7. IP ADDRESS

An **IP address (Internet Protocol address)** is like a **home address** for devices on a network. It helps computers, servers, and other devices find and communicate with each other.

Imagine sending a letter to a friend. If you don't write their address, how will the postman deliver it? **IP addresses work the same way**—they tell computers where to send and receive data.

Every device connected to the internet or a network **must have an IP address** to communicate.

A brand-new computer doesn't have an IP address by itself. But the moment you connect it to a network (like Wi-Fi or Ethernet), it gets one

How Does a New Computer Get an IP Address?

When you connect a new computer to the internet:
1️⃣ **Your router (or network) assigns an IP address (Private IPs)** using something called **DHCP (Dynamic Host Configuration Protocol).**
2️⃣ **Your computer checks the network** and asks, "Hey, can I get an IP address?"
3️⃣ **The router gives your computer an IP address** from its available range.

So, a new computer **doesn't have an IP by default**, but as soon as it joins a network, it gets one automatically! 🚀

Step 2: Types of IP Addresses

There are **four main types** of IP addresses:
1️⃣ **Public IP**
2️⃣ **Private IP**
3️⃣ **Static IP**
4️⃣ **Dynamic IP**

1️⃣ Public IP Address (Like Your Home's Main Address)

- A **Public IP** is an address that is **visible on the internet**.
- This is the **IP address of your router** that your **Internet Service Provider (ISP)** gives you.
- Think of it like your home's **main address** that everyone outside can see.

2️⃣ Private IP Address (Like Room Numbers Inside a House)

- A **Private IP** is used **inside** your home or office network.
- Each device connected to your Wi-Fi (laptop, phone, smart TV) gets a **different Private IP** from your router.
- Think of it like room numbers in a house. The outside world (internet) doesn't see them.

📌 Common Private IP Ranges:

- `192.168.x.x` (Most common, used in homes)
- `10.x.x.x` (Used in large networks)
- `172.16.x.x - 172.31.x.x` (Used in businesses)

🚀 **Key point**: Your Private IP **cannot** be accessed directly from the internet.

3️⃣ Static IP Address (Never Changes)

- A **Static IP** is **fixed** and does **not change** over time.
- Usually used by **servers** and websites.
- Example: Google's servers have a **Static IP** because they must always be available at the same address.

4️⃣ Dynamic IP Address (Changes Over Time)

- A **Dynamic IP** is **assigned automatically** by your ISP (internet provider) and **changes** after some time.
- Most home users have **Dynamic IPs** because ISPs don't give the same IP forever.

🖥️ Example:

- If you restart your Wi-Fi router, your Public IP **might change** (because it's Dynamic).

How Does NAT Work? (The Role of the Router)

Your **Wi-Fi router** is the key player here. It does something called **NAT (Network Address Translation)** to manage internet traffic:

NAT (Network Address Translation) is a feature of the router that allows multiple devices to share a single public IP address.

- A device (e.g., your laptop with private IP 192.168.1.2) sends a request to a website (e.g., google.com).
- The router intercepts this request and replaces the private IP (192.168.1.2) with its public IP (203.0.113.1).
- The router keeps track of this request in a NAT table, noting which device made the request.
- When the website responds, the router uses the NAT table to forward the response back to the correct device (192.168.1.2).

Why Do ISPs Use NAT?

- There are **limited** Public IPs in the world. NAT helps **reduce** the number of Public IPs needed.
- It also adds a **security layer** by keeping your Private IPs hidden from the internet.

Step 3: IPv4 Address Structure

An **IPv4 address** is a **unique number** assigned to a device to identify devices on a network.

It is a **32-bit address**, which means it consists of 32 binary digits (bits).

These 32 bits are divided into **4 octets** (groups of 8 bits), and each octet is represented as a decimal number between 0 and 255.

An IPv4 address is **32 bits long** and is written in **decimal format** like this:

Example: `192.168.1.1`

Each part is called an **octet** because it represents **8 bits** (a total of 4 octets = 32 bits).

Decimal	Binary Representation
192	`11000000`
168	`10101000`
1	`00000001`
1	`00000001`

$2^{32} = 4,294,967,296$
~4.29 billion IPv4

Mobile operators use (CGNAT) carrier Grade NAT. This allow them to assign private IP address

Why always 2?
= Because each bit has 2 options

1 byte = 8 bits
1 Bit = 0 & 1

So, **192.168.1.1 in binary** is:

🖥 `11000000.10101000.00000001.00000001`

- An IPv4 address is written in **dotted-decimal notation**, where each octet is separated by a dot (.).
- Each number (e.g., 192.168.1.1) represents an 8-bit binary value.
- The range of each octet is 0 to 255 (since 2^8 =**256 possible values**).

💡 **Why is this important?**
Computers **read IPs in binary**, while we humans prefer **decimal format**.

📌 IPv4 Address Classes

IPv4 addresses are divided into **classes** based on their range and purpose:

Class	Range	Purpose	Default Subnet Mask
A	`1.0.0.0` to `126.255.255.255`	Large networks (e.g., corporations)	`255.0.0.0`
B	`128.0.0.0` to `191.255.255.255`	Medium-sized networks (e.g., universities)	`255.255.0.0`
C	`192.0.0.0` to `223.255.255.255`	Small networks (e.g., home networks)	`255.255.255.0`
D	`224.0.0.0` to `239.255.255.255`	Multicasting (not for individual devices)	N/A
E	`240.0.0.0` to `255.255.255.255`	Reserved for experimental use	N/A

- **Class A:** First octet: The first bit is always 0
- **Class B:** First octet: The first bit is always 10
- **Class C:** First octet: The first bit is always 110
- **Class D:** First octet: The first bit is always 1110
- **Class E:** First octet: The first bit is always 1111

IPv4 addresses are 32 bits long.
* each bit can be either 0 or 1
* So, with 1 bit, you have 2 Possibilities (0, 1).
* with 2 bits, 2^2 = 4 Possibilities (00, 01, 10, 11).
* with 32 bits, you get 2^{32} = 4.2 B(P).

• Because IP addresses are binary based - computers work in base 2 (binary). So not base 3 or 4

* IP addresses is not about base 3 & 4 - it is about binary representation - which only supports 0 & 1

- introduced in 1993 to replace the old class system

CIDR:

- Modern networking primarily uses **CIDR**, which allows for more flexible allocation of IP addresses.
- **Classless Inter-Domain Routing** (CIDR) allows network routers to route data packets to the respective device based on the indicated subnet. Instead of classifying the IP address based on classes, routers retrieve the network and host address as specified by the CIDR suffix

CIDR format
↳ IP-d / Prefix-length
= 192.168.1.0/24

4 IPv4 Address Ranges

There are special ranges for **private** and **public** IPs:

■ **Private IPv4 Ranges (Not used on the internet)**:

- `10.0.0.0` to `10.255.255.255`
- `172.16.0.0` to `172.31.255.255`
- `192.168.0.0` to `192.168.255.255`

■ **Public IPv4 Addresses:** Any address **outside** the private ranges is public and can be accessed on the internet.

5 Special IPs

Loopback Address:

127.0.0.1 is used to test the network stack on a device (e.g., "localhost").

APIPA (Automatic Private IP Addressing):

169.254.0.0 to 169.254.255.255 (used when a device cannot obtain an IP from a DHCP server).

Broadcast Address:

255.255.255.255 is used to send data to all devices on a network.

Default route:

0.0.0.0 **Default route** (means "any IP")

What is a Broadcast IP?

The **broadcast IP** is a special address used to send data to **all devices** in a network.

- Every device in the network listens to this IP.
- If a message is sent to the **broadcast IP**, every device in that network receives it.

📌 How Do You Find the Broadcast IP?

The **broadcast IP** is always the **last possible IP** in a network.

Example 1:

📌 **IP Address:** `192.168.1.50`
📌 **Subnet Mask:** `255.255.255.0`

Octet	IP Address	Subnet Mask	Network/Broadcast
1st	192	255	Network
2nd	168	255	Network
3rd	1	255	Network
4th	50	0	Host

⬛ **Network IP:** `192.168.1.0`
⬛ **Broadcast IP:** `192.168.1.255`

All valid devices can use **192.168.1.1 to 192.168.1.254** 🚀

Step 4: Subnet Mask

A **Subnet Mask** tells the network **which part of an IP belongs to the network and which part belongs to the device (Host)**

Subnetting involves using a **subnet mask** to divide a network into smaller networks.

A **Subnet Mask** decides how many bits are for the **network** and how many are for **hosts** (devices).

An IPv4 address has two parts:

⬛ **Network Part** - Identifies the network or subnet to which the device belongs.
⬛ **Host Part** - Identifies the specific device within the network.

📌 **Example: In `192.168.1.1`, depending on the subnet, the first 3 parts might represent the network, and the last part (1) is the device.**

Subnet Mask	CIDR	Hosts
255.255.253.0	/24	254
255.255.254.0	/23	510
255.255.255.252	/30	2
255.255.255.128	/25	126

3

Before Subnetting: How Were IPs Being Wasted?

1. Efficient Use of IP Addresses

- IPv4 addresses are limited (only about 4.3 billion unique addresses).
- Without subnetting, large blocks of IP addresses would be wasted. For example:
- A small company with 10 devices might be assigned a Class A network (16 million IPs), which is highly inefficient.
- Subnetting allows networks to be divided into smaller, more manageable chunks, ensuring IP addresses are used efficiently.

2. Better Network Performance – Large networks can become slow because of too much traffic. Subnetting helps **reduce congestion** by breaking networks into smaller parts, so data doesn't have to travel everywhere.

Originally, when networks were assigned IP addresses, they were given in **fixed-size blocks called "Classes" (Class A, B, C, etc.)**. These classes had fixed numbers of IP addresses, even if a network didn't need all of them.

Here's the problem:

- **Class A** – Had **16.7 million IPs** (too many for most organizations).
- **Class B** – Had **65,536 IPs** (often too many for medium companies).
- **Class C** – Had **256 IPs** (too small for some companies).

Example of Wastage:

Let's say a company needed 2,000 IP addresses. They couldn't get exactly 2,000—they had to take a **Class B block (65,536 IPs)**, wasting more than **63,000 IPs!**

◆ CIDR Notation (Classless Inter-Domain Routing)

Instead of writing subnet masks like **255.255.255.0**, we use a shorter format called **CIDR notation**.

- **192.168.1.10/24** → means **24 bits** are for the **network**, and the rest are for **hosts**.

- **Examples of CIDR:**

CIDR	Subnet Mask	Hosts per Subnet
/24	255.255.255.0	256 (254 usable)
/25	255.255.255.128	128 (126 usable)
/26	255.255.255.192	64 (62 usable)

- **How to Read?**

  - `/24` → `11111111.11111111.11111111.00000000` (Network: 24 bits, Hosts: 8 bits)
  - `/25` → `11111111.11111111.11111111.10000000` (Network: 25 bits, Hosts: 7 bits)

Step 5: What is NAT (Network Address Translation)?

NAT is a technology that helps devices with **private IPs** communicate with the **public internet** using a **single public IP**.

NAT **keeps track of which private device requested what data**

💡 **Why is NAT needed?**

- **IPv4 addresses are limited** → Not enough public IPs for every device.
- **Private IPs can't access the internet directly** → They need NAT to convert to a public IP.
- **Security** → NAT hides internal devices from external threats.

🔥 How Does NAT Work?

Let's take an example of a **home network**:

1️⃣ **Your device (private IP) sends a request**

- Your phone (`192.168.1.100`) wants to visit Google.
- The request goes to the router.

2️⃣ **Router replaces private IP with public IP**

- The router has a **public IP** (`203.120.45.67`).
- It changes the source IP from **192.168.1.100** → **203.120.45.67**.

3️⃣ **Google replies to the router's public IP**

- Google doesn't see your private IP, only the public IP.
- It sends the response to **203.120.45.67**.

4️⃣ **Router sends data back to the right private IP**

- The router knows that **192.168.1.100** made the request.
- It forwards Google's response back to your phone.

● **This is how NAT allows multiple devices to use a single public IP!**

◆ Types of NAT

NAT comes in different types based on how it translates IPs.

■ 1️⃣ Static NAT (1-to-1 Mapping)

- Each **private IP is mapped to a specific public IP**.
- Used for **servers** that need a fixed public IP.
- Example:
  - Private IP: `192.168.1.10` → Public IP: `203.120.45.10`

■ 2️⃣ Dynamic NAT (Many-to-Many Mapping)

- The router picks a public IP **from a pool** of available addresses.
- Not every device gets the same public IP every time.
- Example:
  - Private IP: `192.168.1.20` → Public IP: `203.120.45.12` (this time)
  - Next time: `192.168.1.20` → Public IP: `203.120.45.14`

■ 3️⃣ PAT (Port Address Translation) – Many-to-One (Most Common!)

- **Multiple devices share a single public IP**, but different **port numbers** are assigned.
- Example:
  - Phone: `192.168.1.100:5000` → `203.120.45.67:5000`
  - Laptop: `192.168.1.101:5001` → `203.120.45.67:5001`
  - Both devices use **203.120.45.67**, but different **ports** help identify them.

● **PAT is used in almost every home and office network!**

Step 6: DHCP – Dynamic Host Configuration Protocol

🔥 What is DHCP?

DHCP is a **network service** that **automatically assigns IP addresses** to devices. It makes sure that every device gets a unique IP **without manual setup**.

💡 **Why is DHCP needed?**

- **Saves time** → No need to assign IPs manually.
- **Prevents conflicts** → Ensures no two devices get the same IP.
- **Manages networks easily** → Works for homes, offices, and large companies.

🔹 How DHCP Works (Step by Step)

1️⃣Device Requests an IP

- When a device (laptop, phone, etc.) joins a network, it asks for an IP.
- This request is called a **DHCP Discover** message.

2️⃣DHCP Server Offers an IP

- The DHCP server replies with an available IP address.
- This is a **DHCP Offer** message.

3️⃣Device Accepts the IP

- The device confirms and accepts the offered IP.
- This is a **DHCP Request** message.

4️⃣DHCP Server Confirms Assignment

- The DHCP server finalizes the process by sending a **DHCP Acknowledgment**.
- Now, the device can use the IP to communicate on the network.

◆ How Long Does an IP Last? (Lease Time)

DHCP assigns IPs for a limited time, called a **lease**.

- If a device **stays connected**, it renews the lease.
- If a device **disconnects**, the IP is returned to the pool for reuse.

Example:

- Your phone gets `192.168.1.100` for **24 hours**.
- After 24 hours, DHCP checks if your phone still needs it.
- If yes → IP is renewed.
- If no → IP is given to another device.

◆ Where is DHCP Used?

■ **Home Networks** – Your Wi-Fi router assigns IPs automatically.
■ **Offices & Schools** – IT teams use DHCP to manage thousands of devices.
■ **Public Wi-Fi** – Hotels, cafes, and airports assign temporary IPs via DHCP.

◆ What Happens If DHCP Fails?

If the DHCP server is down:
Devices won't get an IP and won't connect to the network.
Some devices may get a **self-assigned IP** (`169.254.x.x`), meaning **no internet**.

Step 7: Why Was IPv6 Introduced?

IPv6 was created because **IPv4 addresses were running out** due to:
■ More internet users
■ More devices (phones, smart TVs, IoT devices)
■ Large companies needing many IPs

IPv6 is 128bits

IPv6 fixes this by:
✔ Providing an **almost unlimited number of addresses**
✔ Eliminating the need for **NAT**
✔ Improving **security** with built-in encryption

What is NIC (Network Interface Card)?

A **Network Interface Card (NIC)** is a **hardware component** that allows a computer to connect to a network. It is responsible for sending and receiving data over the network.

💡 **Think of NIC as the "door" that lets your computer enter the network.**

◆ Types of NICs

1️⃣ **Wired NIC**

- Uses an **Ethernet cable** to connect to a network.
- Example: LAN ports on desktops and laptops.
- **Fast and stable** but requires cables.

2️⃣ **Wireless NIC**

- Uses **Wi-Fi** to connect without cables.
- Found in laptops, smartphones, and tablets.
- **Convenient** but can be affected by signal strength.

3️⃣ **Virtual NIC**

- Used in **virtual machines (VMs)** to simulate a network connection.
- No physical hardware—software-based connection.

◆ How Does NIC Work?

NIC acts as a **bridge** between the **computer and the network**:

1️⃣ **Receives Data** – Gets data from the computer and prepares it for the network.
2️⃣ **Encodes Data** – Converts data into network signals (electrical, optical, or radio).
3️⃣ **Sends Data** – Transmits data to another device using Ethernet or Wi-Fi.
4️⃣ **Receives Data from the Network** – Converts network signals back into usable data.

◆ NIC and MAC Address

Every NIC has a **unique identifier** called a **MAC (Media Access Control) address**.

- **MAC address is a permanent hardware address assigned to the NIC.**
- Example: `00:1A:2B:3C:4D:5E`
- **Used in local networks to identify devices uniquely.**

💡 **MAC address stays the same, while IP addresses can change.**

Step 8: MAC Address (Media Access Control Address)

A **MAC address** is a **unique** identifier assigned to every **NIC (Network Interface Card)**. It is used in a **local network** to identify devices.

■ **MAC addresses are permanent** (hardcoded into the NIC).
■ **IP addresses can change, but MAC addresses don't.**

📌 MAC Address Format

- A MAC address is **48 bits** (6 pairs of hexadecimal numbers).
- Example: **`00:1A:2B:3C:4D:5E`**
- The first **3 pairs** represent the **manufacturer** (OUI – Organizationally Unique Identifier). ← manufacturer ID (eg. Apple, intel)
- The last **3 pairs** are the device's unique ID.

💡 **Example:**

- `00:1A:2B` → Manufacturer (Intel, Realtek, etc.)
- `3C:4D:5E` → Unique identifier

📌 Where is the MAC Address Used?

■ Ethernet (LAN) & Wi-Fi (Wireless networks)
■ Data transmission within local networks (switches use MAC addresses to send data)
■ Security (Filtering, blocking unauthorized devices)

MAC is burned into hardware, but spoofable for Privacy or attack

Step 9: ARP (Address Resolution Protocol)

ARP is a **protocol** used to find the **MAC address** of a device when you only know its **IP address**.

💡 **Think of it like asking: "Hey, who owns this phone number?" and getting a name in return.**

🔹 How ARP Works (Step-by-Step)

🔹 Suppose your computer wants to send data to **192.168.1.10** but doesn't know its MAC address.

■ **Step 1: ARP Request** (Broadcast Message)
Your computer asks:
❝ Who has **192.168.1.10**? Tell me your MAC address! ❞

■ **Step 2: ARP Reply** (Unicast Message)
The device with **192.168.1.10** responds:
❝ My MAC address is **AA:BB:CC:DD:EE:FF**. ❞

■ **Step 3: Communication Begins**
Now that your computer knows the MAC address, it can send data! ●

🔹 Why is ARP Important?

Since **IP addresses change** (dynamic), ARP helps find the correct MAC address so devices can communicate **inside a network**.

🔹 ARP Security Risks 🔒

● **ARP Spoofing (ARP Poisoning)** – Attackers send fake ARP replies to **redirect traffic** or **steal data**.
■ **Solution:** Use **Dynamic ARP Inspection (DAI)** and **VPNs** to stay safe.

The reply is sent as an unicast directly to Host
(The one who asked)

↳ The router replies with a unicast ARP reply
giving its MAC

ARP works only in internal (local) networks (LANs), not on public IPs.

◆ Why ARP Works Only in Local Networks (LANs)?

- ARP is designed to find **MAC addresses**, and **MAC addresses only exist inside a local network**.
- The **internet (WAN)** does not use MAC addresses for communication; it uses **IP addresses only**.
- Routers separate networks, and **ARP cannot cross a router** to find a MAC address on another network.

◆ How Does It Work for Public IPs?

When communicating over the internet (using public IPs), devices use **routing tables** instead of ARP.

1️⃣ If you send data to a **public IP (like Google's 8.8.8.8)**, your device doesn't ask for its MAC address.
2️⃣ Instead, your router checks its routing table and **forwards** the data to the next hop (another router).
3️⃣ Only at the final destination (Google's data center), an **internal ARP process** may happen inside their local network.

◆ How ARP Converts IP to MAC (Step-by-Step)

1️⃣ Your computer wants to send data to **192.168.1.10** but doesn't know the MAC address.
2️⃣ It sends an **ARP Request** as a **broadcast message**:
❝ Who has IP **192.168.1.10**? Tell me your MAC address! ❞
3️⃣ The device with **192.168.1.10** replies with an **ARP Response**:
❝ My MAC address is **AA:BB:CC:DD:EE:FF**. ❞
4️⃣ Now your computer stores this MAC address in its **ARP cache** and starts communication.

◆ Final Summary

✔ **ARP = Finds MAC address using IP address.**
✔ **Used in local networks (LANs) to send data.**
✔ **Can be attacked using ARP spoofing, so security measures are needed.**

Step 10: Routing Table

A **routing table** is a database inside a **router** that helps it decide **where to send network packets**.

A **routing table** contains a list of **IP address destinations** and the best **path** to reach them. It helps the router **forward data** correctly.

💡 **Think of it like a GPS for the internet** – it tells the router which road (network path) to take to send data to its destination.

How a Router Delivers Response to the Correct Device

1️⃣ Your Device Sends a Request

✔ Your laptop (**192.168.1.10**) sends a request to **Google (142.250.182.4)**.
✔ Since **192.168.x.x** is a **private IP**, it **cannot travel over the internet** directly.
✔ Your **router (192.168.1.1) changes** the source IP to your **public IP (e.g., 203.0.113.5)** using **NAT (Network Address Translation)**.

🔍 Example Before NAT (Your Device's Request):

Source IP	Destination IP	Source Port	Destination Port
192.168.1.10	142.250.182.4 (Google)	12345	443 (HTTPS)

🔍 Example After NAT (Router Changes Source IP):

Source IP	Destination IP	Source Port	Destination Port
203.0.113.5 (Router's Public IP)	142.250.182.4	50000 (Random Port)	443

✔ The router **changes the source IP** but **keeps track of the original request** using a **NAT Table**.

2️⃣ Google Sends a Response to Your Public IP

✔ Google **receives your request** and sends a response **back to your public IP (203.0.113.5)**.
✔ **The response contains the same port number (50000) that was assigned by the router**.

🔍 **Google's Response Packet (Sent to Your Router's Public IP):**

Source IP	Destination IP	Source Port	Destination Port
142.250.182.4 (Google)	203.0.113.5 (Router)	443	50000

3 Router Checks NAT Table & Forwards to Correct Device

✔ The **router receives Google's response** and checks its **NAT Table**:

🔍 **Router's NAT Table (Before Forwarding Response)**

Public IP	Public Port	Private IP	Private Port
`203.0.113.5`	`50000`	`192.168.1.10`	`12345`

✔ **The router finds the matching entry and sees that 50000 belongs to 192.168.1.10.**
✔ **It changes the destination IP from 203.0.113.5 to 192.168.1.10 and forwards the packet.**

🔍 **After NAT (Router Delivers to Your Device):**

Source IP	Destination IP	Source Port	Destination Port
`142.250.182.4` (Google)	`192.168.1.10`	`443`	`12345`

✔ **Your laptop receives the response, and the browser displays the webpage!** 🎉

◆ Final Thoughts

✔ **NAT keeps track of which device made which request.**
✔ **Port numbers help the router know which response belongs to which request.**
✔ **Without NAT, your home network couldn't use a single public IP for multiple devices.**

Ethernet came first (1970s), then IP was added later (1980s).

early networks were just local-computers connected by cables in the same building. They only needed MAC address to identify each other locally.

1973: ethernet invented (MAC addresses)

1981: IP protocol created (IP addresses)

An IPv6 is a 128-bit address, which means it consists of 128 binary digits (bits).

These 128 bits are divided into 8 groups of 16 bits, each group is represented as four hexadecimal digits (0-9 & A-F).

This is written in hexadecimal format

2001:0db8:85a3:0000:0000:8a2e:0370:7334

each part is called hextet (also known as segment), because it represents 16 bits (a total of 8 hextets = 128 bits)

* Hex helps shorten the long binary string into something readable. each block is 4 hex digits, which is 16 bits.

* IPv6 has 8 blocks = 8 x16 bits = 128 bits total

* writing 128 bits in decimal or binary would be super long

* Hexadecimal is compact - 1 hex digit = 4 bits

* IPv6 uses hextets not octets like IPv4

* each hextet is shown in hexadecimal -> value range : 0000 - FFFF (which equals decimals 0-65535)

IPv6 are called compression. There are rules to shorten it & making it easier to read.

Rule 1: Removing leading zeros in each block

0db8 -> db8
0370 -> 370
0001 -> 1

Rule 2 - Replace longest sequence of :0: blocks with ::

You can replace one continuous stretch of :0000: blocks with a double colon :: - but only once per address!

= 2001:db8:0:0:0:0:8a2e:370
"
2001:db8::8a2e:370
↳ That is short IPv6 format

* You can only use :: once in an address
* if there are two or more sets of zeros, only the longest one is replaced

↳ Special case :- ::1 - loopback IP in IPv6 just like 127.0.01 in IPv4

= 0000:0000:0000:0000:0000:0000:0000:0001

* Mac format is hexadecimal
* divided into 6 groups of 8 bits (1 byte each)
* Displayed as 12 hexadecimal digits

00:1A:3C:2B:4D:5E

or

00-1A-3C-4D-2B-5E

* Shown as 2 hexadecimal digits
* Hex digits range from 00 to FF (0 to 255 in decimal)

6 × 8 bits = 48 bits total

= 12 hex digits = 6 bytes

* BOTH IPv4 & MAC are based on 8-bit chunks (octets / bytes)
* IPv4 uses 4 octets (32 bits)
* MAC uses 6 octets (48 bits)

=> The difference is:

* IPv4 is for logical IP addressing on network
* MAC is for Physical hardware identification

=> factory-assigned MAC stored in its firmware (EEPROM).