**What is an IP Address?**

An **IP address** (Internet Protocol address) is a unique number assigned to every device connected to the internet or a local network. It helps devices find and communicate with each other.

* Think of it like a **home address** for your computer or phone on the internet.
* It looks like numbers separated by dots (IPv4) such as 192.168.1.1 or like longer hexadecimal numbers separated by colons (IPv6) such as 2001:0db8:85a3::8a2e:0370:7334.

**In IPv4 addresses:**

* An IPv4 address is made up of **4 octets**.
* Each octet is a number between 0 and 255.
* For example, in the IP address 192.168.0.1,
  + 192 is one octet,
  + 168 is another octet, and so on.
* These octets are separated by dots (periods), which is why IPv4 is sometimes called "dotted decimal notation."

**Types of IPv4 Addressing**

IPv4 basically supports three different types of addressing modes:

* **Unicast Addressing Mode**: This addressing mode is used to specify single sender and single receiver. Example: Accessing a website.
* **How it works:** The message is delivered only to the device with that particular IP.
* **Broadcast Addressing Mode:**This addressing mode is used to send messages to all devices in a network. Example: sending a message in local network to all the devices.
* **Usage: Sending a message to all devices in the same network.**.
* **Multicast Addressing Mode:**This addressing mode is typically used within a local network or across networks and sends messages to a group of devices. Example: Streaming audio to multiple devices at once.
* **Usage:** Sending data to a specific group of devices interested in that data.

**How does this help?**

Imagine you want to send a letter:

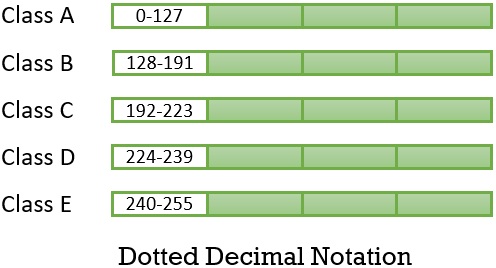
1. **Network part (City):**
   * You first write the **city name** on the envelope so the postal service knows which city to send it to.
   * All houses in that city share the same city name.
2. **Host part (House number):**
   * Inside the city, the postal service uses the **house number** to deliver the letter to the correct house.

**In IPv4 terms:**

* **Network Part:** tells routers *which network* (like a city) the data should be sent to.
* **Host Part:** tells the final device (like a house) on that network *exactly where* to deliver the data.

**Bonus: Subnet Mask in this analogy**

* The **subnet mask** acts like the rule that decides which part of the address is the city and which part is the house.
* For example, subnet mask 255.255.255.0 means first 3 parts (192.168.1) are city, last part (10) is house.

IP addresses use the concept of **classful addressing** which **splits** the available address space into five classes **A, B, C, D & E**. IPv4 addresses are represented using **32-bit** addresses. The **32-bit IPv4 address** is also referred to as the **4-byte address** or **4-octet address**. So, we can conclude that the **address space of IPv4** is **232**which is equal to**4,294,967,296.** 

IPv4 addresses of class A, B & C the first part of the address is considered as net-id (Network id) and the second part of the address is called host-id.

Class A

The **network id** of class A is defined by the **first byte** of the 32-bit IPv4 address. In class A, the **first bit** of the **net-id**stays ‘**0′**to define that the IPv4 address belongs to the class A and the other **7 bits** of the net-id can be changed to defines different blocks in class A. As the first bit is preserved the remaining seven bits calculate the number of blocks in the class A i.e. **27= 128 blocks**. There are 128 blocks in class A, as the addressing would start from 0 the range of blocks will be from 0-127.

Class B

The **network id** or the net-id of **class B** is defined using the **first two bytes** of the IPv4 address. The first **two bits** of **net-id** stays ‘**10**’ to define that the IPv4 address belongs to the**class B** and the remaining **14 bits** of net-id can be changed to calculate the number of **blocks** in class B i.e. **214= 16,384.**

The **next two bytes** to of IPv4 address denotes the**host id** in class B which is **16 bits**. The number of hosts can be calculated as **216= 65,536**. So, we conclude that we can assign 16,384 blocks from class B to 16,384 organizations where each organization can have 65,536 hosts connected to the network.

Class C:

 class C the **network id is** defined by the **first 3 bytes** of the IPv4 address. The **first 3 bits** in **network id** stay ‘**11**0’ to define the**class** and the remaining **21 bits defines the number of block**s in class B. The number of blocks can be calculated as **221= 2,097,152.**

The **last byte** of the IPv4 address in class C defines the **host-id**. The **number of hosts** can be calculated as **28= 256**. So, we conclude that we can assign 2,097,152 blocks from class C to 2,097,152 organizations where each organization can have 256 hosts connected to the network.

Class D:

e class A, B & C, class D does **not divide** IPv4 into **net-id** and **host-id**. **All the addresses** of class D are of **one single block**. The class D addresses are designed for **multicasting**. The **first four-bit** of entire 32-bit addresses of **class D** stays ‘**1110**’ to define the class.

The remaining **28** bits from the 32-bit addresses of class D can be changed to define the**address space** of class D. So, the number of addresses in class D is **228=2,68,435,456.**

ss E addresses are one block addresses. The addresses in class E are not split into net-id and host-id. The addresses in class E are **reserved for future** use. The **first four bits** of entire 32-bit IPv4 addresses of class E stays ‘**1111**’. The remaining **28-bit** changes to define the number of addresses in **class E** i.e. **228=2,68,435,456**.

1. The main issue here is; we are **not assigning** addresses according to **user requirements**. We directly assign a **block of a fixed size** which has a **fixed number of addresses** which leads to wastage of address.
2. Subnetting
3. As class blocks of A & B are too large for any organization. So, they can **divide** their large network in the **smaller subnetwork** and **share** them with other organizations. This whole concept is **subnetting**

[What is Classful Addressing in IPv4? Subnetting, Supernetting & Disadvantages - Binary Terms](https://binaryterms.com/classful-addressing-in-ipv4.html)

If we need toset of company with 254 host then we choos class c as it net id 21 and 8 bit for host so 2^8 is 254 but if u go for 300 then class b only need to choose.

Classless address name cidr-classless interdomain routing

It is possible with subnetting

<IP address>/<prefix length>

Example: 192.168.1.0/24

 192.168.1.0 → the base IP

 /24 → means the first **24 bits are network**, the remaining **8 bits are for hosts**

**Why Classless is Better:**

| **Classful (Old)** | **Classless (CIDR - New)** |
| --- | --- |
| Fixed class ranges (A/B/C) | Flexible allocation |
| Wasted IPs in many cases | Saves IPs using subnetting |
| No customization | Full control over network size |

**TCP (Transmission Control Protocol)**

**✅ Overview:**

TCP is a **connection-oriented** and **reliable** transport layer protocol. It ensures **data integrity**, **correct order**, and **delivery confirmation** between devices on a network.

**🔍 Key Concepts in TCP:**

1. **Connection-Oriented**:
   * TCP requires a **3-way handshake** to establish a connection:
     + **SYN → SYN-ACK → ACK**
   * Ensures both devices are ready before exchanging data.
2. **Reliable Data Transfer**:
   * If a packet is lost or damaged, TCP **retransmits** it.
   * Every packet has a **sequence number** for tracking order.
3. **Ordered Delivery**:
   * TCP uses **sequence numbers** and **acknowledgments (ACKs)** to make sure data is assembled in the correct order.
4. **Flow Control**:
   * TCP manages how fast data is sent using a **window size** to avoid overwhelming the receiver.
5. **Error Checking**:
   * Uses **checksums** to detect errors in data during transmission.
6. **Congestion Control**:
   * Dynamically adjusts data transmission based on network congestion using algorithms like:
     + **Slow Start**
     + **Congestion Avoidance**
     + **Fast Retransmit and Recovery**

**🧾 Real-Life Analogy:**

TCP is like a **phone call**:

* You dial (handshake).
* You talk (send and receive messages in order).
* If you can’t hear, you ask to repeat (retransmit).
* You hang up after confirming the conversation is over.

**🔧 Technical Summary:**

| **Feature** | **TCP** |
| --- | --- |
| Protocol Type | Connection-oriented |
| Reliability | Guaranteed (resends lost packets) |
| Packet Ordering | Guaranteed |
| Speed | Slower due to overhead |
| Use Cases | Web (HTTP, HTTPS), email, file transfer |

**🔷 UDP (User Datagram Protocol)**

**✅ Overview:**

UDP is a **connectionless**, **unreliable**, but **very fast** transport layer protocol. It is ideal for real-time applications where speed is more critical than accuracy.

**🔍 Key Concepts in UDP:**

1. **Connectionless**:
   * No handshake.
   * Simply sends packets called **datagrams** to the receiver.
2. **Unreliable Delivery**:
   * No guarantee the packet will reach the destination.
   * No acknowledgment, retransmission, or ordering.
3. **Fast and Lightweight**:
   * Minimal header (8 bytes compared to TCP’s 20–60 bytes).
   * Less delay and overhead.
4. **No Congestion or Flow Control**:
   * UDP does not reduce or adjust speed based on receiver or network load.

**🧾 Real-Life Analogy:**

UDP is like **sending letters by post without tracking**:

* You send letters quickly.
* You don't know if they arrive or in what order.
* Useful if speed is more important than accuracy.

**🔧 Technical Summary:**

| **Feature** | **UDP** |
| --- | --- |
| Protocol Type | Connectionless |
| Reliability | Not guaranteed |
| Packet Ordering | Not guaranteed |
| Speed | Faster (low overhead) |
| Use Cases | Video streaming, VoIP, online gaming |

[IP Commands for Interviews](https://chatgpt.com/c/68493599-b028-8006-9595-3271a7800adf)

**✅ What is a VPN (Virtual Private Network)?**

A **VPN (Virtual Private Network)** is a secure way to connect to another network over the internet. It **creates a private, encrypted tunnel** between your device and the VPN server, hiding your data from hackers, ISPs, and even government surveillance.