# Chapter 26 Fundamentals of Computer Networks (Part 1)

## Content



## Syllabus Learning Outcomes

4.1 **Fundamentals of Computer Networks**

Understand computer network technology

- 4.1.1 Explain the concepts of LAN, WAN, intranet, and the structure of the internet.
- 4.1.2 Understand the concepts of IP addressing and domain name server (DNS).
- 4.1.3 Explain the need for communication protocols in a network.
- 4.1.4 Explain how data is transmitted in a packet-switching network.

# 1 Computer Network

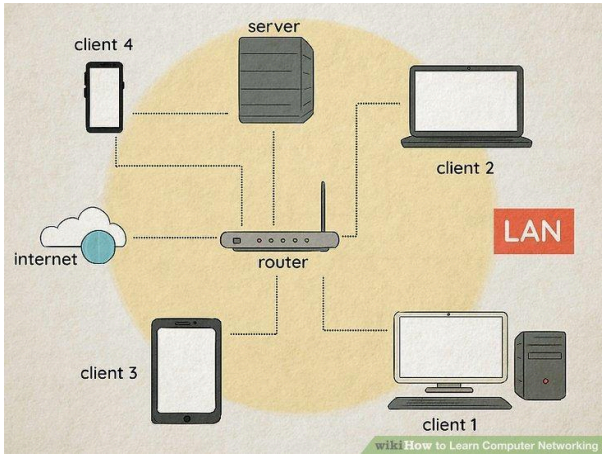
A computer network is a system of two or more computers that are connected together by a transmission medium for the exchange of data.

# 2 Types of Computer Networks

Computer networks can be classified according to their geographical size, transmission medium and network organisation type.

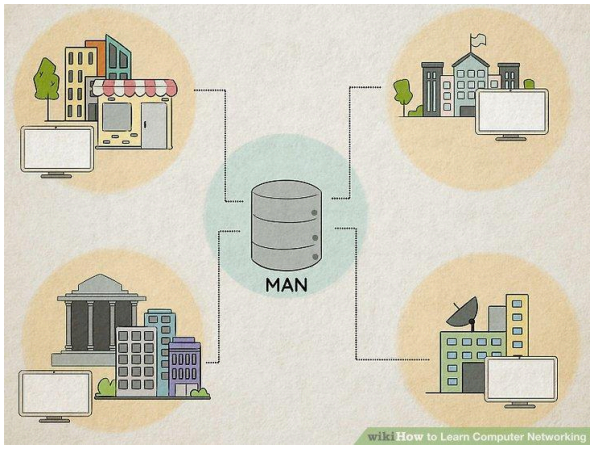
## 2.1 Local Area Network

A local area network (LAN) is a network of computing devices connected within a small geographical area, typically within the same building, such as a home, school, office or building. Due to the small number of connections supported and the close proximity of the devices, LANs typically provide faster data transfer than the other networks. Devices on a LAN can easily share resources like printers, files, and applications. A WLAN is the wireless version of a LAN.



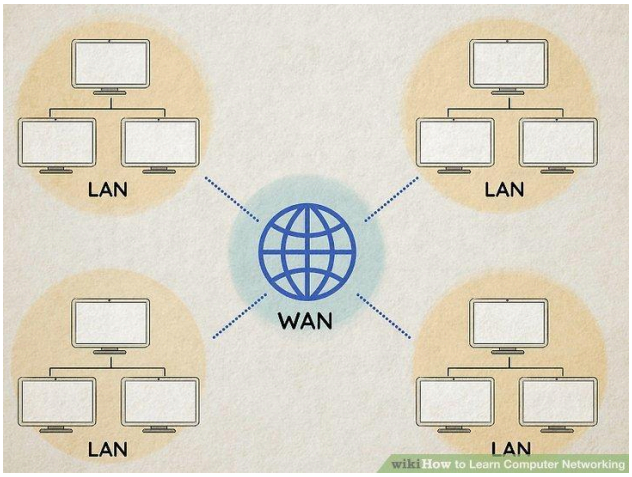

## 2.2 Metropolitan Area Network

A metropolitan area network (MAN) is a network of computing devices covering a larger geographical area (two or more buildings within the same town or city) than a LAN. A MAN is typically owned and operated by a large organisation such as a cities, business or government body.



## 2.3 Wide Area Network

A wide area network (WAN) is a network of computing devices covering a large-scale geographical area, typically across multiple geographical locations.




A WAN generally consists of multiple smaller networks such as LANs or MANs. A WAN differs from a LAN in several ways. Unlike a LAN, a WAN is not limited to a single geographical location and can span long distances via long-range transmission media such as telephone lines, fibre optic cables or satellite links. A WAN also uses more expensive and high-speed technology than a LAN.

WANs can be either private or public. Private WANs are built and maintained by large multinational companies and government organisations. The internet is an example of a public WAN and is considered as the largest WAN in the world.

## 2.4 Intranet

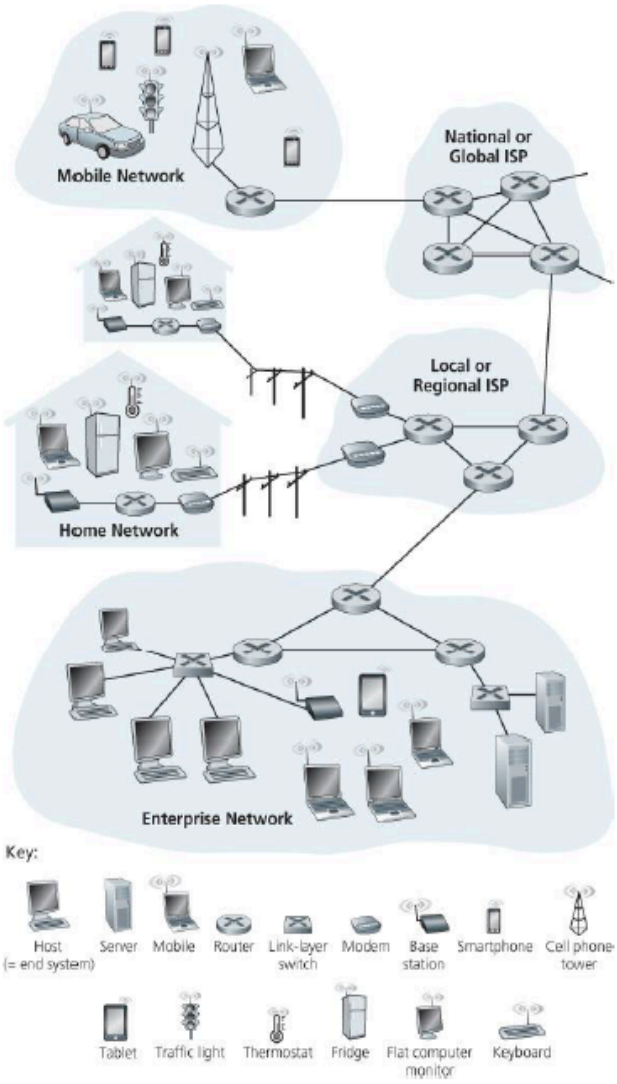
An intranet is a private network built within an organisation, like a company, school, or government agency. Access to the intranet is typically restricted to authorised users within the organisation. This allows them to share information and resources securely within a controlled environment.

Intranets are often used to host internal resources like company documents, employee portals, collaboration tools, and communication platforms. Since access is restricted, intranets offer a more secure environment for sharing sensitive information compared to the public internet.

## 2.5 Internet

The internet is a global, public network accessible to anyone with an internet connection. It is a network of networks linked by a broad array of electronic, wireless, and optical networking technologies. It is essentially an infrastructure that provides services to applications.

The internet functions through a layered system, with data traveling in packets across various networks based on unique IP addresses. Routers act as traffic directors, guiding this data flow, while Internet Service Providers provide the connection devices and this vast network. This complex system allows for global communication, information sharing, and access to a multitude of online resources and services.




*Source: Kurose, J., & Ross, K. (2016). Computer Networking: A Top-Down Approach (7th ed.). Pearson.*

## 2.6 Intranet vs Internet

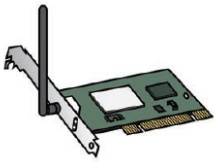
| Feature | Intranet | Internet |
|---|---|---|
| **Scope** | Private network within an organization | Public network accessible globally |
| **Access** | Restricted to authorized users within the organization | Open to anyone with an internet connection |
| **Content** | Internal resources relevant to the organization (documents, portals, tools) | Diverse content from various sources (news, social media, entertainment) |
| **Security** | More secure due to restricted access | Less secure due to open nature |
| **Connection** | Can be isolated from the internet or connected with security measures | Connects devices across the globe |
| **Purpose** | Internal communication, collaboration, secure resource sharing | Global communication, information sharing, access to online services |

# 3 Network Hardware

Network hardware refers to the physical components that make up a computer network. These components work together to enable devices to communicate and share resources.

Some essential network hardware includes:

- Computers with Network Interface Cards (NICs)



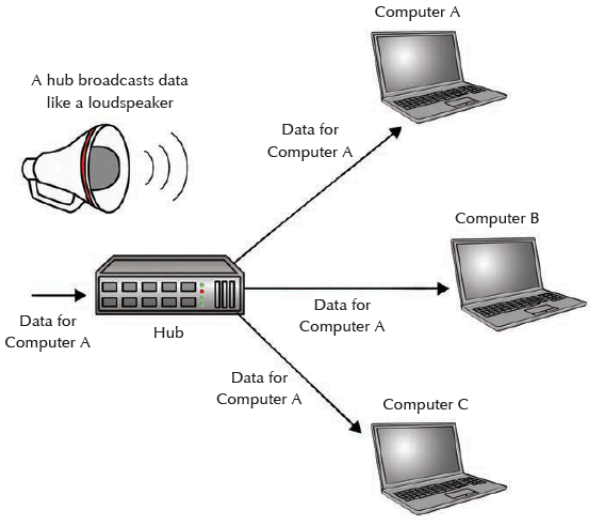
A network interface controller (NIC) provides the hardware interface to enable the transfer of data between a device and a network. An NIC may connect to a network physically or wirelessly. Most devices are equipped with a built-in NICs, one for Ethernet and one for Wi-Fi. The NIC should come with driver software that the operating system automatically configures and updates. Each NIC has a unique 'physical' address. This is sometimes referred to as the MAC address (explained later in the chapter). NIC gives a device its identification on the network. If the NIC is removed and inserted into a different device, the new device takes the address with it.

- Cables

Unshielded Twisted Pair (UTP) cables are the standard (CatX cables, where X is a number that indicates the speed a cable will fetch). For fast speeds, fibre optic cables are used.

- Hubs

A network hub is the simplest way to connect multiple devices to the same network, typically used in LAN. When a hub receives a packet, the packet is transmitted to all the devices that are connected to the hub. Likewise, when a device responds, the response is sent to every device that is connected to the hub. In this way, a hub acts like a loudspeaker as it broadcasts the data to all its connected devices without limiting the data to only the specific device it was intended for.



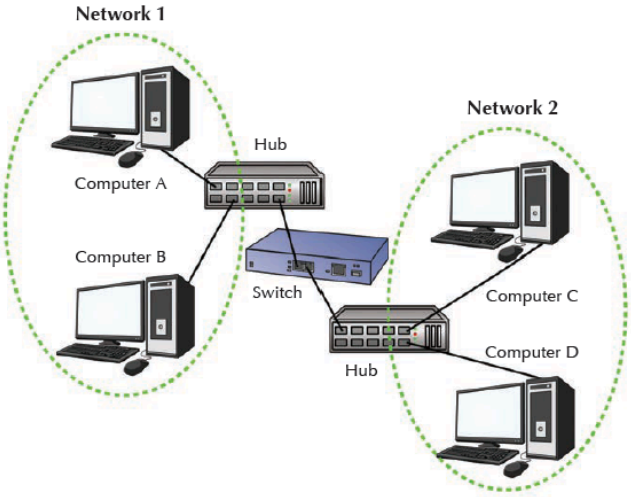

A hub is the cheapest option among all the network connection devices as it does not store any information about the devices that are connected to it. However, because the hub transmits data to all connected devices regardless of the intended recipient, it can cause bottlenecks that reduce the overall efficiency of the network.

- Switches

Switches are a fundamental component of most modern LANs. They connect multiple devices within a limited area (like a home, office, or school) and efficiently manage data flow.

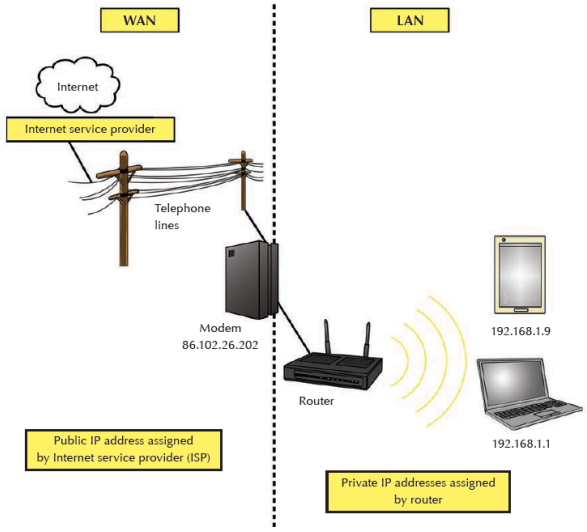
A switch can function on as a hub but it is a more intelligent device and, in particular, can keep track of the addresses of connected devices.

Switches receive data packets from devices, examine the destination address within the packet, and forward it to the intended recipient on the network. This dedicated forwarding helps avoid congestion and improves overall network performance compared to hubs, which simply broadcast data to all connected devices.




- Routers:

Network routers allow you to connect to other LANs and they assign IP addresses to your devices. A wireless router will allow computers with Wi-Fi NICs to connect with each other and with the internet. Routers are the most intelligent of the connecting devices. They are more complex devices that connect multiple networks and direct data packets to their final destination across different networks. They use routing protocols to determine the most efficient path for data to travel.




- Wireless Access Points

Wireless Access Points (WAP) are devices that create wireless local area networks (WLANs) allowing devices to connect to the network without cables. They communicate with devices using Wi-Fi technology.

- Modem:

A modem facilitates signal conversion and reversion (analog to digital and vice versa). Modems act as translators, converting the digital signals used by computers into a format suitable for transmission (analog), known as modulation, over physical media like telephone lines or cable. The reverse process is known as demodulation. Hence, the device is known as modem. They are typically used to connect a LAN to other LANs and to the internet to form a WAN.

# 4 Network Addresses

For devices to successfully communicate or exchange data over a computer network, they must be able to locate each other so that transmitted data can be directed to the correct destination.

This is similar to how mailing addresses are used to locate buildings or apartments for postal mail. For instance, mail can be successfully delivered to any location with a mailing address as the mailing address can be used to locate the correct destination. On the other hand, mail cannot be delivered to a location that does not have a mailing address as there would be no way to locate the destination. Just like how buildings and apartments have mailing addresses, each device on the Internet must have addresses for them to communicate and exchange data.

Devices on the Internet need two different kinds of addresses for data transmission across the network. They are Internet Protocol (IP) address and Media Access Control (MAC) address. Both MAC Address and IP Address are used to uniquely define a device on the internet. Network Interface Card (NIC)'s Manufacturer provides the MAC Address while the Internet Service Provider (ISP) provides IP Address.

The main difference between MAC and IP address is that MAC Address is used to ensure the physical address (or hardware address) of the computer. It uniquely identifies the devices on a network. IP addresses are used to uniquely identifies the connection of the network in which the device is in. As such, we can say that IP Addresses handle the logical routable connection from both computer to computer as well as network to network.

## 4.1 Media Access Control Address

A Media Access Control (MAC) address is a unique identifier assigned to a network interface controller (NIC) in a device. It acts like a permanent hardware fingerprint for that particular network connection. This ensures no two devices on the same network have the same MAC address.


The MAC address plays a crucial role in identifying devices within a local network segment (like your home network or office network). When a device transmits data on the network, the MAC address is included in the header of the data packet. Switches and routers use these MAC addresses to determine the specific destination within the network for the data packet.

MAC Addresses are unique 48-bits hardware number of a device, which is embedded into NIC during the time of manufacturing. Each MAC address is unique to the network card installed on a device, but the number of device-identifying bits is limited, which means manufacturers do reuse them. Each manufacturer has about 1.68 million available addresses, so when it burns a device with a MAC address ending in FF-FF-FF, it starts again at 00-00-00. This approach

assumes it is highly unlikely two devices with the same address will end up in the same local network segment.

## 4.2 Internet Protocol Address

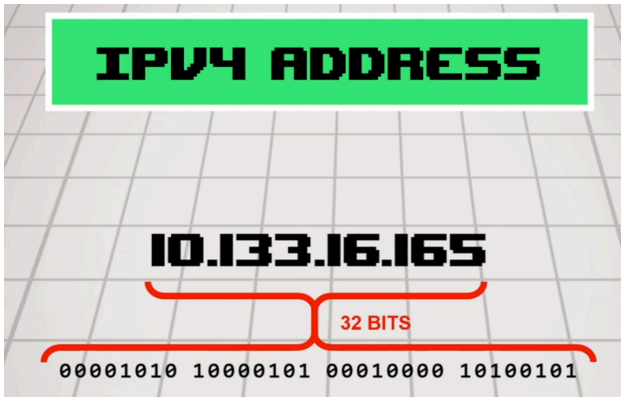
An Internet Protocol (IP) address is a unique numerical label assigned to devices connected to a network that uses the Internet Protocol (a set of rules for data transmission which are agreed by sender and receiver) for communication. It acts like an online mailing address, allowing information to be sent and received precisely between devices over the internet. Here's a deeper look at IP addresses:

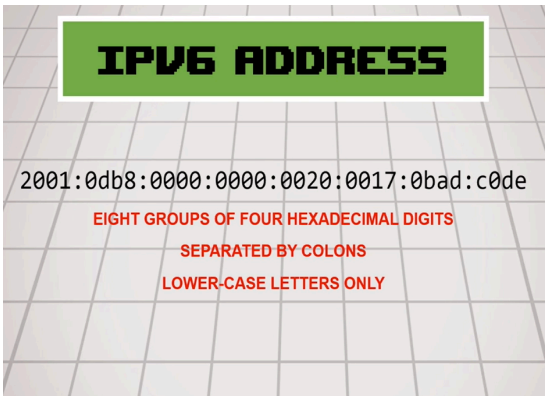
The primary function of an IP address is to identify and locate devices on a network. Just like a mailing address helps mail reach your doorstep, an IP address helps data packets reach the correct device on the internet.

IP addresses are typically written as four sets of numbers separated by periods, for example, 192.168.1.38. This format is known as Internet Protocol version 4 (IPv4), which is the most common version used today. A newer version, IPv6, is also being implemented to accommodate the growing number of internet-connected devices. IPv6 uses eight groups of hexadecimal digits separated by colons (e.g., 2001:db8:0:1234:0:567:8:1). IPv6 has been designed to coexist with IPv4, so both systems can run in parallel.

IP addresses can be assigned dynamically or statically. An Internet Service Provider (ISP) typically assigns dynamic IP addresses to home users. These addresses can change periodically. Static IP addresses, on the other hand, are fixed and do not change. They are often used for servers or other devices that need to be easily reachable on the internet.

IP addresses are essential for routing data traffic across the vast internet. When you request a website or online resource, your device sends a data packet with your IP address as the source address. Routers use this IP address to determine the most efficient path to send the data packets back to your device.



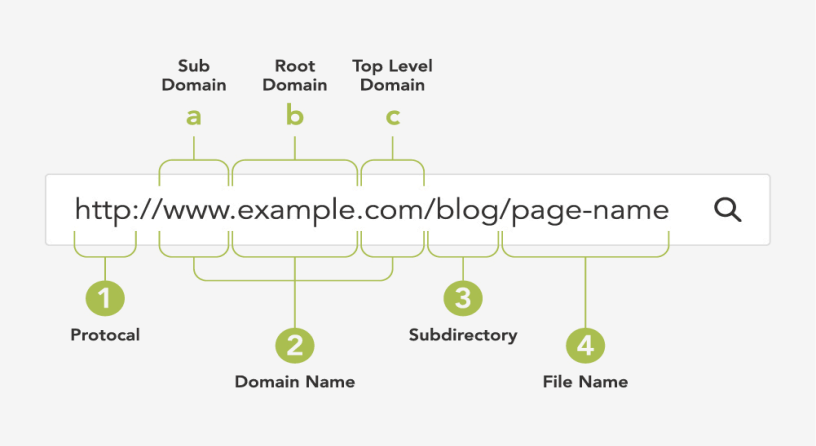
## 4.3 MAC vs IPv4 vs IPv6

| | MAC | IPv4 | IPv6 |
|---|---|---|---|
| **Function** | Used by network switches to direct data within the same network | Used by routers to direct data across different networks | Used by routers to direct data across different networks |
| **Length of Address** | 48 bits | 32 bits | 128 bits |
| **Address Format** | Six groups of two hexadecimal digits separated by colons or dashes | Four denary numbers between 0 and 255 separated by dots | Eight groups of four hexadecimal digits separated by colons<br>Can be shortened by omitting leading zeros as well as a section of consecutive zeroes |
| **Permanent** | Yes | No | No |

# 5 Domain Name Service

The Domain Name Service (DNS, also known as domain name system) is the primary way users refer to resources on the Internet. While computers talk to each other using MAC addresses and IP addresses, people prefer to use names and words instead.

The DNS service allocates readable domain names for Internet hosts and provides a system for finding the IP address for an individual domain name. Domain names eliminate the need for a user to memorise IP addresses. The DNS process involves converting a host name into an IP address the computer can understand. DNS also stores other information such as the list of mail servers. Often, DNS servers contain a database of uniform resource locators (URLs) with the matching IP addresses.




*Source: https://goup.la/technical-seo/learn/best-domain-strategy-for-international-expansion/*

The system is set up as a hierarchical distributed database which is installed on a large number of domain name servers covering the whole of the Internet. The domain name servers are connected in a hierarchy, with powerful root servers at the top of the hierarchy supporting the whole Internet. The root servers are replicated, meaning that multiple copies of all their data are kept at all times. DNS name space is then divided into non-overlapping zones. Each zone has a primary name server with the database stored on it. Secondary servers get information from this primary server.

There are more than 250 Top-Level Domains which are either generic (e.g. .com, .edu, and .gov) or represent countries (e.g. .uk and .sg).

The domain name is included in a URL, which identifies a web page, or an email address. Taking the URL, https://www.wikipedia.org/, for example:

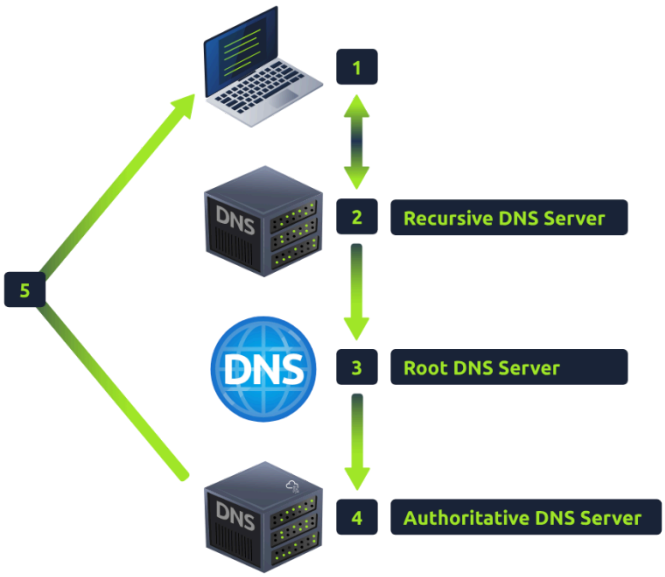
- www - refers to the subdomain
- .wikipedia – refers to the root domain
- .org – refers to the Top-Level Domain
- wikipedia.org - is the domain name

Looking up a domain name to find an IP address is called ‘name resolution’.

An example of the process of DNS request:

1. When you request a domain name, your computer first checks its local cache to see if you've previously looked up the address recently; if not, a request to your Recursive DNS Server will be made.

2. A Recursive DNS Server is usually provided by your Internet Service Provider, but you can also choose your own. This server also has a local cache of recently looked up domain names. If a result is found locally, this is sent back to your computer, and your request ends here (this is common for popular and heavily requested services such as Google, Facebook, Twitter). If the request cannot be found locally, a journey begins to find the correct answer, starting with the internet's root DNS servers.




3. The root servers act as the DNS backbone of the internet; their job is to redirect you to the correct Top-Level Domain (TLD) server, depending on your request. If, for example, you request www.tryhackme.com, the root server will recognise the Top-Level Domain of .com and refer you to the correct TLD server that deals with .com addresses.

4. The TLD server holds records for where to find the authoritative server to answer the DNS request. The authoritative server is often also known as the nameserver for the domain. For example, the name server for tryhackme.com is kip.ns.cloudflare.com and uma.ns.cloudflare.com. You'll often find multiple nameservers for a domain name to act as a backup in case one goes down. In summary, this authoritative DNS server is responsible for storing the DNS records for a particular domain name and where any updates to your domain name DNS records would be made.

5. Depending on the record type, the DNS record is then sent back to the Recursive DNS Server, where a local copy will be cached for future requests and then relayed back to the original client that made the request. DNS records all come with a TTL (Time To Live) value. This value is a number represented in seconds that the response should be saved for locally until you have to look it up again. Caching saves on having to make a DNS request every time you communicate with a server.

# 6 Transmission Control Protocol/Internet Protocol Model

The Transmission Control Protocol/Internet Protocol (TCP/IP) model is the fundamental framework for communication across the internet. It defines how data is broken down, addressed, routed, and delivered between devices. Think of it as a universal language that allows all sorts of computers and devices to exchange information over the vast network.

Here's a breakdown of the key aspects of the TCP/IP model, typically viewed as a 5-layered approach:

**Layers of the TCP/IP Model:**

1. **Application Layer:** It provides high-level functionality to end-users.

   It provides services directly to applications we use every day, such as web browsing (HTTP), email (SMTP), file transfer (FTP), and video conferencing.

2. **Transport Layer:** This layer provides functionality to transmit messages between any two programs.

   Two key protocols operate here:

   - **TCP (Transmission Control Protocol):** Ensures reliable and ordered delivery of data. It acts like a reliable courier service, establishing a connection, checking for errors, and guaranteeing packets arrive in the correct order.
   - **UDP (User Datagram Protocol):** Focuses on speed over guaranteed delivery. It's suitable for applications where speed is more critical than perfect accuracy, like online gaming or streaming media.

3. **Internet Layer:** This layer provides functionality to determine a route between any two devices.

   It is the heart of routing, handling addressing and routing data packets across networks using the Internet Protocol (IP). Imagine IP addresses like zip codes for devices on the internet; this layer ensures packets are delivered to the correct destination.

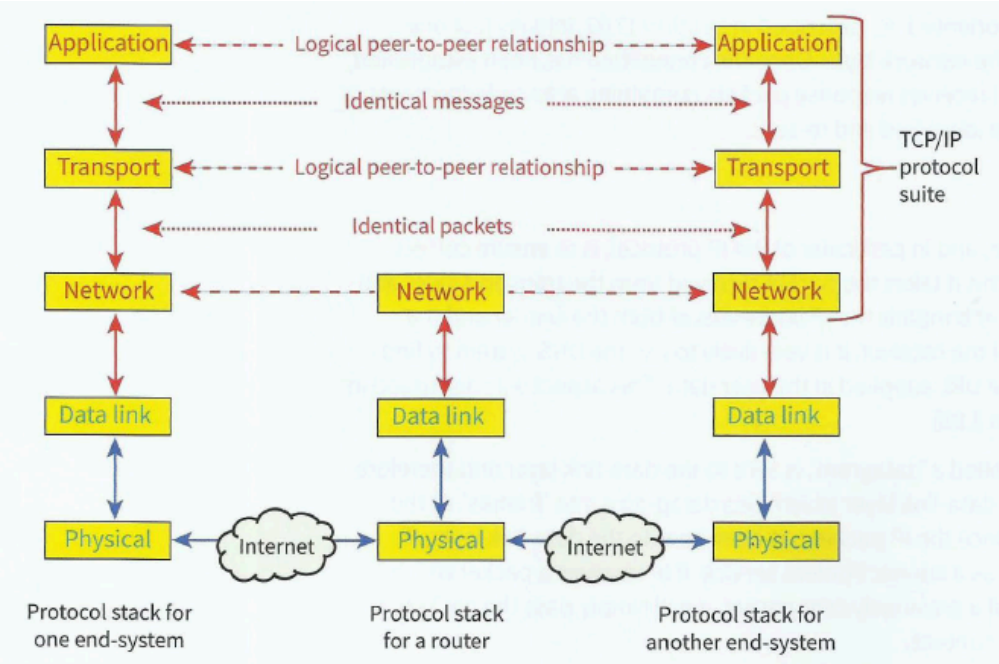
4. **The Link Layer:** This layer provides functionality to transmit packets from one device to an adjacent device.
5. **The Physical Layer:** This layer provides functionality to transmit individual bits through a transmission medium.

Note that in some literature, the Link layer and the Physical layers are known as the **Network Access Layer** which combines the functionalities of the physical and link layers. It is responsible for the physical transmission of data packets over the network medium (cables, Wi-Fi) and includes technologies like Ethernet and Wi-Fi.

## 6.1 TCP/IP protocol suite

Protocols are essential for successful transmission of data over a network. Each **protocol** defines a set of rules that must be agreed between sender and receiver.

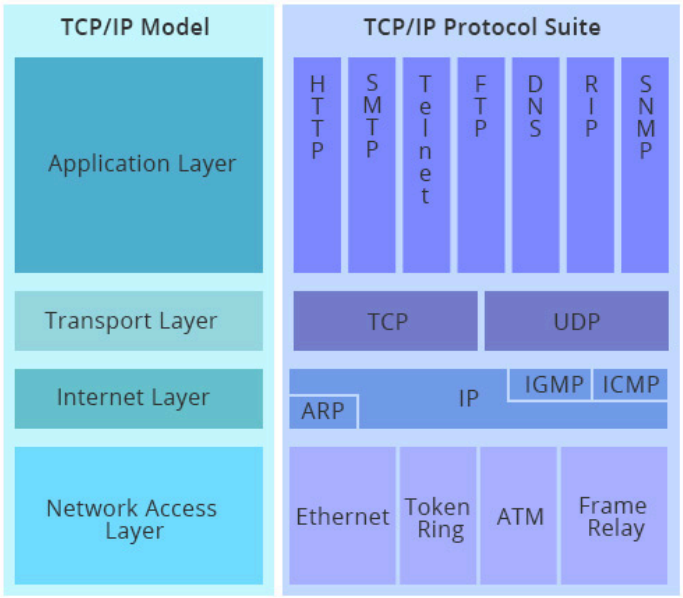
The complexity of networking requires a very large number of protocols. A protocol suite is a collection of related protocols. Transmission Control Protocol/Internet Protocol (TCP/IP) is the dominant protocol suite for Internet usage. TCP/IP can be explained on the basis of the network model shown below.




The TCP/IP protocol suite only operates at the top three layers. The lower layers operate with a different protocol suite, such as Ethernet. A router has no awareness of the two highest layers. The TCP/IP suite comprises a number of protocols, including the following:

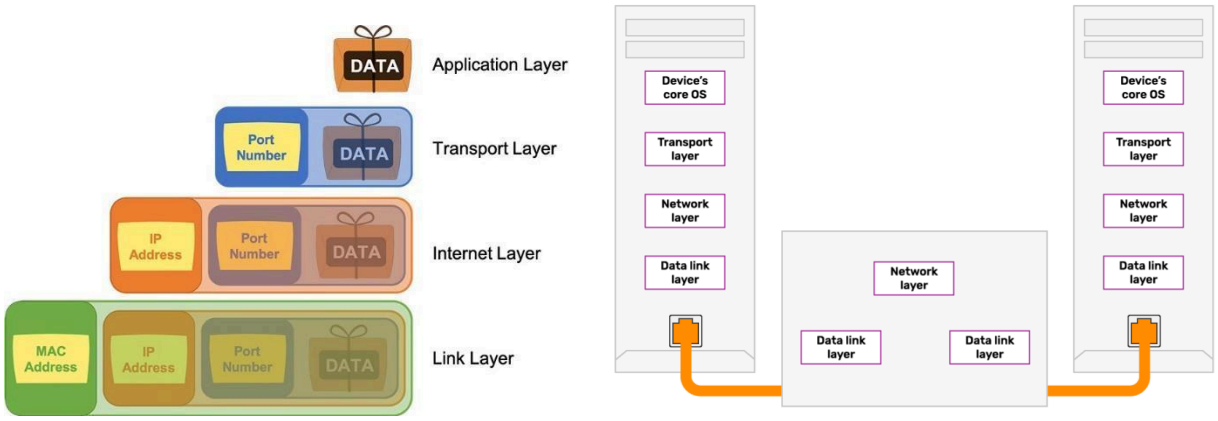
- Application layer: HTTP, SMTP, DNS, FTP, POP3
- Transport layer: TCP, UDP, SCTP
- Network layer: IP, IGMP, ICMP, ARP

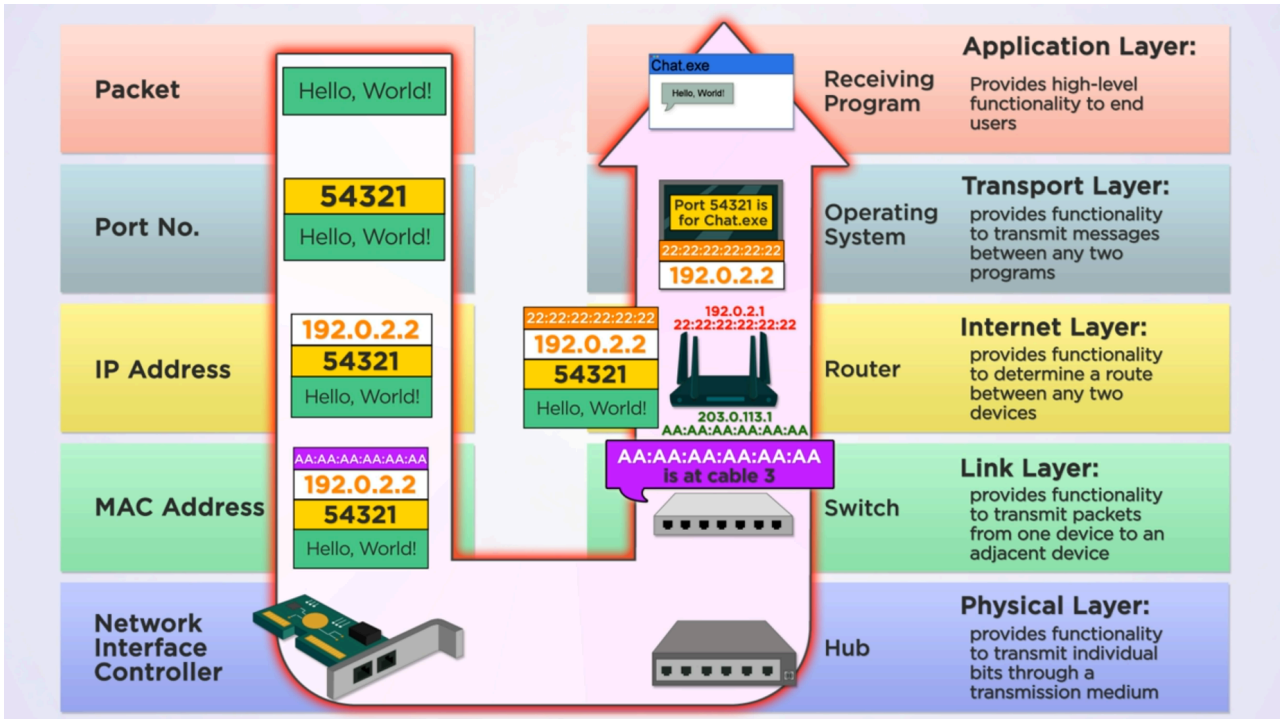
The selection has been chosen to illustrate that the TCP/IP suite encompasses a very wide range of protocols which is still evolving. We will look at some of the protocols listed in further details.




*Source: https://community.fs.com/blog/tcpip-vs-osi-whats-the-difference-between-the-two-models.html*

| Layer | Description | Protocols | Hardware/Software |
|---|---|---|---|
| Physical Layer | Handles physical transmission of data packets over the network medium cables, wires, or wireless signals. | Ethernet | Network cables, Wireless adapters, Network Interface Cards (NICs), Modems |
| Data Link Layer | Packages the data into frames and ensures error-free transmission between devices on the same network segment. | Ethernet, Wi-Fi, PPP (Point-to-Point Protocol) ARP (Address Resolution Protocol) | Network Interface Cards (NICs), Switches |
| Internet Layer | Routes data packets across different networks, determining the best path to reach the destination device | IP (Internet Protocol), ICMP (Internet Control Message Protocol) | Routers |
| Transport Layer | Manages reliable data transfer between applications | TCP (Transmission Control Protocol), UDP (User Datagram Protocol) | Not directly associated with specific hardware, but relies on the functionality of the Network Access Layer |
| Application Layer | Provides services directly to applications like web browsing, email, and file transfer. | HTTP (Hypertext Transfer Protocol), HTTPS (Secure Hypertext Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), DNS (Domain Name System) | Web browsers, Email clients, FTP clients, Operating Systems (for application support) |


DATA
Application Layer
Port Number
DATA
Transport Layer
IP Address
Port Number
DATA
Internet Layer
MAC Address
IP Address
Port Number
DATA
Link Layer
Device's core OS
Transport layer
Network layer
Data link layer
Network layer
Data link layer
Data link layer
Device's core OS
Transport layer
Network layer
Data link layer


Packet
Hello, World!
Port No.
54321
Hello, World!
IP Address
192.0.2.2
54321
Hello, World!
MAC Address
AA:AA:AA:AA:AA:AA
192.0.2.2
54321
Hello, World!
Network Interface Controller
Chat.exe
Hello, World!
Receiving Program
Application Layer:
Provides high-level functionality to end users
Port 54321 is for Chat.exe
22:22:22:22:22:22
192.0.2.2
Operating System
Transport Layer:
provides functionality to transmit messages between any two programs
22:22:22:22:22:22
192.0.2.2
54321
Hello, World!
192.0.2.1
22:22:22:22:22:22
203.0.113.1
AA:AA:AA:AA:AA:AA
Router
Internet Layer:
provides functionality to determine a route between any two devices
AA:AA:AA:AA:AA:AA is at cable 3
Switch
Link Layer:
provides functionality to transmit packets from one device to an adjacent device
Hub
Physical Layer:
provides functionality to transmit individual bits through a transmission medium

## 6.2 Transmission Control Protocol

If an application is running on an end-system where a 'message' is to be sent to a different end-system the application will be controlled by an application layer protocol. The protocol will transmit the user data to the transport layer. The TCP protocol operating in the transport layer now has to take responsibility for ensuring the safe delivery of the 'message' to the receiver. To do this, it creates sufficient packets to hold all of the data.

Each packet consists of a header plus the user data. As well as needing to ensure safe delivery, TCP also has to ensure that any response is directed back to the application protocol. Thus, one item in the header is the port number which identifies the application layer protocol. For example, for HTTP, the port number is 80. The packet must also include the port number for the application layer protocol at the receiving end-system. However, TCP is not concerned with the address of the receiving end system. If the packet is one of a sequence, a sequence number is included to ensure eventual correct reassembly of the user data.

The TCP protocol is connection-oriented. Initially just one packet of a sequence is sent to the network layer. Once the connection has been established, TCP sends the other packets and receives response packets containing acknowledgements. This allows missing packets to be identified and re-sent.

## 6.3 Internet Protocol

The function of the network layer, and in particular of the Internet Protocol (IP), is to ensure correct
routing over the Internet. To do this, it takes the packet received from the transport layer and adds a further header. This header contains the IP addresses of both the sender and the receiver. To find the IP address of the receiver, it is very likely to use the DNS system to find the address corresponding to the URL supplied in the user data. This aspect was discussed in some detail in Section 5.

The IP packet, which is usually called a 'datagram', is sent to the data-link layer and therefore to a different protocol suite. The data-link layer assembles datagrams into 'frames'. At this stage, transmission can begin. Once the IP packet has been sent to the data-link layer, IP has no further duty. IP functions as a connectionless service. If IP receives a packet which contains an acknowledgement of a previously sent packet, it will simply pass the packet on to TCP with no awareness of the content.

## 6.4 Ethernet Protocol

Ethernet is the other dominant protocol in the modern networked world. It is primarily focused on LANs. Ethernet protocol encapsulates the data from the previous layer into frames (another kind of packet) with a source and destination MAC address, and manages multiple transmissions on the media.

## 6.5 Other protocols

| Protocol | Layer | Full name and purpose |
|---|---|---|
| HTTP | Application | Hypertext Transfer Protocol. Used to make a request for a webpage. The server returns the page or an error code if there was a problem with the request. |
| HTTPS | Application | Hypertext Transfer Protocol Secure. Sends an encrypted request for a webpage. The server returns the encrypted page or an error code if there was a problem with the request. |
| FTP | Application | File Transfer Protocol. Used to upload or download a file. The server opens a data connection (over which the file will be transferred) or sends an error code if there was a problem with the request. |
| SMTP | Application | Simple Mail Transfer Protocol. Used to send an email to an email server. The server returns a code indicating whether or not the email could be delivered. |
| POP | Application | Post Office Protocol. Used to request any new emails for a specific email account. The server returns the emails (if there are any). |
| IMAP | Application | Internet Message Access Protocol. Used to synchronise a client email account with an account on a mail server. The server returns new emails (if there are any) and deletes any emails that were deleted locally on the client application. This allows a user to use multiple devices to access their email account. |
| DHCP | Application | Dynamic Host Configuration Protocol. Used to assign IP addresses and other configuration options to devices in a network. |
| TCP | Transport | Transmission Control Protocol. When data is to be sent (whether from client or server), the data is split into packets and each packet is given a sequence number. This is a reliable transmission protocol. At the receiving end, the packets are checked. If any packets go missing, they will be resent. |
| UDP | Transport | User Datagram Protocol. Data is split into packets (as with TCP). However, this is an unreliable transmission protocol. If any packets arrive out of sequence or are missing, they are ignored. UDP is suitable where data does not have to be 100% accurate but speed is |

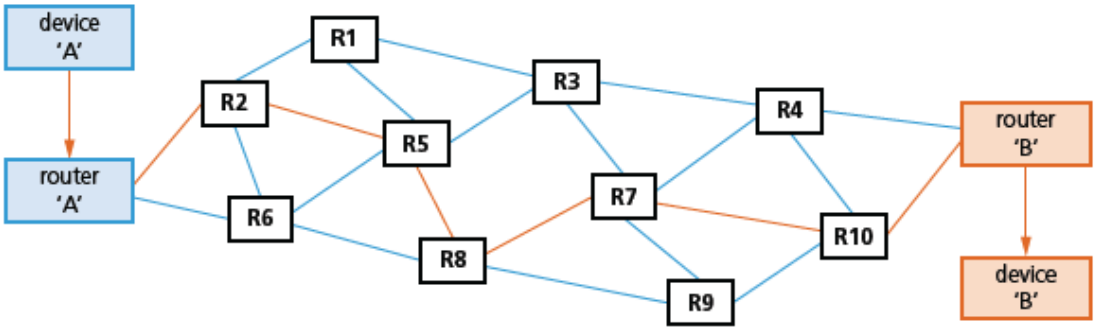
| | | |
|---|---|---|
| | | important, e.g. with some video streaming services or VOIP. |
| IP | Network (Internet) | Internet Protocol. Creates a new packet (imagine putting the packet from the transport layer into a new envelope). Adds the source and destination IP addresses to allow the packet to be delivered. These are the packets that are routed across the internet. |
| Ethernet and Wi-Fi protocols | Link (data link) | Encapsulates the data from the previous layer into frames (another kind of packet) with a source and destination MAC address, and manages multiple transmissions on the media. |

# 7 Transmission mode

For communication over an internetwork there are two possible approaches: circuit switching or packet switching.

## 7.1 Circuit switching

Circuit switching is a communication method where a dedicated communication path, or circuit, is established between two devices before data transmission begins. The circuit remains dedicated to the communication for the duration of the session, and no other devices can use it while the session is in progress. Circuit switching is commonly used in voice communication and some types of data communication.



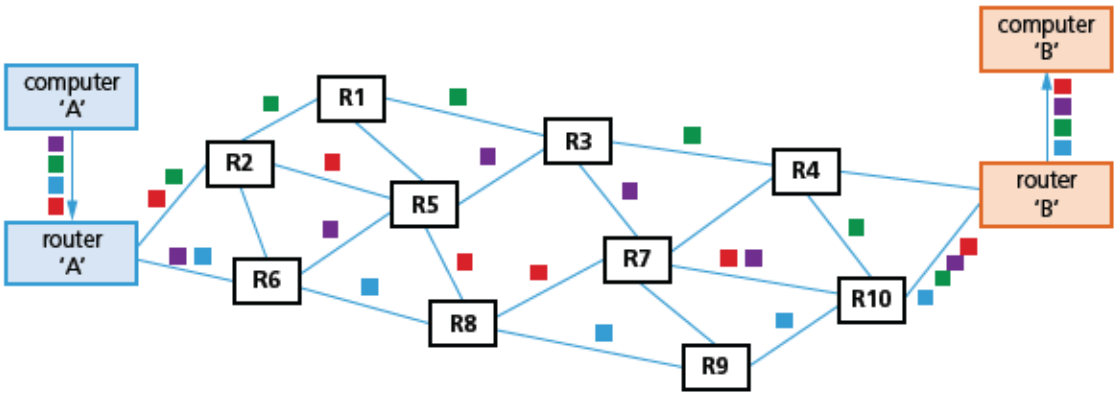

## 7.2 Packet switching

Packet switching is a communication method where data is divided into smaller units called packets and transmitted over the network. Each packet contains the source and destination addresses, as well as other information needed for routing. The packets may take different paths to reach their destination, and they may be transmitted out of order or delayed due to network congestion. Upon reaching the destination, the packets are reassembled in the correct order to form the original data.

When packet switching is used, there are two ways that the network can provide a service: connectionless service or connection-oriented service.

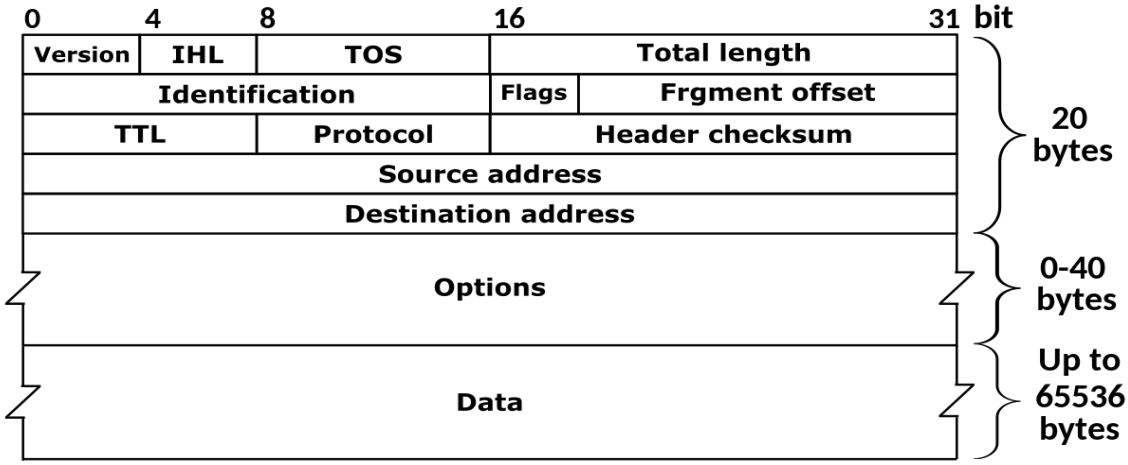
If a connectionless service is provided, a packet is dispatched with no knowledge of whether or not the receiver is ready to accept it.

In a connection-oriented service, the first packet sent includes a request for an acknowledgement. If this is received, the sender transmits further packets. If no acknowledgement is received the sender tries again with the first packet.




### 7.2.1 Packet Header

The diagram below shows an example of the header of a packet.




- IHL indicates the header length.
- Packets use Type of Service (TOS) to request special treatment (e.g. being put at the front of any queues).
- The identification, flags, and fragmentation offset fields keep track of each fragment when an IP packet is split up into smaller packets.
- The Time To Live (TTL) is set when the IP packet is sent off, and the number is reduced by 1 each time the packet goes through a router. When the time to live gets to 0, the packet is deleted. This stops packets from circulating in a loop.

### 7.2.2 Packet loss

At times, some routers may receive packets faster than they are able to route them on. These packets are buffered in memory and this introduces delays (referred to as a 'high latency'). For most traffic this is not an issue; it just means that the web page takes longer to load or a file download takes more time than expected. However, for other network traffic these delays may have more of an impact, particularly if you are talking over a voice call or playing an online game. If the buffering is severe, the router may run out of memory and packets are simply discarded.

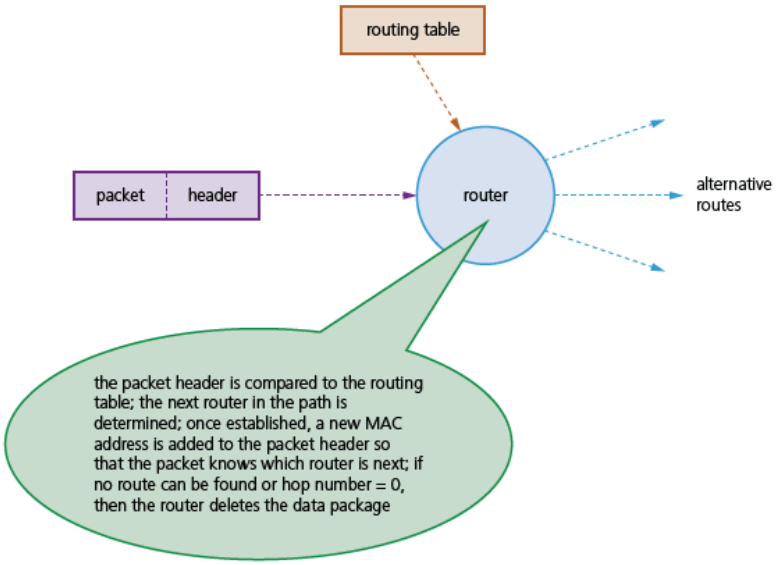
The TOS field in the packet header makes it possible to mark packets with a priority level and thus request special treatment i.e. to be placed at the front of the queue of packets to be routed. However, routers may choose to implement or ignore these requests.

A situation can occur where packets are sent to a destination address that is unreachable. Routers, unaware of this, may route such packets toward a default device. This default device may also pass the packet on, causing a loop to occur. In such circumstances, a packet could loop forever between routers. To prevent this, internet packets have a time to live (TTL) counter in their header. This is initially set when the packet is created, and reduced by one every time it goes through a router. If the counter reaches zero, the packet is discarded.

### 7.2.3 Routing tables

A router routes IP Packets across networks: a router device connects different links. It examines IP packet headers, looking at the destination address and consulting a routing table of known networks.

Routing tables contain the information necessary to forward a package along the shortest/best route to allow it to reach its destination. As soon as the packet reaches a router, the packet header is examined and compared with the routing table. The table supplies the router with instructions to send the packet to the next available router.



## 7.3 Packet switching vs Circuit switching

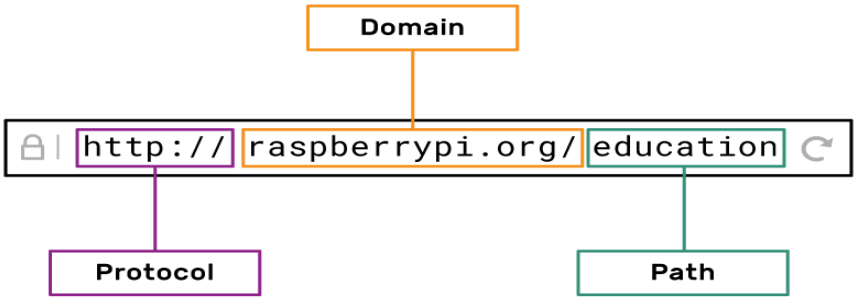
Circuit switching pre-allocates the use of the transmission link regardless of demand, with allocated but unneeded link-time going unused. Packet switching, on the other hand, allocates link use on demand. Link transmission capacity will be shared on a packet-by-packet basis only among those users who have packets that need to be transmitted over the link.

Although packet switching and circuit switching are both prevalent in today's telecommunication networks, the trend has certainly been in the direction of packet switching. Many of today's circuit-switched telephone networks are slowly migrating towards packet switching. In particular, telephone networks often use packet switching for the expensive overseas portion of telephone calls.

| **Feature** | **Packet Switching** | **Circuit Switching** |
|---|---|---|
| Data Transfer | Breaks data into packets, sent independently | Dedicated path established between sender & receiver |
| Routing | Packets can take different routes based on traffic | Dedicated path remains fixed for entire communication |
| Bandwidth | Dynamically allocated based on traffic | Guaranteed bandwidth for the communication |
| Efficiency | More efficient for bursty data traffic | Less efficient for bursty data traffic |
| Cost | Generally considered more cost-effective | Can be more expensive, especially for unused bandwidth |
| Applications | Ideal for data transfer (web browsing, email) | Ideal for real-time applications (voice calls, video conferencing) |

**Annex – HTTP, FTP, SMTP, POP, IMAP**

## How HTTP Works (How the Server Handles Requests from Clients)




HTTP is a protocol for any data exchange on the Web such as fetching resources. It is a client-server protocol, which means requests are initiated by the recipient, usually the Web browser. HTTP requests web pages from a web server and can send information when a user fills in a form. The web pages that get sent back are formatted using Hypertext Markup Language (HTML). A complete document is reconstructed from the different sub-documents fetched, for instance, text, layout description, images, videos, scripts, and more.

HTTP is gradually being replaced with Hypertext Transfer Protocol Secure (HTTPS). HTTPS encrypts the traffic between the application and the server to protect user's data. Many browsers denote HTTPS with a padlock symbol to show the link is secure. You should only enter personal information into websites that support HTTPS.

Refer to the following for more details on HTTP.

https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview

https://adacomputerscience.org/concepts/internet_application_layer?examBoard=all&stage=all&topic=the_internet

As a quick summary, the HTTP protocol works as follows:

- The client (usually a browser) opens a connection to the server and sends a request.
- The server processes the request generates a response and closes the connection if it finds a `Connection: Close` header. The request consists of a line indicating a method such as GET or POST, a Uniform Resource Identifier (URI) indicating which resource is being requested, and an HTTP protocol version separated by spaces.
  This is normally followed by a number of headers, a blank line indicating the end of the headers, and sometimes body data. Headers may provide various information

about the request or the client's body data. Headers are typically only sent for POST and PUT methods. The example request shown below would be sent by a browser to request the server foo.com to send back the resource in /index.html. In this example, no body data is sent because the method is GET (the point of the request is to get some data, not to send it).

```
GET /index.html HTTP/1.0
User-agent: Mozilla
Accept: text/html, text/plain, image/jpeg, image/gif, */*
Host: foo.com
```

The server receives the request and processes it. It handles each request individually, although it may process many requests simultaneously. Each request is broken down into a series of steps that together make up the request-handling process.

When this request message reaches the server, the server can take either one of these actions:

- The server interprets the request received, maps the request into a file under the server's document directory, and returns the file requested to the client.
- The server interprets the request received, maps the request into a program kept in the server, executes the program, and returns the output of the program to the client.
- The request cannot be satisfied, the server returns an error message.

The server generates a response that includes the HTTP protocol version, HTTP status code, and a reason phrase separated by spaces. This is normally followed by a number of headers. The end of the headers is indicated by a blank line. The body data of the response follows. A typical HTTP response might look like this:

```
HTTP/1.0 200 OK
Server: Sun-Java System-Web-Server/6.1
content-type: text/html
Content-length: 83

<HTML>
<HEAD><TITLE>Hello World</Title></HEAD>
<BODY>Hello World</BODY>
</HTML>
```

The status code and reason phrase tell the client how the server handled the request. Normally the status code 200 is returned, indicating that the request was handled successfully and the

body data contains the requested item. Other result codes indicate a redirection to another server or the browser's cache, or various types of HTTP errors such as 404 Not Found.

The browser receives the response message, interprets the message and displays the contents of the message on the browser's window according to the media type of the response (as in the Content-Type response header). Common media type include "text/plain", "text/html", "image/gif", "image/jpeg", "audio/mpeg", "video/mpeg", "application/msword", and "application/pdf".

To learn about other Application-layer protocols in action, refer to:
https://adacomputerscience.org/concepts/internet_application_layer?examBoard=all&stage=all

### File Transfer Protocol (FTP)

As the name suggests, FTP moves files from a client device to a file server. The FTP application may be a dedicated application on the client device or built into a web browser or other software.

Some examples where FTP would be used are:

- A website developer designing a website on a local device and then transferring all the files to the web server
- Downloading software
- Downloading a music file
- Some video clips embedded in web pages

Generally, if you end up with an actual file in a folder on your device, you probably used FTP to get that file.

Just like HTTP, FTP is not secure, but there is a secure version of it: SFTP.

### Simple Mail Transfer Protocol (SMTP)

Email was initially one of the main uses of the internet and still accounts for a significant number of data transactions.

Users have email addresses such as test@raspberrypi.org. A specific email server manages all the accounts with the same address after the @.

When a user sends a message, using an email client, an app, or a webmail page, the message is sent to the mail server using a protocol called SMTP.

If the mail is for someone in the same organisation, the mail is stored until that users connects. If not, the domain name of the destination is resolved using the DNS and the email is forwarded to the destination mail server. Again SMTP is used to transfer the email.

### Post Office Protocol (POP) and Internet Message Access Protocol (IMAP)

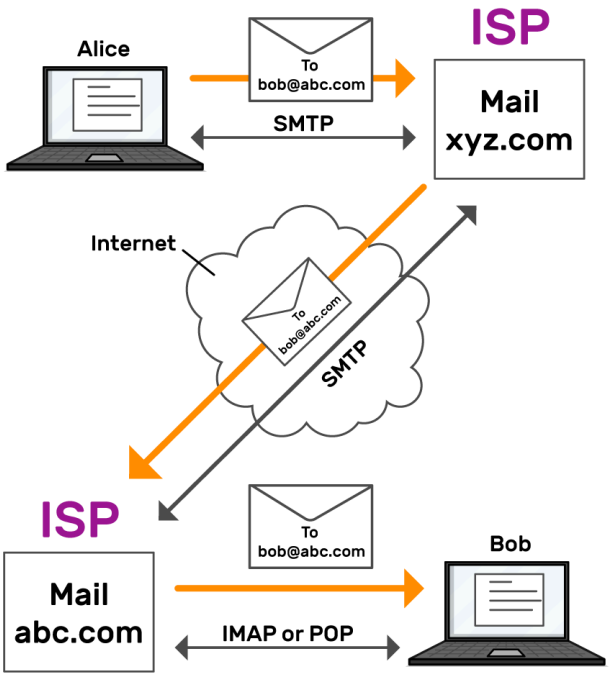
Once an email arrives at the destination mail server, it is stored in the recipient's 'mailbox' until they connect to read it.

If a user connected using the Post Office Protocol, or POP, then the actual email is transferred to the client's device and deleted from the server.

This used to cause problems when groups of people were sharing an email account, e.g. a specific company account that people message when they need support (such as support@acme.com).

To combat this and give additional functionality, the Internet Message Access Protocol, or IMAP was developed. IMAP allows messages to be retrieved while a copy is maintained on the server; a specific request to delete an email is sent when the email is no longer needed.

With webmail, IMAP is used to pull the emails from the web server so they are displayed on the web page.



**References**


1. https://www.wikihow.com/Learn-Computer-Networking#
2. O level textbook Computing (2020)
3. https://www.openbookproject.net/courses/intro2ict/networking/intro.html
4. David Watson & Helen Williams: Cambridge International AS and A level computer science coursebook
5. Sylvia Langfield and Dave Duddell: Cambridge International AS and A level computer science coursebook
6. https://tryhackme.com/r/room/dnsindetail
7. https://adacomputerscience.org/concepts/network_protocols?examBoard=all&stage=all&topic=networking
8. https://www.tutorialspoint.com/circuit-switched-vs-packet-switched-networks
9. https://adacomputerscience.org/concepts/internet_structure?examBoard=all&stage=all&topic=the_internet
10. https://adacomputerscience.org/concepts/internet_structure?examBoard=all&stage=all&topic=the_internet