters to devices connected to the network using a client–server architecture.
- **SMTP**(Simple Mail Transfer Protocol) - Send and distribute outgoing emails.
- **POP3 & IMAC** - Receive emails.
- **SSH** - Login to a terminal of someone else's computer.

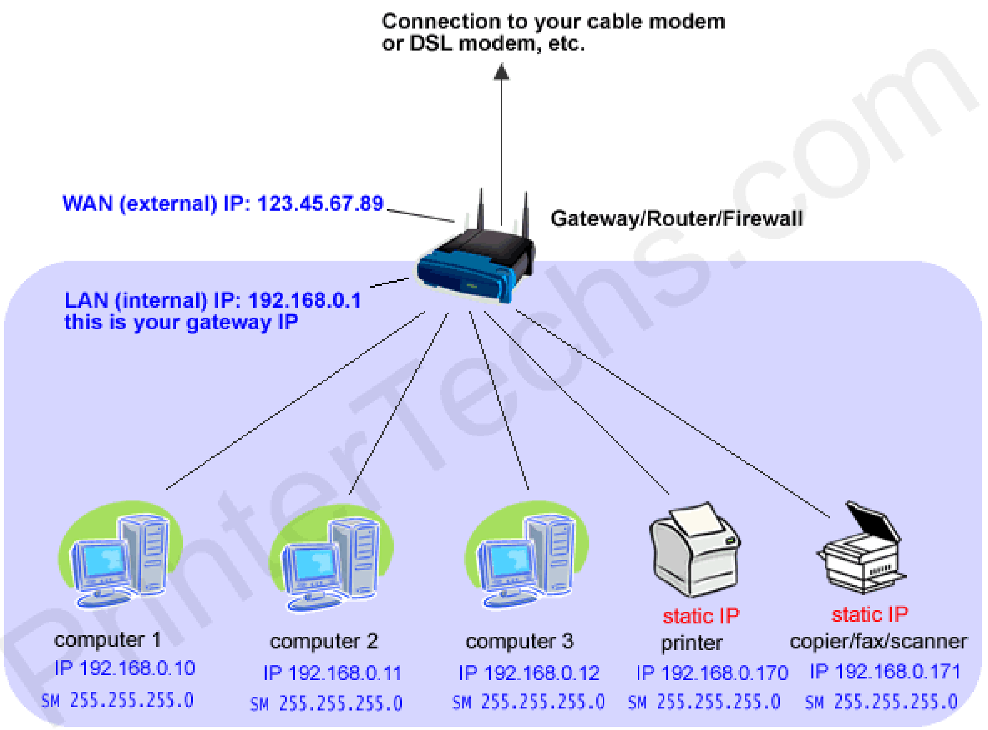
- **PPP**(Point To Point Protocol) - Create a direct connection between two communicating devices. It defines rules using which two devices will authenticate with each other and exchange information.
- **FTP**(File Transfer Protocol) - Transfer files from one system to another. Works on client-server model. When a machine requests for file transfer from another machine, FTP sets up a connection between the two and authenticates each other using ID and password.
- **SFTP**(Secure File Transfer Protocol) - Encrypts both commands and data while in transmission. Encrypts files and sends them over a shell data stream.
- **TELENET**(Terminal Network) - Used for virtual terminal service given by ISO. This enables one local machine to connect with another. The computer being connected is the remote computer and which is connecting is the local computer. TELENET operation lets us display anything being performed on the remote computer in the local computer.

Basic Concepts

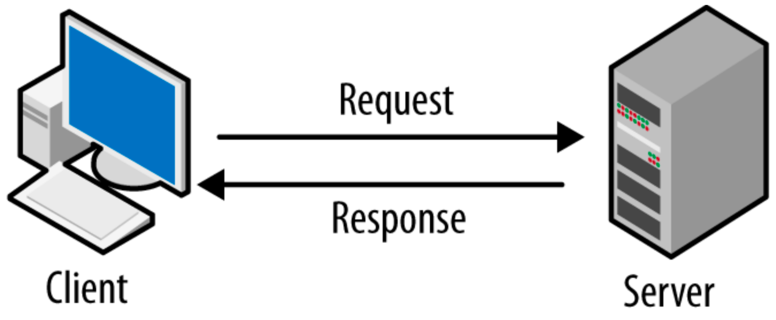
- Data is transferred in the form of Packets.
- IPv4 Addresses : X - X - X - X where X ranges from 0 to 255. Denotes the location of your computer. To check IP Address of your system, use command :

```
hostname -i | awk '{print $1}'
```

- Modem/Router converts analog signals into digital signals and vice versa. Routers have a global IP address and assign unique IP Addresses to each device connected to it.



- Thread is a lighter version of a process. One process can have multiple running threads.
- Client - Server Architecture



Port Numbers

- Port Numbers are 16 Bit Numbers. Denotes which application is to be communicated with in a computer. Denotes which application we are working with.
- Total Port Number possible are 2^16 ~ 65,000.
- Reserved Ports - 0 to 1023
- Reserved for certain types of applications - 1024 to 49152

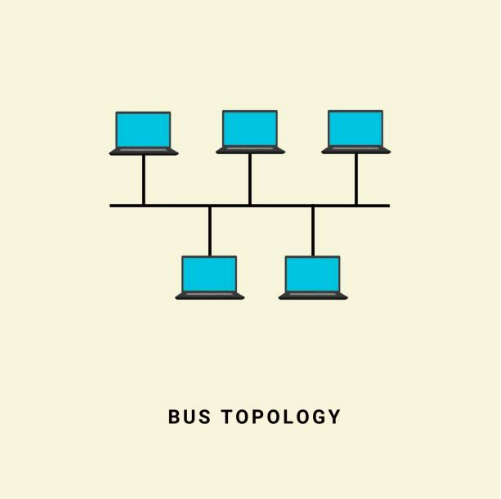
- Remaining ones can be used by you.
- Ephemeral Ports - Application internally assigns itself random port numbers. Used internally within the computer. Ex - mutiple tabs in chrome will have different port numbers assigned to them by the browser.
- Sockets - Interface between a process and the internet. When you need to send messages from one system to another, sockets can be used.

Data Speeds

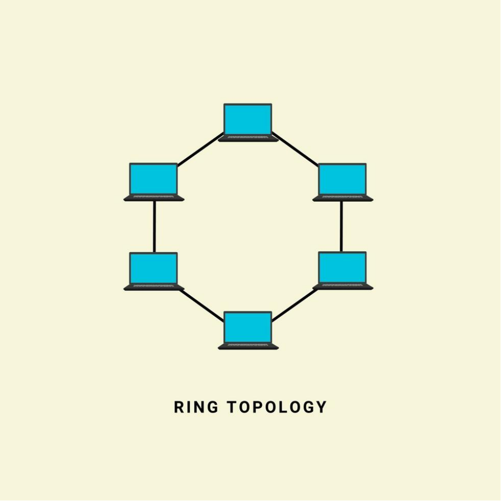
- 1 mbps = 1,000,000 bits/second
- 1 gbps = 10^9 bits/second
- 1 kbps = 1000 bits/second

Topologies

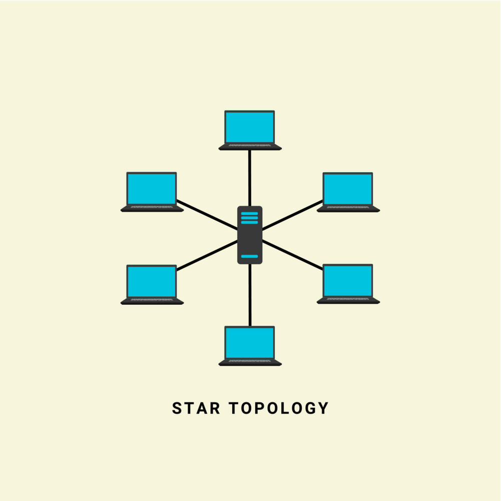
- Physical Connection - optical fibre cables, coaxial cables etc.
- Wirless Connection - Bluetooth, Wifi, 4G, 5G etc.
- Local Area Network(LAN) - Small House, Office, through Ethernet or Wifi.
- Metropolitan Area Network(MAN) - Across a city through phone towers.
- Wide Area Network(WAN) - Across countries through optical fibre cables
    - SONET - Through optical fibre cables, large distances.
    - Frame Relay - Connect LAN to WAN.
- Bus Topology



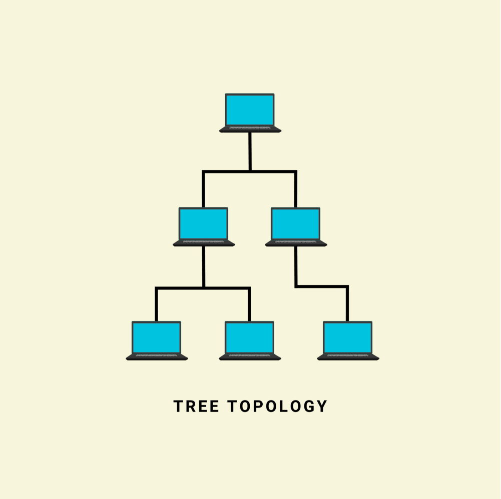
- Ring Topology



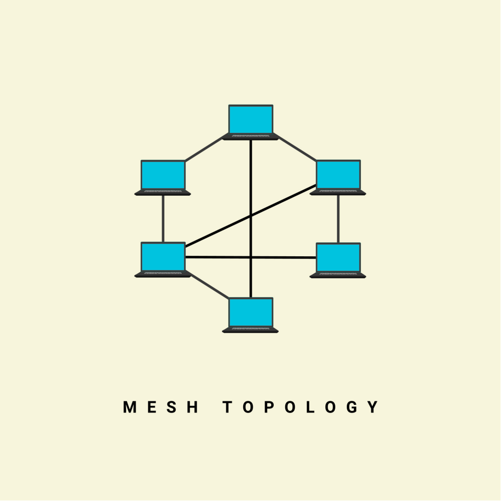
- Star Topology



- Tree Topology

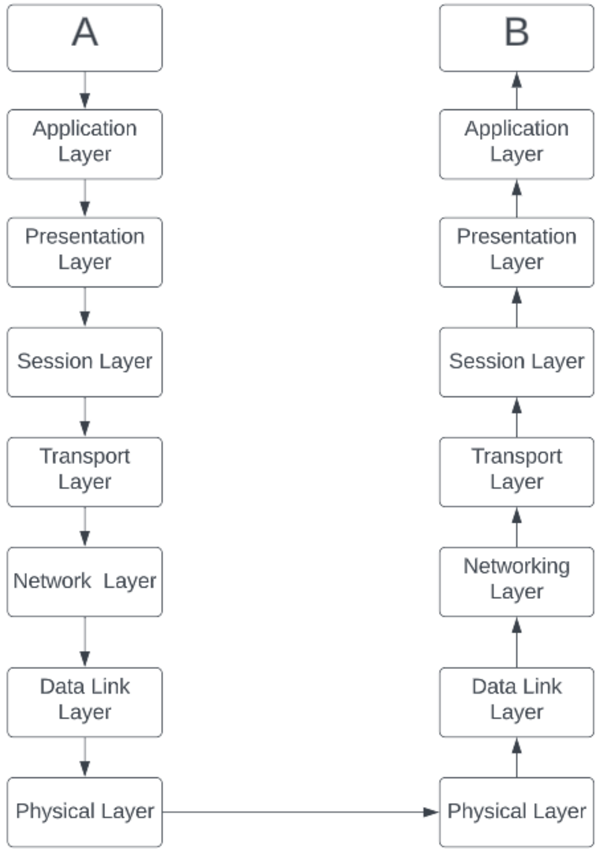


- Mesh Topology



Structure Of The Network

Open Systems Interconnection Model (OSI Model)



7 Layers of OSI Model :

1. Application Layer
2. Presentation Layer
3. Session Layer
4. Transport Layer
5. Network Layer
6. Data Link Layer

7. Physical Layer

Layers 1 to 4 are present inside your computer, while Layers 5 to 7 are present outside.

OSI Model : Each Layer In Detail

1. **Application Layer**

- Softwares used by end-users, Front-end, Graphical User Interface(GUI). Users interact directly with this layer.

2. **Presentation Layer**

- Receives data from the application layer.
- Data from the application layer could be in the form of words, characters, numbers etc.
- Data is converted into machine representable binary format in a process known as translation.
- Before this data is transmitted further, it is encoded and encrypted by using various cryptographic algorithms so that privacy of the sender and receiver is not compromised. It provides abstraction to the data.
- This layer assumes that the data will be taken care of after sending it downwards to the next layer.
- The data is compressed to make it easy to transfer.

3. **Session Layer**

- Helps in setting up and managing connections.
- Enables sending and receiving of data followed by termination of the connected sessions.
- Does authentication like username or password before a session is established, followed by authorisation to access the file or data.

- Assumes that layers downward in the network will do their job.
- Ex - When you open Amazon website, a session is being created between your computer and Amazon's servers. After you're done, it logs you out of the session.

4. **Transport Layer**

- Has protocols like UDP and TCP.
- Works with data to make sure it is transported easily. It does this in three ways.
- One part is known as Segmentation where data received from session layer will be divided into small data units called segments.
- Every segment will contain the source and destination port numbers and a sequence number.
- Sequence helps to reassemble the segments in a proper order.
- Works on flow control. It controls the amount of data being transferred per second.
- Works on error control if data packets get lost or corrupted.
- Adds a checksum to every data segment for error control purposes.

5. **Network Layer**

- Works for transmission of received data segments from one computer to another that is located in a different network.
- Wifi routers are situated in this layer
- Performs logical addressing. IP Addressing is known as logical addressing

- Network layer assigns the sender and receiver's IP Address to every segment and it forms an IP data packet so that every data packet reaches its correct destination.
- Performs routing which means moving one data packet from source to destination.
- Load balancing also happens here to make sure data traffic is not overloaded.

6. **Data Link Layer**

- Directly communicates with the computer and hosts.
- Receives data packets from network layer and it contains IP Addresses of both sender and receiver.
- After logical addressing done in network layer, the data link layer performs the physical addressing(MAC Addresses) which denote the application the data has to be sent to.
- MAC Addresses of sender and receiver are assigned to the data packets to form a Frame.
- Frame is the data unit of data link layer.
- MAC Address is a 12 digit alphanumeric number of the network interface of the computer.
- A device's bluetooth, wifi etc all have different MAC Addresses.
- Data link layer has 2 functions, it allows upper layers to access these frames.
- It also controls how the data is placed and received from the medias using Media Access Control which are techniques used to get frames on and off media like error detection.

7. **Physical Layer**

- Hardware section like the wires, router, CPU, Memory, RAM etc.

TCP/IP Model

Internet Protocol Suite, developed by ARPA. TCP/IP Model has more practical real-life applications, unlike OSI Model, which is more theoretical.

5 Layer Of TCP/IP Model :

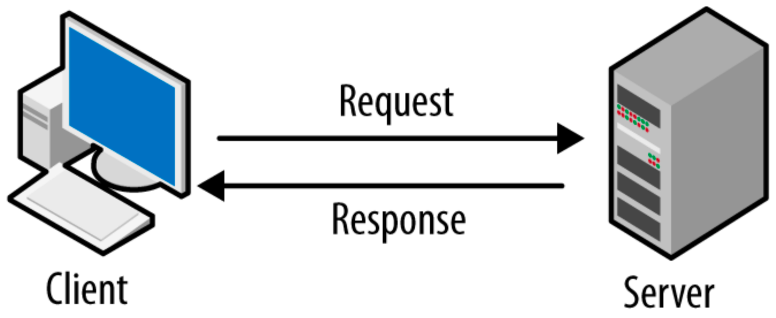
1. Application Layer
2. Transport Layer
3. Network Layer
4. Data Link Layer
5. Physical Layer

TCP/IP Model : Each Layer In Detail

1. Application Layer

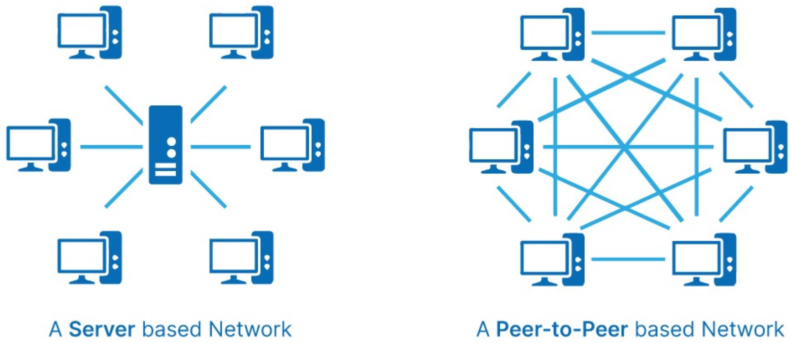
Users interact directly with this layer. Ex - mobile apps, browsers etc.

Client - Server Architecture



- Collection of Servers is called a Data Centre.

Peer To Peer Architecture



- There is no one dedicated server. Very easily scalable. Decentralised network.

HTTP

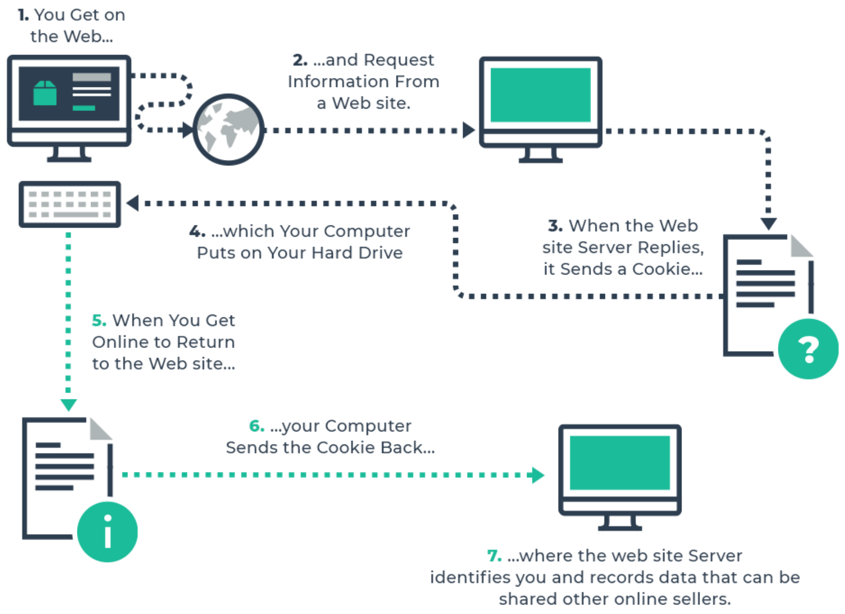
- A stateless application layer protocol. Application layer protocols also require a transport layer protocol. Ex - HTTP methods like GET, POST, PUT, DELETE etc.
- Uses TCP/IP Protocol as its transport layer protocol.

Error/Status Codes

- Know wether a request was successful or it failed.
- Ex - 200 means successful, 404 means couldn't find it, 500 for internal server error.
- 1XX - Informational
- 2XX - Success
- 3XX - Redirecting
- 4XX - Client Side Error

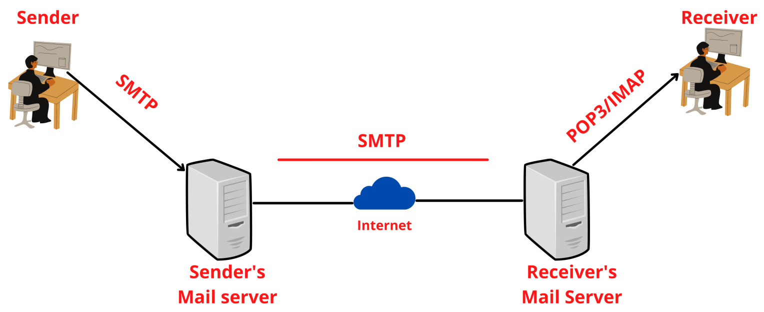
- 5XX - Server Side Error

Cookies


1. You Get on the Web...
2. ...and Request Information From a Web site.
3. When the Web site Server Replies, it Sends a Cookie...
4. ...which Your Computer Puts on Your Hard Drive
5. When You Get Online to Return to the Web site...
6. ...your Computer Sends the Cookie Back...
?
i
7. ...where the web site Server identifies you and records data that can be shared other online sellers.

- Unique string stored on the client's browser.
- When you visit an app or website for the first time, it will set up a cookie.
- Whenever you make a new request, the browser will set a cookie on the request's header.
- This way the server will know from that cookie which data is to be sent and who is contacting the server.
- Third Party Cookies - Cookies that are set for URLs you don't visit. Ex - Websites displaying ads through third party cookies.

How Email Works?



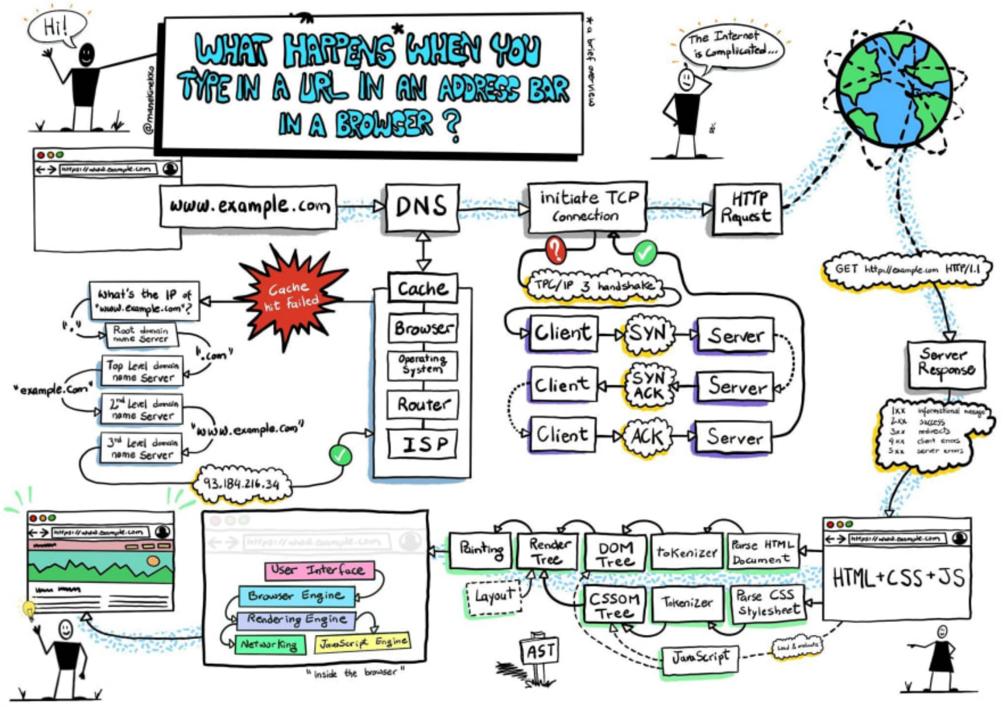
- SMTP, POP3 - Application Layer Protocols.
- TCP/IP - Transport Layer Protocol.
- POP uses port number 110.
- IMAP - Allows viewing emails on mutiple devices. Emails are stored in a server unless you delete them. Local copies of it are available on devices.

Domain Name System(DNS)

- Database service used to store IP Addresses for all the domain names which are accessed by HTTP protocol
- When you type in URLs, HTTP Protocol converts that domain name to its corresponding IP Address using DNS and connects you to the server.
- icann.org manages DNS.
- Ex - mail.google.com. Here, mail is the sub-domain, google is the second level domain, com is the top level domain.

- Top level domains are stored in Root DNS Servers as they are the first point of contact when you write a domain name.
- You cannot buy a domain name, you can only rent them.

What happens when you type a URL? Step by Step Process



1. Check in own computer if the IP Address is already stored in cache memory or in cookies. Browsers store the IP Addresses, which are visited for the first time, so that they can be accessed readily if needed.
2. If not found in cache, then look in the Local DNS Server. It is the first point of contact. Many times, your ISP becomes the local DNS Server. ISPs usually have information of all the websites you are using.
3. If not found in Local DNS Servers, then it looks in Root DNS Servers.

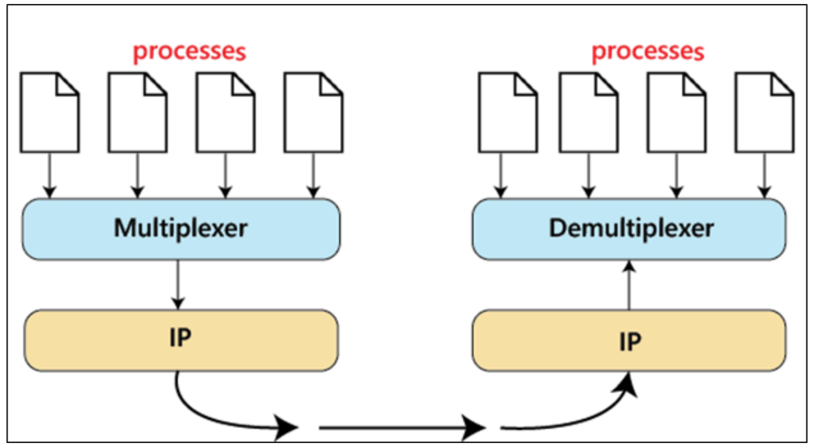
This was all about the Application Layer of TCP/IP Model. Now, moving onto the Transport Layer.

2. Transport Layer

Difference between Transport & Network Layer

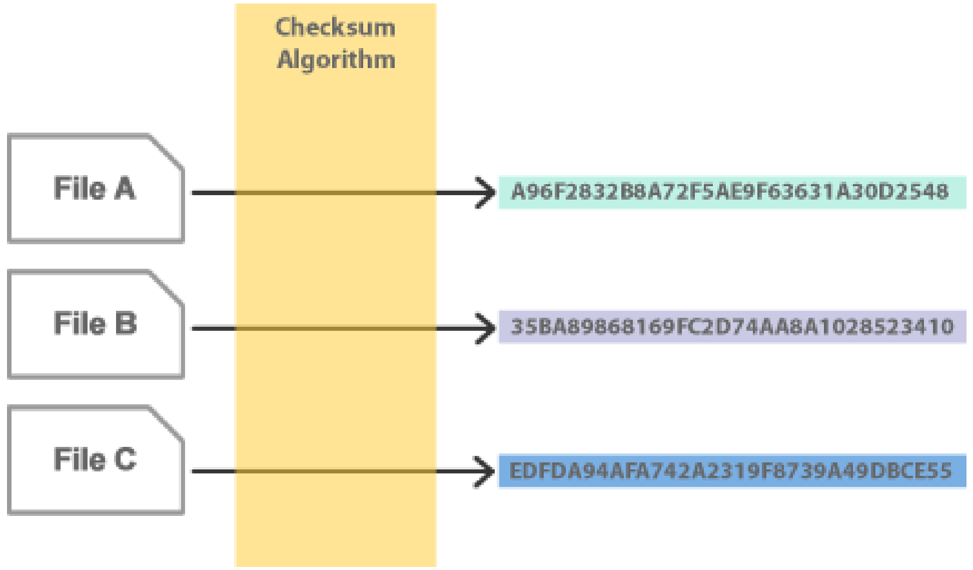
- Transport Layer - Lies inside the device. Its job is to take data from the network to the application and vice versa.
- Network Layer - Lies outside the device. Transportation from network to network is done here.
- Network layer takes care of delivering the message from one computer to another but when the message is reached on the computer, the transport layer delivers it to the applications.
- Important protocols in the Transport Layer are Transmission Control Protocol(TCP) and User Datagram Protocol(UDP).
- Transport Layer takes care of congestion control
- Congestion control algorithms are inbuilt in TCP.
- Data is in the form of Segments in Transport Layer, while data is in the form of Packets in Network Layer.
- Transport Layer has TCP & UDP. Network Layer has IP.

Multiplexing & De-Multiplexing



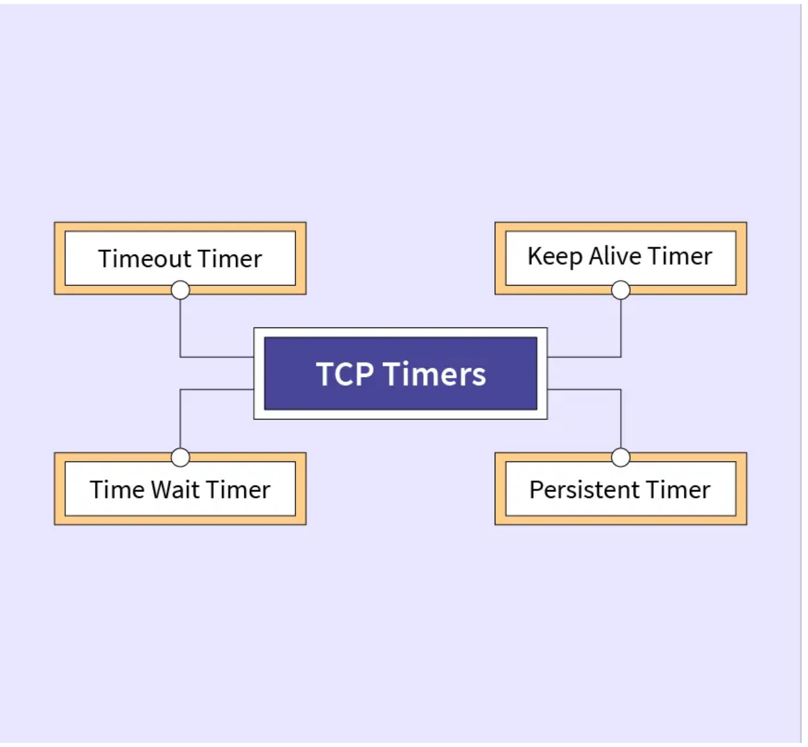
- Sockets have port numbers.
- Data Travels in packets. Transport layer attaches socket port numbers on these data packets, so that they reach the correct applications in the device.

Checksums


Checksum Algorithm
File A
A96F2832B8A72F5AE9F63631A30D2548
File B
35BA89868169FC2D74AA8A1028523410
File C
EDFDA94AFA742A2319F8739A49DBCE55

- When data is sent, you will calculate a checksum for your data using an algorithm.
- When the receiver receives the data, that computer also calculates a checksum using the same algorithm used by sender.
- If checksum values match, then all data has been received without corruption.
- If checksum values don't match, then some data or files are lost.

Timers



There are 4 types of Timers in TCP :

1. **Retransmission Timeout (RTO) Timer:**
    - **Purpose:** The RTO timer is used to manage retransmissions of data segments that are not acknowledged within a certain time frame.
    - **Function:** When a TCP sender transmits a segment and does not receive an acknowledgment within a specified time (RTO), it assumes that the segment was lost or corrupted and retransmits it.
    - **Adjustment:** The RTO timer dynamically adjusts based on network conditions, such as round-trip time (RTT) and variance.
2. **Persistence Timer:**

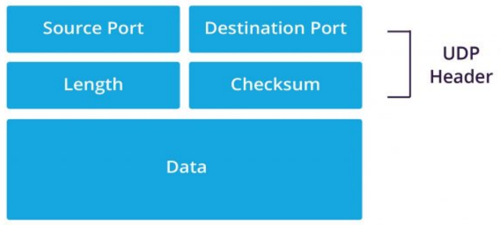
- **Purpose:** The persistence timer is used to manage situations where a sender has data to transmit, but the window size of the receiver is zero (i.e., the receiver's buffer is full).
- **Function:** When the sender encounters a zero window condition, it sets the persistence timer. Upon expiration, it sends a small probe segment to the receiver, allowing it to update its window size.

3. **Keepalive Timer:**
    - **Purpose:** The keepalive timer is employed to detect if a connection is still active in the absence of data traffic.
    - **Function:** After a certain period of inactivity, the TCP stack can send a small keepalive segment to the peer. If no response is received, the connection may be considered idle or broken.
    - **Usage:** Keepalive is often used in applications where long-lived connections need to be maintained, such as those used for file transfers or remote login sessions.
4. **Time Wait Timer:**
    - **Purpose:** The time wait timer is associated with the TIME_WAIT state, which occurs after a TCP connection is closed.
    - **Function:** It ensures that any delayed segments related to the closed connection are not misinterpreted as belonging to a new connection. The TIME_WAIT state allows the network to clear any lingering segments.
    - **Duration:** The time wait timer typically lasts twice the maximum segment lifetime (2MSL), where MSL is the maximum time a segment can exist in the network.

User Datagram Protocol(UDP)

- Data may or may not be delivered.
- Data may change on the way.
- Data may not be in order.
- It is a connection-less protocol.
- UDP uses checksums. It will be aware if the data has been corrupted or lost but it will not do anything about it.

UDP Packets



- Total size = 2^16 bytes - 8 bytes for the header = 65,536 bytes.
- 65,536 bytes is the amount of data you can send in one UDP Packet.
- UDP Packet contains a header which is 8 bytes in size.
- Header contains :
  - Source Port Number(2 bytes)
  - Destination Port Number(2 bytes)
  - Length of Datagram/Length of Packet(2 bytes)
  - Checksum(2 bytes).

Use Cases

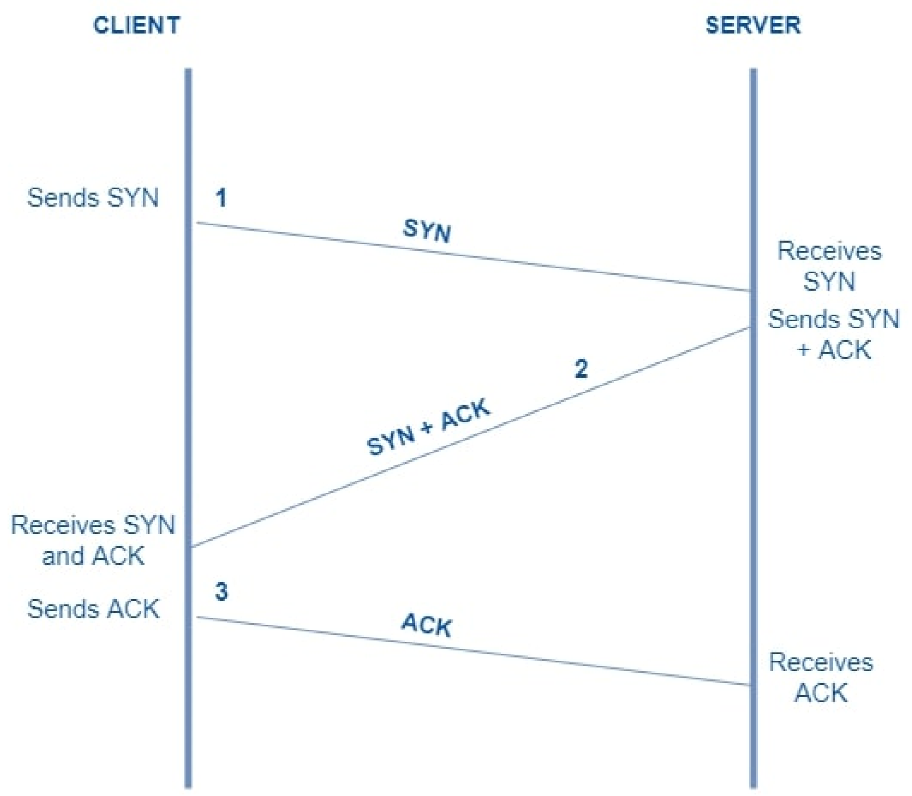
- Very fast.
- Used in gaming, video conferencing apps.
- DNS also uses UDP because it is faster.

Transmission Control Protocol(TCP)

- Transport layer protocol.
- Application Layer sends a lot of data. TCP segments this data in a process called segmentation
- Divide data in chunks, add headers, add checksums.
- It may also collect the data from network layer.
- Congestion Control
- Takes care of :
    - When data doesn't arrive.
    - Rearranges or maintains the order of data.

Features of TCP

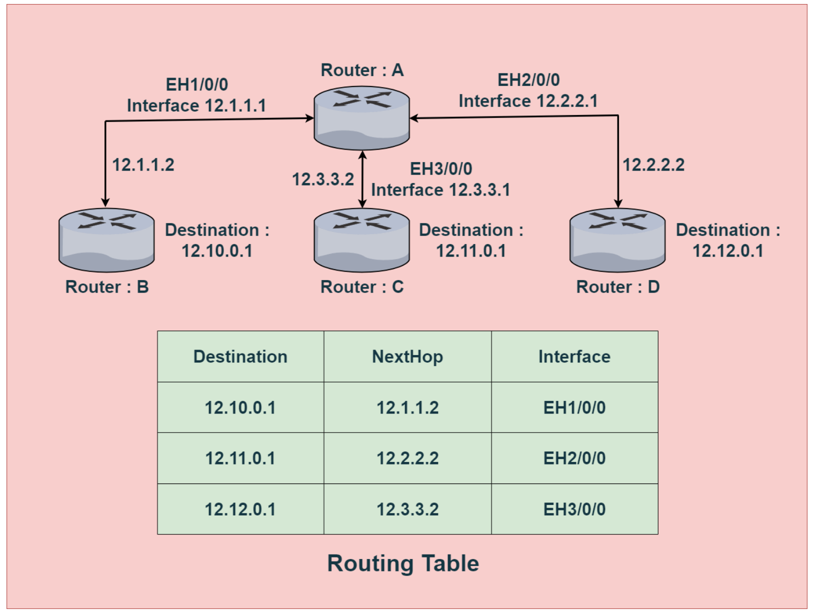
- Connection Oriented
- Error Control
- Congestion Control
- Full Duplex - Send files from both directions simultaneously.
- One TCP Connection has only 2 computers, not more than that.
- 3-Way Handshake - Sequence numbers are random because otherwise they would be very easy to guess for hackers to create a connection with the server. It is for security purposes.



This was all about the Transport Layer of TCP/IP Model. Now, moving onto the Network Layer.

3. Network Layer

- Routers are located in this layer.
- Every router has a Network Address.
- Router checks in its routing table that consists of every destination address.
- Hop by Hop forwarding means hopping router to router until it reaches the correct router.
- Routing table also has a forwarding table inside.



Destination	NextHop	Interface
12.10.0.1	12.1.1.2	EH1/0/0
12.11.0.1	12.2.2.2	EH2/0/0
12.12.0.1	12.3.3.2	EH3/0/0

Routing Table

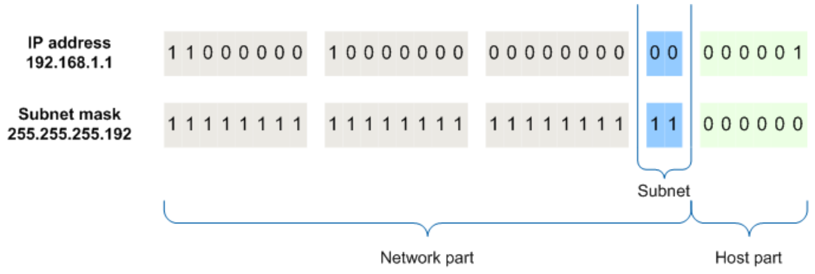
- Ex - 192.168.2.30. Here, 192.168.2 is the Network Address Subnet ID and 30 is the device Address Host ID.
- Control Plane - It creates the database of routing and forwarding tables.
- Router -> Nodes
- Links -> Edges
- Static Routing - Add the route and address manually. Time consuming.
- Dynamic Routing - Evolves with the change in network. They use pathfinding algorithms.

Internet Protocol (IP)

- IPv4 -> 32 bits, 4 words
- IPv6 -> 128 bits, alphanumeric
- Classes of IPv4 Addresses
    - Class A -> 0.0.0.0 - 127.255.255.255
    - Class B -> 128.0.0.0 - 191.255.255.255
    - Class C -> 192.0.0.0 - 223.255.255.255
    - Class D -> 224.0.0.0 - 239.255.255.255
    - Class E -> 240.0.0.0 - 255.255.255.255
- Maintained by Internet Engineering Task Force - ietf.org
- IETF assigns addresses based on regions instead of classes because it is more efficient.

Subnet Masking

- Mask the network part of IP Addresses and it leaves for us to use the host part.

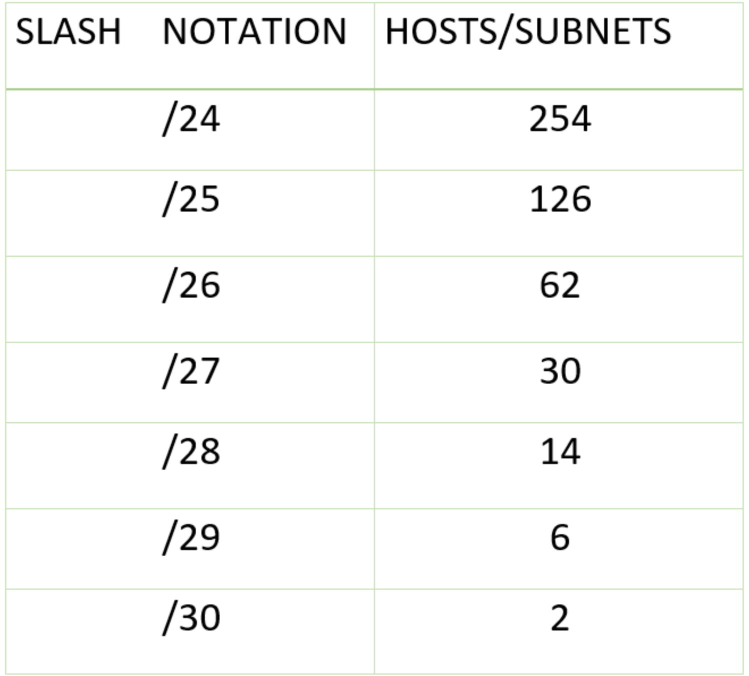


IP address **192.168.1.1**	1 1 0 0 0 0 0 0	1 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0	0 0 0 0 0 1
Subnet mask **255.255.255.192**	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1	0 0 0 0 0 0

Variable Length Subnets

- 12.0.0.0 / 31 -> First 31 bits are part of subnet. Remains 1 bit.

- 192.0.1.0 / 24 -> First 24 bits occupied by subnet. Remains 8 bits.
- 192.0.1.0 - 192.0.1.255 -> Subnet size = 256.



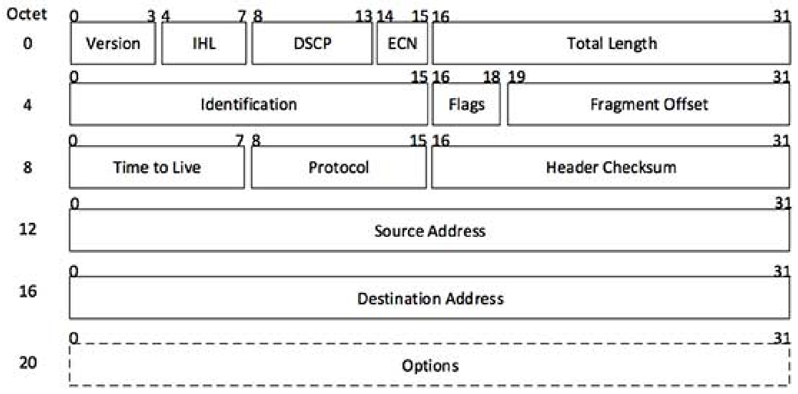
SLASH NOTATION	HOSTS/SUBNETS
/24	254
/25	126
/26	62
/27	30
/28	14
/29	6
/30	2

Reserved Addresses

- 127.0.0.0 / 8 -> First 8 bits are reserved.
- Ex - Local Host 127.0.0.1, Loopback Addresses etc.

IP Packets

- Header -> 20 bytes.
- Contains IPv, Length, Identification, Flag, Protocols, Checksums, Addresses, Time to Live(TTL) etc.



[Image: IP Header]

IPv6 Addresses

- IPv4 -> 2^32 ~ 4.3 billion unique addresses.
- IPv6 is 4 times larger than IPv4.
- IPv6 -> 2^(32X4) = 2^128 ~ 3.4 X 10^38 unique addresses.
- Cons :
  - Not backward compatible.
  - Requires lot of efforts to shift to IPv6 from IPv4 because so many devices would have to shift. Requires lot of hardware changes.
- An IPv6 address typically looks like :
  - A : A : A : A : A : A : A : A, where A is a 16 bit hexadecimal value.
  - Ex - ABFE : F001 : 3210 : 9182 : 0 : 0 : 1 : 3
- Subnet masking like IPv4 can be done with IPv6 as well.
  - Ex - ABFE : F001 : 3210 : 9182 : : / 60
- Ways of representation :
  - 0000 -> 0 (single zero for mutiple ones).

- 1 : 0000 : 0000 : 0000 : 9 ~ 1 : 0 : 0 : 0 : 9 ~ 1 : : 9
- ': :' signifies 'full of zeroes in between'.

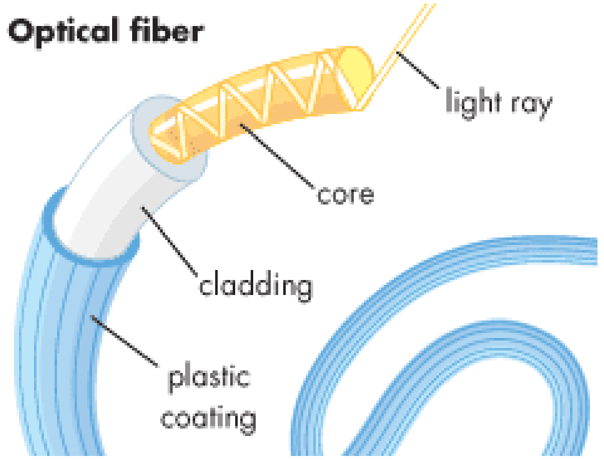
Middleboxes

1. Firewalls for global network, your trusted network.
    - Filters out IP Addresses based on various rules.
    - Addresses
    - Modify Packets
    - Port Numbers
    - Flags
    - Protocols
    - Stateless vs Stateful Firewalls
    - Stateful firewalls are more efficient as they use cache memory.
    - Firewalls are present in network layer as well as transport layer.
2. Network Address Translation(NAT)

4. Data Link Layer

- Packets received from Network Layer.
- Data Link Layer is responsible for sending it over a physical link.
- Converts data packets into Frames.
- Data flows in the form of Frames in Data Link Layer.
- Contains protocols like the Dynamic Host Configuration Protocol(DHCP).
- Each device connected to a network is assigned a data link layer address, which is called the MAC Address.
- Lets say a new device connects to a LAN, now the DHCP server will assign an IP Address to the new device from its pool of IP Addresses.
- Each Frame contains the Data Link Layer Address(MAC Address) of Sender and the IP Address of destination.
- Also contains ARP Cache. ARP stands for Address Resolution Protocol.
- These frames are finally sent over to the physical layer.

5. Physical Layer



- Hardware section like the Wires, Optical Fibre Cables, Coaxial Cables, Routers, CPU, Memory, RAM etc.

Thanks For Reading! 💙

By GARVIT SINGH

Information Technology