Article 1 – What is the Internet and How does it Work?

The Internet is a large “network of networks” that interconnects millions of people across the globe. From watching videos of your favorite YouTuber to sending an email to your professor, all is done via the Internet.

**The Foundation: The Physical Layer**

In essence, the Internet is a vast mesh of wires and cables that carry data between devices:

* Undersea and Terrestrial Cables: Fiber-optics carry data across countries and continents at the speed of light.
* Data Centers and Servers: Large facilities house servers, specialized computers that are built to store and serve websites, applications, and other data.
* Network Devices: Routers and similar devices ensure the correct addressing of data for efficient means of transport.

**The Process: Packet Switching**

Data is transferred via the Internet, not through a continuous stream, but rather in millions of fundamental units known as Packets:

1. Fragmentation: A large file, such as an HD movie, is broken down into blocks of data. Each packet contains a portion of the data and its addressing information.
2. Routing: Each packet travels independently through the network. Routers are used to inspect their addresses and send them along the optimal path.
3. Reassembly: Once all the packets arrive, the receiving device reassembles them into the original file. It compensates for the loss of data by requesting retransmission.

**A Framework: The OSI Model**

The Open Systems Interconnection (OSI) is a theoretical framework that explains how transfer of data transfer takes place via a network place in the Internet. It consists of 7 layers:

1. Physical: Defines the electrical and mechanical characteristics of transmitting raw bits over physical media such as cables.
2. Data Link: Packages raw packets into frames, provides error detection and correction of addressing, and controls the flow of data across shared media.
3. Network: Moves packets from the source host to the destination host, even if on different networks, performs path determination (routing), and packet forwarding to the next appropriate network until the destination host is reached.
4. Transport: Describes the protocol to be used for transmission of data, including fragmentation/reassembly, flow control, and reliability (e.g: TCP vs UDP)
5. Session: Manages dialogs (sessions) between applications – establishing, maintaining, and removing connections, and handling synchronization.
6. Presentation: Translates, encrypts/decrypts, and compresses data for the application layer.
7. Application: The layer closest to end-users provides network services directly to applications, such as web browsing, streaming online, and file transferring.

**Protocols: The Rules for Communication**

Protocols define the rules of communication between devices.

Hypertext Transfer Protocol (HTTP): Text that contains links to reveal new texts are known as Hyperlinks. The protocol used to transfer Hyperlinks is known as HTTP. It governs how web browsers request pages from servers.

Properties of HTTP

1. Human Readable: HTTP is always in a human-readable format
2. Stateless Protocol: You are a new user every time you visit a page

Session: A stored state between client and server. This is done by transferring cookies (data).

The actual/primary data transferred is known as the Payload. HTTP Headers are used to transfer additional information such as the cookies, name of the browsers, etc.

Transport Layer Security (TLS): A cryptographic protocol that secures communication by encrypting data exchanged between the client and the server.

HTTP + TLS = HTTPS

**TCP v/s UDP**

1) Transmission Control Protocol (TCP): A connection-oriented protocol that works by establishing a three-way handshake (SYN 🡪 SYN-ACK 🡪 ACK). Reliable and Ordered.

SYN: Client requests to establish the connection

SYN-ACK: Server acknowledges the request and agrees.

ACK: Client confirms to the server that its response has been received

Use Case: Any application needing guaranteed delivery, such as Web Browsing (HTTP/HTTPS) and File Transfer (FTP)

2) User Datagram Protocol (UDP): A connectionless protocol that does not perform a handshake.

Unreliable and Unordered: No guarantee of delivery or ordering. If data is lost or out of order, the application must handle it.

Use Case: For time-sensitive scenarios where occasional data loss is acceptable to minimize delay, such as live broadcasts, phone calls, and gaming.

**Domain Name System**

The DNS is the phonebook of the Internet. It translates human-friendly domain names into numerical IP addresses that computers use to uniquely identify themselves.

The DNS process is as follows:

1. User Request: The client requests the IP address.
2. DNS Resolver: The Internet Service Provider (ISP) uses a DNS Resolver as the middleman to process the request.
3. Root DNS Server: The resolver asks the Root DNS server where to find the Top-Level Domain (.com, .org, .in)
4. Authoritative DNS Server: Once the TLD has been found, the Resolver asks the Root DNS for the Authoritative DNS Server of the IP Address ( such as GoDaddy)
5. IP Address: Finally, the Authoritative DNS Server provides the destination IP Address to the resolver, and in turn, back to the client

**DNS Records**

DNS Records are entries in a DNS file that tell DNS resolvers how to handle queries for the domain. In essence, they’re structured pieces of data that map human-readable domain names to the technical information needed to route to them.

A Address Record: Maps a domain name to an IPv4 address.

example.com in A 🡪 93.184.216.34

AAAA Address Record: Maps a domain name to an IPv6 address.

example.com in AAAA 🡪 2606:2800:220:1:248:1893:25c8:1946

CNAME (Canonical Name) Record: Creates an alias from one name to another

www.example.com in CNAME 🡪 example.com

**The World Wide Web (WWW)**

The World Wide Web (or the Web) is an information-sharing system built on top of the Internet. It is what transformed the Internet from a collection of isolated data packages to a truly interconnected, user-friendly medium.

* Uniform Resource Locator (URL): A standardized address that specifies the location of a resource as well as the protocol used to access it.
* HyperText Markup Language (HTML): The language used to structure web content (texts, images, videos) on web pages.
* HyperText Transfer Protocol (HTTP/HTTPS): The application-layer protocol that defines how browsers (clients) request information and how servers respond.
* Web Browsers: Client applications such as Chrome and Safari that interpret HTML/CSS/JS files to render pages and enable user interaction.