1. Difference between HTTP1.1 vs HTTP2

HTTP stands for hypertext transfer protocol & it is used in client-server communication. By using

HTTP user sends the request to the server & the server sends the response to the user. There are several stages of development of HTTP but we will focus mainly on HTTP/1.1 which was created in 1997 & the new one is HTTP/2 which was created in 2015.

**HTTP/1.1:** For better understanding, let’s assume the situation when you make a request to the server for the geeksforgeeks.html page & server responds to you as a resource geeksforgeeks.html page. before sending the request and the response there is a TCP connection established between client & server. again you make a request to the server for image img.jpg & the server gives a response as an image img.jpg. the connection was not lost here after the first request because we add a keep-alive header which is the part of the request so there is an open connection between the server & client. there is a persistent connection which means several requests & responses are merged in a single connection. These are the drawbacks that lead to the creation of HTTP/2: The first problem is HTTP/1.1 transfer all the requests & responses in the plain text message form. The second one is head of line blocking in which TCP connection is blocked all other requests until the response does not receive. all the information related to the header file is repeated in every request.

**HTTP/2:** HTTP/2 was developed over the SPDY protocol. HTTP/2 works on the binary framing layer instead of textual that converts all the messages in binary format. it works on fully multiplexed that is one TCP connection is used for multiple requests. HTTP/2 uses HPACK which is used to split data from header. it compresses the header. The server sends all the other files like CSS & JS without the request of the client using the PUSH frame.

|  |  |
| --- | --- |
| **HTTP/1.1** | **HTTP/2** |
| It works on the textual format. | |  |  | | --- | --- | |  | It works on the binary protocol. | |
| There is head of line blocking that blocks all the requests behind it until it doesn’t get its all resources. | It allows multiplexing so one TCP connection is required for multiple requests. |
| It uses requests resource Inlining for use getting multiple pages | It uses PUSH frame by server that collects all multiple pages |
| It compresses data by itself. | |  |  | | --- | --- | |  | It uses HPACK for data compression. | |

1. a blog about objects and its internal representation in Javascript

Objects, in JavaScript, is it’s most important data-type and forms the building blocks for modern JavaScript. These objects are quite different from JavaScript’s primitive data-types(Number, String, Boolean, null, undefined and symbol) in the sense that while these primitive data-types all store a single value each (depending on their types).

Objects are more complex and each object may contain any combination of these primitive data-types as well as reference data-types.  
An object, is a reference data type. Variables that are assigned a reference value are given a reference or a pointer to that value. That reference or pointer points to the location in memory where the object is stored. The variables don’t actually store the value.

Loosely speaking, objects in JavaScript may be defined as an unordered collection of related data, of primitive or reference types, in the form of “key: value” pairs. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object.

For Eg. If your object is a student, it will have properties like name, age, address, id, etc and methods like updateAddress, updateNam, etc.

**Objects and properties**

A JavaScript object has properties associated with it. A property of an object can be explained as a variable that is attached to the object. Object properties are basically the same as ordinary JavaScript variables, except for the attachment to objects. The properties of an object define the characteristics of the object. You access the properties of an object with a simple dot-notation:

objectName.propertyName

Like all JavaScript variables, both the object name (which could be a normal variable) and property name are case sensitive. You can define a property by assigning it a value. For example, let’s create an object named myCar and give it properties named make, model, and year as follows:

var myCar = new Object();  
myCar.make = 'Ford';  
myCar.model = 'Mustang';  
myCar.year = 1969;

Unassigned properties of an object are [undefined](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/undefined) (and not [null](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/null)).

myCar.color; // undefined

Properties of JavaScript objects can also be accessed or set using a bracket notation (for more details see [property accessors](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Property_Accessors)). Objects are sometimes called *associative arrays*, since each property is associated with a string value that can be used to access it. So, for example, you could access the properties of the myCar object as follows:

myCar['make'] = 'Ford';  
myCar['model'] = 'Mustang';  
myCar['year'] = 1969;

An object property name can be any valid JavaScript string, or anything that can be converted to a string, including the empty string. However, any property name that is not a valid JavaScript identifier (for example, a property name that has a space or a hyphen, or that starts with a number) can only be accessed using the square bracket notation. This notation is also very useful when property names are to be dynamically determined (when the property name is not determined until runtime). Examples are as follows:

// four variables are created and assigned in a single go,   
// separated by commas  
var myObj = new Object(),  
 str = 'myString',  
 rand = Math.random(),  
 obj = new Object();  
myObj.type = 'Dot syntax';  
myObj['date created'] = 'String with space';  
myObj[str] = 'String value';  
myObj[rand] = 'Random Number';  
myObj[obj] = 'Object';  
myObj[''] = 'Even an empty string';console.log(myObj);

You can also access properties by using a string value that is stored in a variable:

var propertyName = 'make';  
myCar[propertyName] = 'Ford';propertyName = 'model';  
myCar[propertyName] = 'Mustang';

You can use the bracket notation with [for...in](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in) to iterate over all the enumerable properties of an object. To illustrate how this works, the following function displays the properties of the object when you pass the object and the object's name as arguments to the function:

function showProps(obj, objName) {  
 var result = ``;  
 for (var i in obj) {  
 // obj.hasOwnProperty() is used to filter out properties from the object's prototype chain  
 if (obj.hasOwnProperty(i)) {  
 result += `${objName}.${i} = ${obj[i]}\n`;  
 }  
 }  
 return result;  
}

So, the function call showProps(myCar, "myCar") would return the following:

myCar.make = Ford  
myCar.model = Mustang  
myCar.year = 1969

**Creating Objects In JavaScript :**

**Create JavaScript Object with Object Literal**

One of easiest way to create a javascript object is object literal, simply define the property and values inside curly braces as shown below

let bike = {name: 'SuperSport', maker:'Ducati', engine:'937cc'};

**Create JavaScript Object with Constructor**

Constructor is nothing but a function and with help of new keyword, constructor function allows to create multiple objects of same flavor as shown below

function Vehicle(name, maker) {  
 this.name = name;  
 this.maker = maker;  
}  
let car1 = new Vehicle(’Fiesta’, 'Ford’);  
let car2 = new Vehicle(’Santa Fe’, 'Hyundai’)  
console.log(car1.name); //Output: Fiesta  
console.log(car2.name); //Output: Santa Fe

**Using the JavaScript Keyword new**

The following example also creates a new JavaScript object with four properties:

Example

var person = new Object();  
person.firstName = “John”;  
person.lastName = “Doe”;  
person.age = 50;  
person.eyeColor = “blue”;

**Using the Object.create method**

Objects can also be created using the [Object.create()](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/create) method. This method can be very useful, because it allows you to choose the prototype object for the object you want to create, without having to define a constructor function.

// Animal properties and method encapsulation  
var Animal = {  
 type: 'Invertebrates', // Default value of properties  
 displayType: function() { // Method which will display type of Animal  
 console.log(this.type);  
 }  
};  
// Create new animal type called

animal1   
var animal1 = Object.create(Animal);  
animal1.displayType(); // Output:Invertebrates  
// Create new animal type called Fishes  
var fish = Object.create(Animal);  
fish.type = 'Fishes';  
fish.displayType(); // Output:Fishes

1. Ip Address

## What is an IP address?

IP address stands for “Internet Protocol address.” The Internet Protocol is a set of rules for communication over the internet, such as sending mail, streaming video, or connecting to a website. **An IP address identifies a network or device on the internet.**

The internet protocols manage the process of assigning each unique device its own IP address. (Internet protocols do other things as well, such as routing internet traffic.) This way, it’s easy to see which devices on the internet are sending, requesting, and receiving what information.

IP addresses are like telephone numbers, and they serve the same purpose. When you contact someone, your phone number identifies who you are, and it assures the person who answers the phone that you are who you say you are. IP addresses do the exact same thing when you’re online — that’s why **every single device that is connected to the internet has an IP address**.

There are two types of IP addresses: [IPv4 and IPv6](https://www.avast.com/c-ipv4-vs-ipv6-addresses). It’s easy to recognize the difference if you count the numbers. IPv4 addresses contain a series of four numbers, ranging from 0 (except the first one) to 255, each separated from the next by a period — such as 5.62.42.77.

IPv6 addresses are represented as eight groups of four hexadecimal digits, with the groups separated by colons. A typical IPv6 address might look like this: 2620:0aba2:0d01:2042:0100:8c4d:d370:72b4.

You can easily find your IP address. In fact, we have a cheat sheet that shows [how to find your IP address on Mac or Windows](https://www.avast.com/c-how-to-find-ip-address).

### The parts of your IP address

An IP address has two parts: the **network ID**, comprising the first three numbers of the address, and a **host ID**, the fourth number in the address. So on your home network — 192.168.1.1, for example – 192.168.1 is the network ID, and the final number is the host ID.

The Network ID indicates which network the device is on. The Host ID refers to the specific device on that network. (Usually your router is .1, and each subsequent device gets assigned .2, .3, and so on.)

You may not always want the outside world to know exactly which device and network you're using. In this case, it’s possible to mask your IP address from the outside world through a [Virtual Private Network (VPN)](https://www.avast.com/c-what-is-a-vpn). When you use a VPN, it prevents your network from revealing your address.

#### Where do IP addresses come from?

IPv4 dates back to the early 1980s, when the internet was a private network for the military. IPv4 has a total pool of 4.3 billion addresses, which sounds like a lot. But with all the computers, smartphones, and tablets that connect to the internet, not to mention [IoT devices](https://www.avast.com/c-what-is-the-internet-of-things), we have run out of IPv4 addresses. In fact, we began running out in the 1990s. Very clever technical networking tricks have kept things going.

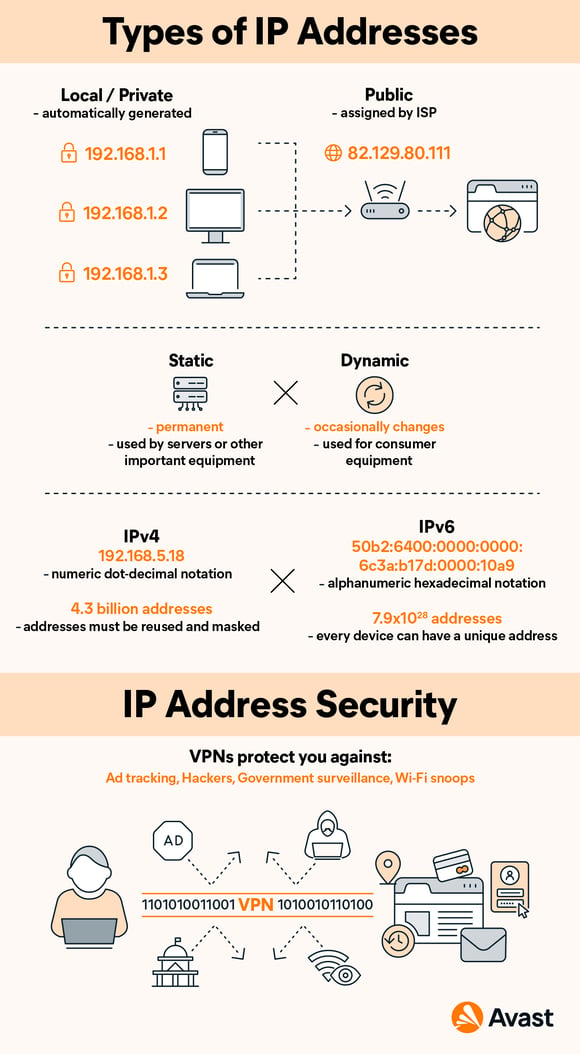
The Internet Engineering Task Force (IETF), which designs the backbone technologies of the internet, came up with IPv6 about a decade ago. It has a potential pool of 340 undecillion addresses — that’s the number 340 followed by 36 zeroes — meaning we can (in theory) never run out of addresses. It is slowly replacing IPv4, but for now, the two co-exist.

### Public vs. local IP addresses

There are two types of IP addresses: [external, or public IP addresses; and internal, also called local or private addresses](https://www.avast.com/c-ip-address-public-vs-private). Your internet service provider (ISP) gives you your external address. When you surf the web, the site you're visiting needs to know who you are (for traffic-monitoring reasons). Your ISP uses your external IP address to introduce you to the website and establish the connection.

You have a different IP address for internal purposes, such as identifying your devices within a home network or inside a business office. The local or internal IP address is assigned to your computer by the router, which is the hardware that connects a local network to the internet. In most cases, that internal IP address is assigned automatically by the router (or cable modem).

Here’s what matters: In most cases, you'll have a different IP address internally than you do on the public internet. Your local IP address represents your device on its network, and your public IP address is the face of your network to the greater internet.



## How do IP addresses work?

The post office uses your physical address as a marker for the real-world location of a person, residence, or business. It’s how mail is routed. It’s where you reside. It’s how others know where to find you.

All of these descriptions apply to an IP address, in a digital way. An IP address is where a computer resides, in a virtual sense. IP addresses may identify your own computer, a favorite website, a network server, or even a device (such as a webcam).

IP addresses are especially important for sending and receiving information. They route internet traffic where it needs to go, and they direct emails to your inbox.

The important thing to remember is this: **Every active device on the internet has an IP address.**

### First, TCP/IP...

IP addresses are only one part of the internet’s architecture. After all, having a postal address for your house is meaningless unless there’s a post office responsible for delivering the mail. In internet terms, IP is one part of TCP/IP.

The Transmission Control Protocol/Internet Protocol (TCP/IP) is a set of rules and procedures for connecting devices across the internet. [TCP/IP specifies how data is exchanged](https://www.avast.com/c-what-is-tcp-ip): **Data is broken down into packets and passed along a chain of routers from origin to destination.** This is the basis for all internet communication.

TCP defines how applications communicate across the network. It manages how a message is broken down into a series of smaller packets, which are then transmitted over the internet and reassembled in the right order at the destination address.

The IP portion of the protocol directs each packet to the right destination. Each gateway computer on the network checks this IP address to determine where to forward the message.

### How IP addresses are assigned: dynamic vs. static

IP addresses can be permanent (static) or temporary (dynamic). [The difference between static and dynamic IP addresses](https://www.avast.com/c-static-vs-dynamic-ip-addresses) is that while the former never change, the latter can and do.

Static addresses are mostly used by businesses, since their websites and web applications must be reliably accessible at all times. But your home IP address doesn’t have to stay the same, since it’s only needed when you’re using the internet.

Your ISP will usually give you a dynamic IP address. While your IP address may not change often, it’s possible to receive a new one from your ISP every time you reboot your computer. The same holds true with the local IP addresses your home wireless router assigns to your laptop, tablet, or smartphone. These devices might get a new address every time you restart your router.

The only real negative with dynamic addresses is that a given computer can’t be reliably found. That makes it difficult to, say, run a web server in your home, as the address might change and no one would be able to find you. Many ISPs allow you to make arrangements for a business connection with a static address if you want to run a server.

### A packet's journey

Every time you visit a website, your trip is routed through a complex and hidden series of hops through major traffic backbones. It’s akin to an integrated highway system. A visit to a website might involve a dozen hops, all of which take place near-instantaneously and behind the scenes.

Whether you’re sending an email, loading a webpage or watching a video, **all data sent over the internet is broken up into packets**. Think of it like a [bucket brigade](https://www.dictionary.com/browse/bucket-brigade), where buckets of water are passed down a line of people.

Best of all, you usually don’t need to know how it works. One of the coolest things about the internet’s design is that it keeps this structure wholly invisible to ordinary users. This way, you're free to focus on what you need to do, without worrying about how it gets done.

Each packet has a maximum size of 1,500 bytes and includes a wrapper with a header and a footer. The information in the wrapper communicates the type of data in the packet, where it came from, where it’s going, and how it fits together with other packets.

As packets travel, they move in a stream, but all the packets don’t always take the exact same path. If there is congestion across the internet, various packets from the same message might travel across different network backbones. As the packets arrive, the receiving computer reassembles the packets into the original (and final) data.

That’s one reason that IP addresses are so important. Because every packet includes the IP address for its origin and destination, the internet can make sure that all packets reach the right destination.

### DNS servers

The Domain Name System (DNS) makes the modern internet possible. The DNS pairs website names you can easily remember with IP addresses that computers can use.

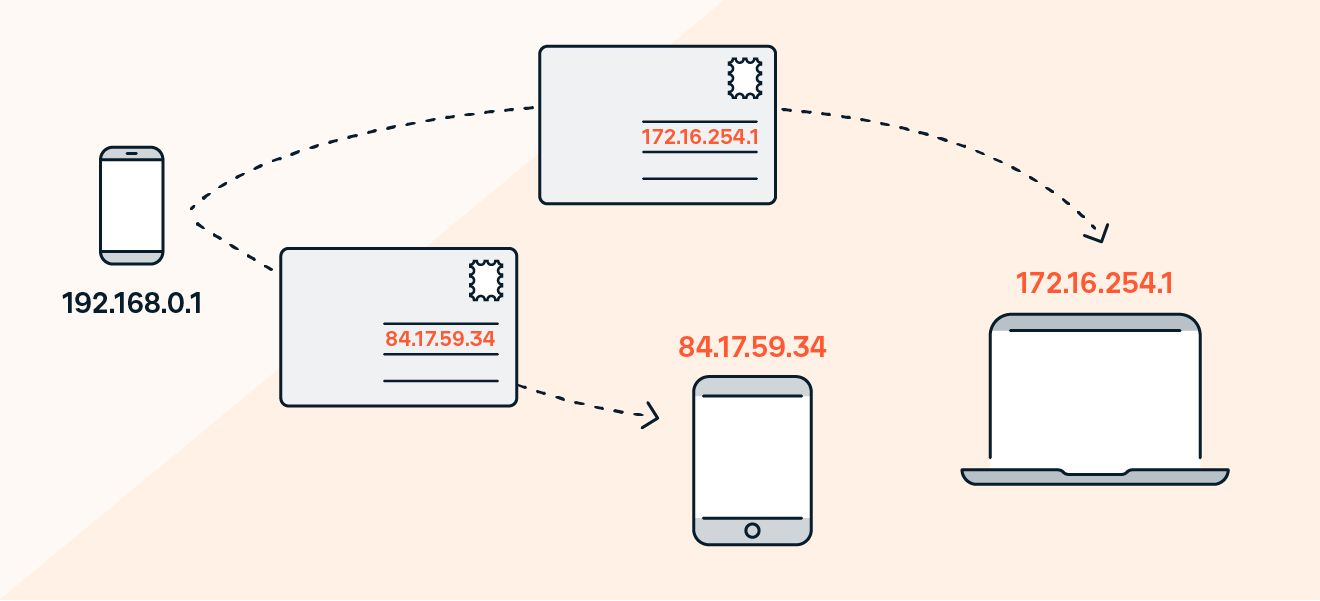
IPv4 addresses are every bit as valid as a website’s alphabetical name. You can type **157.240.22.35** into your web browser, and it’ll take you to Facebook. But who’s going to remember that? Most of us barely remember our own phone numbers.

Familiar domain names are substitutes for real IP addresses. In the early days of the internet, you could connect to another site only by typing in the numbers of its IP address. But thanks to the DNS, we don't need to do this anymore.

The DNS freed us from the headache of remembering IP addresses by giving each site a name. DNS servers sit between your browser and the site you want to visit. When you type a URL into your browser, your browser looks up that domain name in the DNS server, retrieves the corresponding IP address, and sends you to the site you want. All without you, the user, needing to do a thing.

## What is the purpose of an IP address?

**The purpose of an IP address is to handle the connection between devices that send and receive information across a network**. The IP address uniquely identifies every device on the internet; without one, there’s no way to contact them. IP addresses allow computing devices (such as PCs and tablets) to communicate with destinations like websites and streaming services, and they let websites know who is connecting.



An IP address also acts like a return address on postal mail. When a letter you've mailed is delivered to the wrong address, you get the letter back if you include a return address on the envelope.

The same holds true for email. When you write to an invalid recipient (such as someone who left their job and no longer has a company email address) your IP address lets the company’s mail server send you back a bounce message so that you know your email wasn't sent to the right place.

### IP addresses and geolocation

In addition to making sure other computers on the internet can communicate with yours, IP addresses also mark the real-world location of your device. This comes in handy when a website wants to customize itself to suit your needs — such as by automatically changing its language, or showing you products that are available in your country. Streaming platforms can also use your IP address to show you the content that they’re allowed to provide you based on where you live.

While your IP address won’t give away your precise location, it can get pretty close. Your IP address can be used to identify your city, postal code, internet service provider (ISP), and latitude and longitude. **Since you may not want everyone in the world knowing roughly where you are, consider hiding your IP address with a VPN.**

Rather than broadcasting your real IP address to the world, a VPN, like [Avast SecureLine VPN](https://www.avast.com/secureline-vpn), masks your IP address behind another one. This way, anyone else on the internet will only see your VPN’s IP address, while yours remains protected.

## How to find your IP address?

Whether you’re using Windows or macOS, you can easily find your IP address within your computer’s network settings. On Windows 10, visit your Network & Internet settings from the Settings menu. If you’re using a Mac, you’ll want to visit the Network section of your System Preferences. Alternatively, simply visit Avast's [IP address checker](https://www.avast.com/what-is-my-ip) to find your public IP address.  
  
Check out our [comprehensive guide to finding your IP address](https://www.avast.com/c-how-to-find-ip-address) for complete directions on all Windows versions as well as on macOS.

## IP address security

You should protect your IP address for all the reasons you would protect your home address. [Cybercriminals](https://www.avast.com/c-cybercrime) may attempt to take advantage of you for a variety of reasons. With your IP address, they might be able to:

* **Download or access illegal content:** Pirates can download movies, music, and other content and appear to be you, so you could get in trouble with your ISP for something you didn’t do.
* **Trace your home address:** It is possible to trace an IP address to a real-world address, exposing you to potential physical danger or robbery when you are not home. Security experts have long advised people to never say on social media that you are on vacation, because that tells criminals you’re not home. It’s a good idea to [follow that advice for privacy reasons](https://blog.avast.com/online-privacy-this-is-a-conversation-we-all-need-to-have).
* **Spy on your private internet traffic:** Your sensitive personal data, including financial information, can be at risk when [hackers](https://www.avast.com/c-hacker) are able to access your IP traffic.
* **Directly attack you**: In a [DDoS (distributed denial-of-service) attack](https://www.avast.com/c-ddos), cybercriminals flood a server with so much traffic that it becomes overwhelmed and shuts down. Consider adding [privacy tools](https://www.avast.com/c-category-privacy) to your online toolset, including one that [hides your IP address](https://www.avast.com/c-hide-my-ip-address).

## Protect your IP address

The Consumer Technology Association recommends that you protect your IP address, such as by using a virtual private network (VPN) to hide it from cybercriminals seeking to commit identity theft or fraud.

[Avast SecureLine VPN](https://www.avast.com/secureline-vpn) can protect your IP address from anyone attempting to spy on your IP traffic to capture sensitive financial or personal data. With Avast’s bank-grade encryption, which masks your IP address and sends your communications through a secure tunnel, you’ll enjoy a safe and private internet experience every time you log on.

1. port

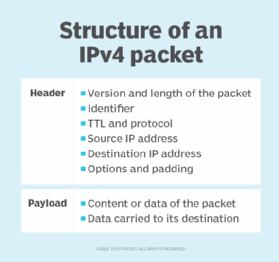
### What is a port number?

A port number is a way to identify a specific process to which an internet or other network message is to be forwarded when it arrives at a [server](https://www.techtarget.com/whatis/definition/server). All network-connected devices come equipped with standardized ports that have an assigned number. These numbers are reserved for certain [protocols](https://www.techtarget.com/searchnetworking/definition/protocol) and their associated function. Hypertext Transfer Protocol ([HTTP](https://www.techtarget.com/whatis/definition/HTTP-Hypertext-Transfer-Protocol)) messages, for example, always go to [port 80](https://www.techtarget.com/searchnetworking/definition/port-80) -- one of the most commonly used ports.

Developers of the [Advanced Research Projects Agency Network](https://www.techtarget.com/searchnetworking/definition/ARPANET), an informal cooperation of system administrators and software authors, proposed the concept of port numbers. Once known as [*socket numbers*](https://www.techtarget.com/searchnetworking/answer/What-is-the-difference-between-a-port-and-a-socket), the early incarnation of port numbers is similar to the Internet Protocol ([IP](https://www.techtarget.com/searchunifiedcommunications/definition/Internet-Protocol)) address class used today.

### What is the difference between an IP address and a port number?

An [IP address](https://www.techtarget.com/whatis/definition/IP-address-Internet-Protocol-Address) identifies a machine in an IP network and is used to determine the destination of a data [packet](https://www.techtarget.com/searchnetworking/definition/packet). Port numbers identify a particular application or service on a system.

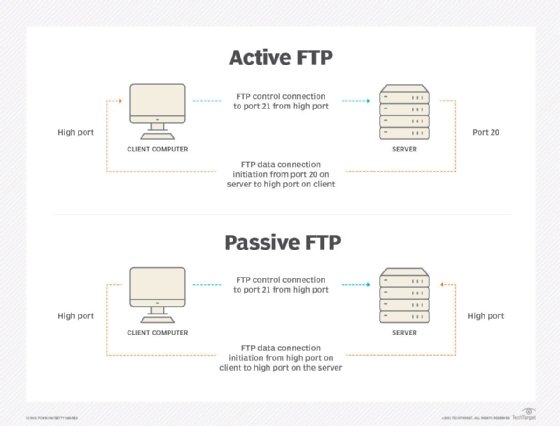
An IP address identifies a machine in an IP network and determines the destination of a data packet, while port numbers identify particular applications or services on a system.

An IP address is a logical address used to identify a device on the network. Any device connected to the internet is assigned a unique IP address for identification. This identifying information enables devices to communicate over the internet.

Port numbers are part of the addressing information that helps identify senders and receivers of information and a particular application on the devices. Port numbers consist of 16-[bit](https://www.techtarget.com/whatis/definition/bit-binary-digit) numbers.

For example, a user request for a file transfer from a [client](https://www.techtarget.com/searchenterprisedesktop/definition/client), or local host, to a remote server on the internet uses File Transfer Protocol ([FTP](https://www.techtarget.com/searchnetworking/definition/File-Transfer-Protocol-FTP)) for the transaction. Both devices must be configured to [transfer files via FTP](https://www.techtarget.com/searchnetworking/tip/Understanding-the-FTP-PORT-command). To transfer the file, the Transmission Control Protocol ([TCP](https://www.techtarget.com/searchnetworking/definition/TCP)) software layer in local host identifies the port number of 21, which, by convention, associates with an FTP request -- in the 16-bit port number integer that is appended to the request.

At the server, the TCP layer will read port number 21 and forward the request to the FTP program at the server.

FTP uses ports 20 and 21 to transfer files between a client and a server.

### What are the different types of port numbers and their uses?

There are 65,535 port numbers, but not all are used every day.

Restricted port numbers or well-known port numbers are reserved by prominent companies and range from 0 to 1023. [Apple QuickTime](https://www.techtarget.com/whatis/definition/Quicktime), [Structured Query Language](https://www.techtarget.com/searchdatamanagement/definition/SQL) services and [Gopher](https://www.techtarget.com/whatis/definition/Gopher) services use some of these restricted ports.

Those who want to register a specific port number can choose from 1024 to 49151. Software companies typically register these port numbers. [Dynamic](https://www.techtarget.com/searchnetworking/definition/dynamic-port-numbers) or private ports ranging from 49152 to 65536 are available for anyone to use.

In another scenario, a port number is assigned temporarily -- for the duration of the request and its completion -- from a range of assigned port numbers. This is called a temporary port number.

Here are some commonly used ports and their associated networking protocols:

* **Ports 20 and 21.** FTP is used to transfer files between a client and a server.
* **Port 22.** [Secure Shell](https://www.techtarget.com/searchsecurity/definition/Secure-Shell) is one of several tunneling protocols used to build secure network connections.
* **Port 25.** Simple Mail Transfer Protocol ([SMTP](https://www.techtarget.com/whatis/definition/SMTP-Simple-Mail-Transfer-Protocol)) is commonly used for email.
* **Port 53.** Domain name system ([DNS](https://www.techtarget.com/searchnetworking/definition/domain-name-system)) is a critical process that matches human-readable domain names to machine-readable IP addresses on the modern internet. It helps users load websites and applications without typing in a long list of IP addresses.
* **Port 80.** HTTP is the protocol that enables the [World Wide Web](https://www.techtarget.com/whatis/definition/World-Wide-Web).
* **Port 123.** [Network Time Protocol](https://www.techtarget.com/searchnetworking/definition/Network-Time-Protocol) helps computer clocks sync with each other. It's a vital process in [encryption](https://www.techtarget.com/searchsecurity/definition/encryption)
* **Port 179.** Border Gateway Protocol ([BGP](https://www.techtarget.com/searchnetworking/definition/BGP-Border-Gateway-Protocol)) helps establish efficient routes between the large networks or autonomous systems that make up the internet. These large networks use BGP to broadcast which IP addresses they control.
* **Port 443.** HTTP Secure ([HTTPS](https://www.techtarget.com/searchsoftwarequality/definition/HTTPS)) is like HTTP but more [secure](https://www.techtarget.com/searchsecurity/tip/How-to-encrypt-and-secure-a-website-using-HTTPS). All HTTPS web traffic goes straight to port 443. Any network service that uses HTTPS for encryption, such as DNS over HTTPS, also connects directly to this port.
* **Port 500.** [Internet Security Association and Key Management Protocol](https://www.oreilly.com/library/view/the-ims-ip/9780470019061/9780470019061_internet_security_association_and_key_ma.html) helps set up secure [IP Security](https://www.techtarget.com/searchsecurity/definition/IPsec-Internet-Protocol-Security)
* **Port 3389.** [Remote Desktop Protocol](https://www.techtarget.com/searchenterprisedesktop/definition/Remote-Desktop-Protocol-RDP) enables users to connect to their desktop computers from another device remotely.

The [Internet Assigned Numbers Authority](https://www.techtarget.com/whatis/definition/IANA-Internet-Assigned-Numbers-Authority) allocates and maintains all the port numbers listed above.

### What are common questions raised about port numbers?

#### What is the port number for localhost?

Localhost is the default name used to establish a connection with a computer. The IP address is usually 127.0.0.1. This is done by using a loopback address network. Port 80 is the common standard port for HTTP.

#### What is port number 8080 used for?

Port number 8080 is usually used for [web servers](https://www.techtarget.com/whatis/definition/Web-server). When a port number is added to the end of the domain name, it drives traffic to the web server. However, users can not reserve port 8080 for secondary web servers.

#### What is port number 3360 used for?

[TCP/IP](https://www.techtarget.com/searchnetworking/definition/TCP-IP) networks use port 3360. The connection-oriented protocol TCP demands handshaking to set up end-to-end communications. Upon establishing the connection, user data is transferred bidirectionally over the connection.

#### What is my IP address and port number?

The easiest way to find a [router](https://www.techtarget.com/searchnetworking/definition/router)'s public IP address is to search "what is my IP?" on a [search engine](https://www.techtarget.com/whatis/definition/search-engine) like Google. Identifying a port number will depend on the operating system.

For Windows:

* Go to the command prompt.
* Type ipconfig.
* Then, type netstat to populate a list of all the port numbers.

For macOS:

* Go to System Preferences.
* Next, go to Network > Advanced.
* Click on the Port Scan tab, and enter the user's IP address.

#### What is a proxy server address and port number?

A [proxy server](https://www.techtarget.com/whatis/definition/proxy-server) is, essentially, a computer on the internet with its own IP address. It sits between the client device and the remote server and acts as an intermediary to handle communication requests over the internet.

When a local computer sends a web request, it automatically goes through the proxy server. The proxy server uses its own IP address for the web request and not the user's. Proxy servers offer privacy benefits -- for example, the ability to change the client IP address, masking the user's location.

The proxy server address includes an IP address with the port number attached to the end of the address. The port number 8080 is usually used for web servers, proxy and [caching](https://www.techtarget.com/whatis/definition/caching).

### What is the port number for Gmail?

Gmail uses both Internet Message Access Protocol ([IMAP](https://www.techtarget.com/whatis/definition/IMAP-Internet-Message-Access-Protocol)) and SMTP. The IMAP port is 993, and the SMTP port is 25.

1. HTTP methods
2. The set of common methods for HTTP/1.1 is defined below and this set can be expanded based on requirements. These method names are case sensitive and they must be used in uppercase.

|  |  |
| --- | --- |
| **S.N.** | **Method and Description** |
| 1 | **GET**  The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data. |
| 2 | **HEAD**  Same as GET, but transfers the status line and header section only. |
| 3 | **POST**  A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms. |
| 4 | **PUT**  Replaces all current representations of the target resource with the uploaded content. |
| 5 | **DELETE**  Removes all current representations of the target resource given by a URI. |
| 6 | **CONNECT**  Establishes a tunnel to the server identified by a given URI. |
| 7 | **OPTIONS**  Describes the communication options for the target resource. |
| 8 | **TRACE**  Performs a message loop-back test along the path to the target resource. |

## GET Method

1. A GET request retrieves data from a web server by specifying parameters in the URL portion of the request. This is the main method used for document retrieval. The following example makes use of GET method to fetch hello.htm:
2. GET /hello.htm HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. Host: www.tutorialspoint.com
5. Accept-Language: en-us
6. Accept-Encoding: gzip, deflate
7. Connection: Keep-Alive
8. The server response against the above GET request will be as follows:
9. HTTP/1.1 200 OK
10. Date: Mon, 27 Jul 2009 12:28:53 GMT
11. Server: Apache/2.2.14 (Win32)
12. Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT
13. ETag: "34aa387-d-1568eb00"
14. Vary: Authorization,Accept
15. Accept-Ranges: bytes
16. Content-Length: 88
17. Content-Type: text/html
18. Connection: Closed
19. <html>
20. <body>
21. <h1>Hello, World!</h1>
22. </body>
23. </html>

## HEAD Method

1. The HEAD method is functionally similar to GET, except that the server replies with a response line and headers, but no entity-body. The following example makes use of HEAD method to fetch header information about hello.htm:
2. HEAD /hello.htm HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. Host: www.tutorialspoint.com
5. Accept-Language: en-us
6. Accept-Encoding: gzip, deflate
7. Connection: Keep-Alive
8. The server response against the above HEAD request will be as follows:
9. HTTP/1.1 200 OK
10. Date: Mon, 27 Jul 2009 12:28:53 GMT
11. Server: Apache/2.2.14 (Win32)
12. Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT
13. ETag: "34aa387-d-1568eb00"
14. Vary: Authorization,Accept
15. Accept-Ranges: bytes
16. Content-Length: 88
17. Content-Type: text/html
18. Connection: Closed
19. You can notice that here server the does not send any data after header.

## POST Method

1. The POST method is used when you want to send some data to the server, for example, file update, form data, etc. The following example makes use of POST method to send a form data to the server, which will be processed by a process.cgi and finally a response will be returned:
2. POST /cgi-bin/process.cgi HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. Host: www.tutorialspoint.com
5. Content-Type: text/xml; charset=utf-8
6. Content-Length: 88
7. Accept-Language: en-us
8. Accept-Encoding: gzip, deflate
9. Connection: Keep-Alive
10. <?xml version="1.0" encoding="utf-8"?>
11. <string xmlns="http://clearforest.com/">string</string>
12. The server side script process.cgi processes the passed data and sends the following response:
13. HTTP/1.1 200 OK
14. Date: Mon, 27 Jul 2009 12:28:53 GMT
15. Server: Apache/2.2.14 (Win32)
16. Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT
17. ETag: "34aa387-d-1568eb00"
18. Vary: Authorization,Accept
19. Accept-Ranges: bytes
20. Content-Length: 88
21. Content-Type: text/html
22. Connection: Closed
23. <html>
24. <body>
25. <h1>Request Processed Successfully</h1>
26. </body>
27. </html>

## PUT Method

1. The PUT method is used to request the server to store the included entity-body at a location specified by the given URL. The following example requests the server to save the given entity-body in **hello.htm** at the root of the server:
2. PUT /hello.htm HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. Host: www.tutorialspoint.com
5. Accept-Language: en-us
6. Connection: Keep-Alive
7. Content-type: text/html
8. Content-Length: 182
9. <html>
10. <body>
11. <h1>Hello, World!</h1>
12. </body>
13. </html>
14. The server will store the given entity-body in **hello.htm** file and will send the following response back to the client:
15. HTTP/1.1 201 Created
16. Date: Mon, 27 Jul 2009 12:28:53 GMT
17. Server: Apache/2.2.14 (Win32)
18. Content-type: text/html
19. Content-length: 30
20. Connection: Closed
21. <html>
22. <body>
23. <h1>The file was created.</h1>
24. </body>
25. </html>

## DELETE Method

1. The DELETE method is used to request the server to delete a file at a location specified by the given URL. The following example requests the server to delete the given file **hello.htm** at the root of the server:
2. DELETE /hello.htm HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. Host: www.tutorialspoint.com
5. Accept-Language: en-us
6. Connection: Keep-Alive
7. The server will delete the mentioned file **hello.htm** and will send the following response back to the client:
8. HTTP/1.1 200 OK
9. Date: Mon, 27 Jul 2009 12:28:53 GMT
10. Server: Apache/2.2.14 (Win32)
11. Content-type: text/html
12. Content-length: 30
13. Connection: Closed
14. <html>
15. <body>
16. <h1>URL deleted.</h1>
17. </body>
18. </html>

## CONNECT Method

1. The CONNECT method is used by the client to establish a network connection to a web server over HTTP. The following example requests a connection with a web server running on the host tutorialspoint.com:
2. CONNECT www.tutorialspoint.com HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. The connection is established with the server and the following response is sent back to the client:
5. HTTP/1.1 200 Connection established
6. Date: Mon, 27 Jul 2009 12:28:53 GMT
7. Server: Apache/2.2.14 (Win32)

## OPTIONS Method

1. The OPTIONS method is used by the client to find out the HTTP methods and other options supported by a web server. The client can specify a URL for the OPTIONS method, or an asterisk (\*) to refer to the entire server. The following example requests a list of methods supported by a web server running on tutorialspoint.com:
2. OPTIONS \* HTTP/1.1
3. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
4. The server will send an information based on the current configuration of the server, for example:
5. HTTP/1.1 200 OK
6. Date: Mon, 27 Jul 2009 12:28:53 GMT
7. Server: Apache/2.2.14 (Win32)
8. Allow: GET,HEAD,POST,OPTIONS,TRACE
9. Content-Type: httpd/unix-directory

## TRACE Method

1. The TRACE method is used to echo the contents of an HTTP Request back to the requester which can be used for debugging purpose at the time of development. The following example shows the usage of TRACE method:
2. TRACE / HTTP/1.1
3. Host: www.tutorialspoint.com
4. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
5. The server will send the following message in response to the above request:
6. HTTP/1.1 200 OK
7. Date: Mon, 27 Jul 2009 12:28:53 GMT
8. Server: Apache/2.2.14 (Win32)
9. Connection: close
10. Content-Type: message/http
11. Content-Length: 39
12. TRACE / HTTP/1.1
13. Host: www.tutorialspoint.com
14. User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)

MAC ADDRESS

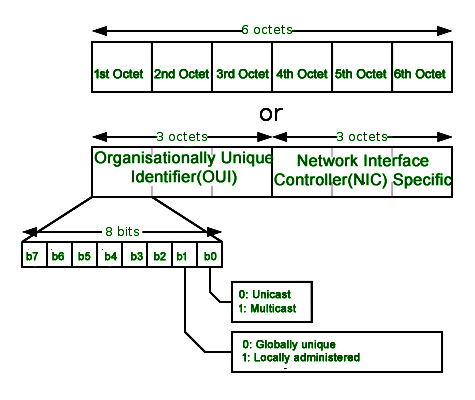
To communicate or transfer data from one computer to another, we need an address. In Computer networks, various types of addresses are introduced; each works at a different layer. [A MAC address](https://www.geeksforgeeks.org/mac-address-in-computer-network/), which stands for Media Access Control Address, is a physical address that works at the Data Link Layer. In this article, we will discuss addressing a DLL, which is MAC Address.

## MAC (Media Access Control) Address

MAC Addresses are unique **48-bit** hardware number of a computer, that is embedded into a network card (known as a [**Network Interface Card**](https://www.geeksforgeeks.org/nic-full-form/) during manufacturing. The MAC Address is also known as the [**Physical Address**](https://www.geeksforgeeks.org/logical-and-physical-address-in-operating-system/) of a network device. In IEEE 802 standard, the data link layer is divided into two sublayers:

1. Logical Link Control (LLC) Sublayer
2. Media Access Control (MAC) Sublayer

**The MAC** address is used by the Media Access Control (MAC) sublayer of the [Data-Link Layer](https://www.geeksforgeeks.org/data-link-layer/). MAC Address is worldwide unique since millions of network devices exist and we need to uniquely identify each.



## Format of MAC Address

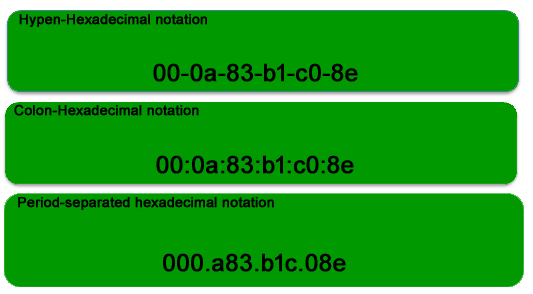
MAC Address is a 12-digit hexadecimal number (6-Byte binary number), which is mostly represented by Colon-Hexadecimal notation. The First 6 digits (say 00:40:96) of the MAC Address identify the manufacturer, called the OUI (**Organizational Unique Identifier**). IEEE Registration Authority Committee assigns these MAC prefixes to its registered vendors.

Here are some OUI of well-known manufacturers:

CC:46:D6 - Cisco   
3C:5A:B4 - Google, Inc.  
3C:D9:2B - Hewlett Packard  
00:9A:CD - HUAWEI TECHNOLOGIES CO.,LTD

The rightmost six digits represent **Network Interface Controller**, which is assigned by the manufacturer.

As discussed above, the MAC address is represented by Colon-Hexadecimal notation. But this is just a conversion, not mandatory. MAC address can be represented using any of the following formats:



**Note:** Colon-Hexadecimal notation is used by Linux OS and Period-separated Hexadecimal notation is used by Cisco Systems.

## How do I find the MAC Address?

A MAC address is mostly used to configure a router for a network device or during troubleshooting. The address of our computer device can be easily checked with any operating device. All the Apple devices connected to our home network contain a unique MAC address. Manufacturers may identify a MAC address by other names, such as the physical address, hardware ID, wireless ID, and Wi-Fi address.

Following are the steps which help to find MAC addresses for different OS

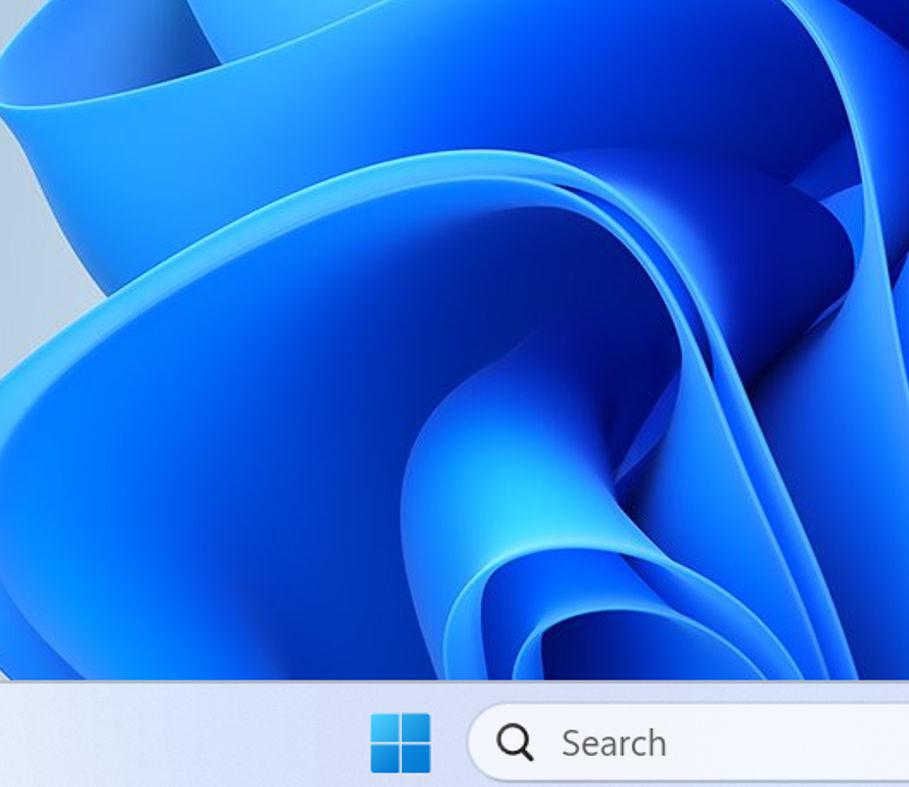
### 1. Command for a MAC Address in Windows OS

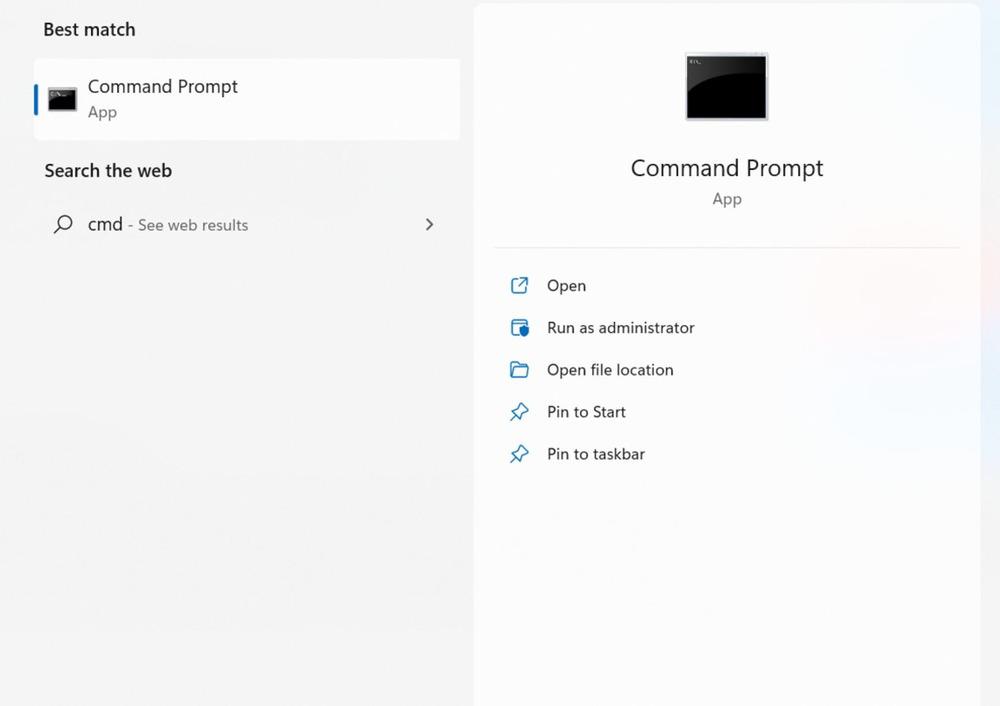
**Command:**

ipconfig /all

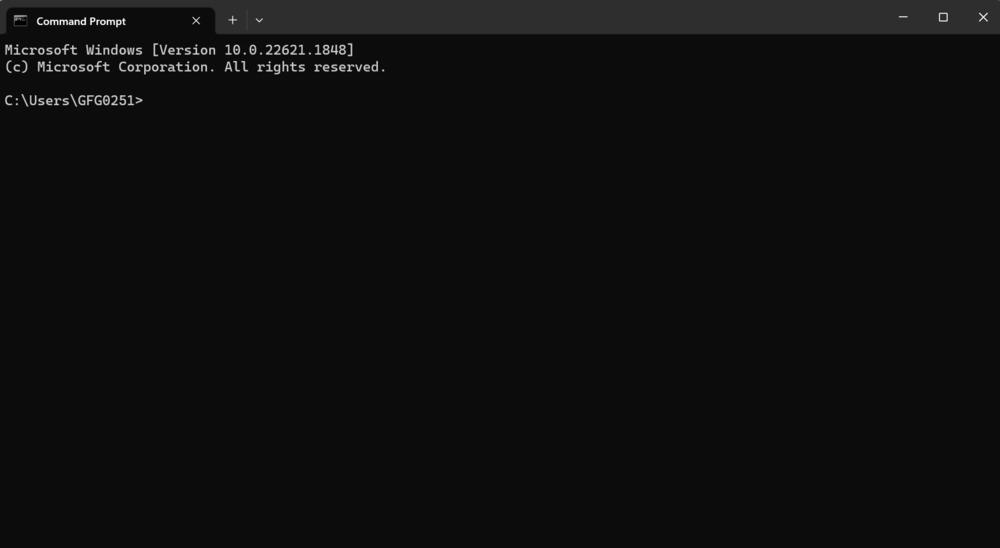
**Steps:**

**1.** Press **Window Start** or the Windows Key.

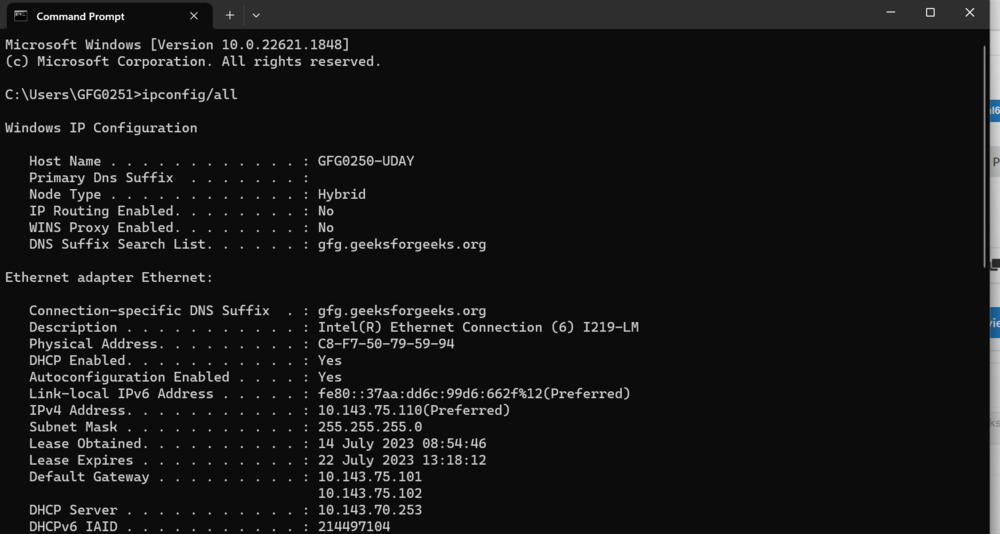
**2**. In the search box, type **cmd,** and the command prompt will get open.



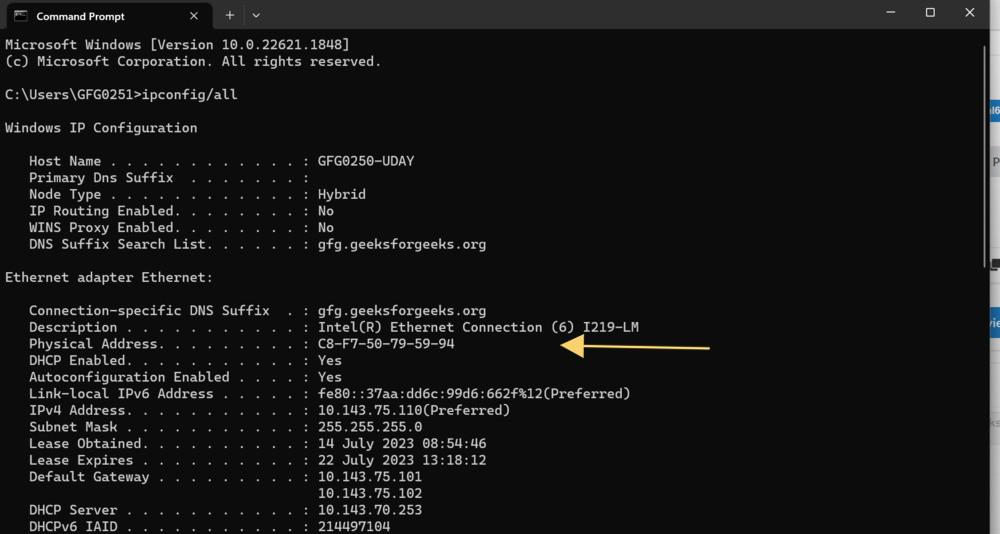
**3**. Click on cmd, the command prompt window will display,



**4.** In the command prompt type **ipconfig/all** command and then press enter.



**5**. As you will scroll down, each physical address is the MAC address of your device.



### 2. Command for MAC Address in MacOS

**Command:**

TCP/IP Control Panel

**Steps:**

**1**. Click on System Settings.



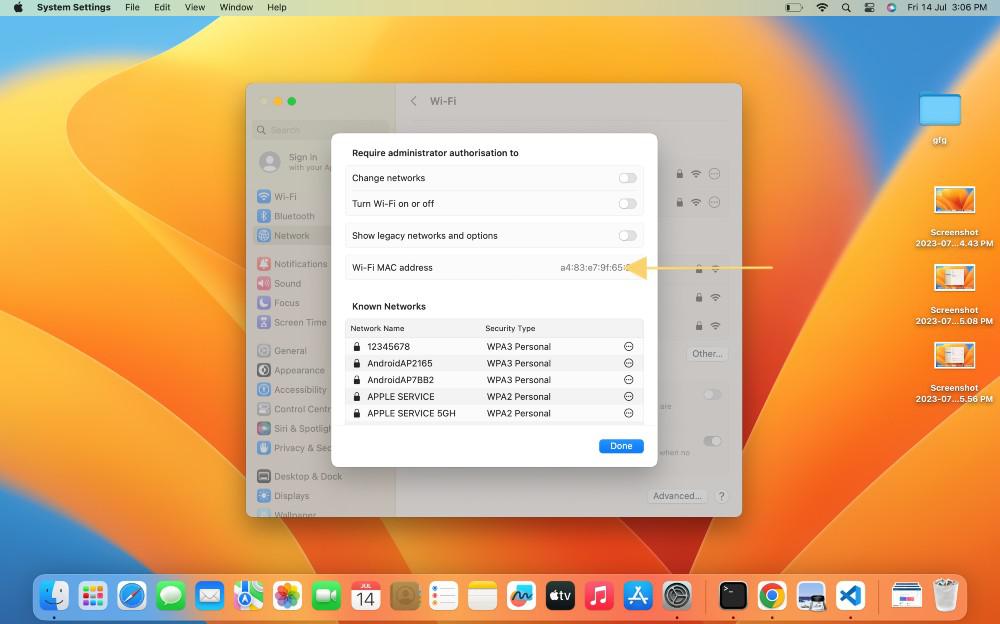
2. In the system settings, click on the **MAC network** option.



3. Then go to the **advanced settings.**



4. Here you find your MAC address.



### 3. Command For MAC Address in Unix/Linux

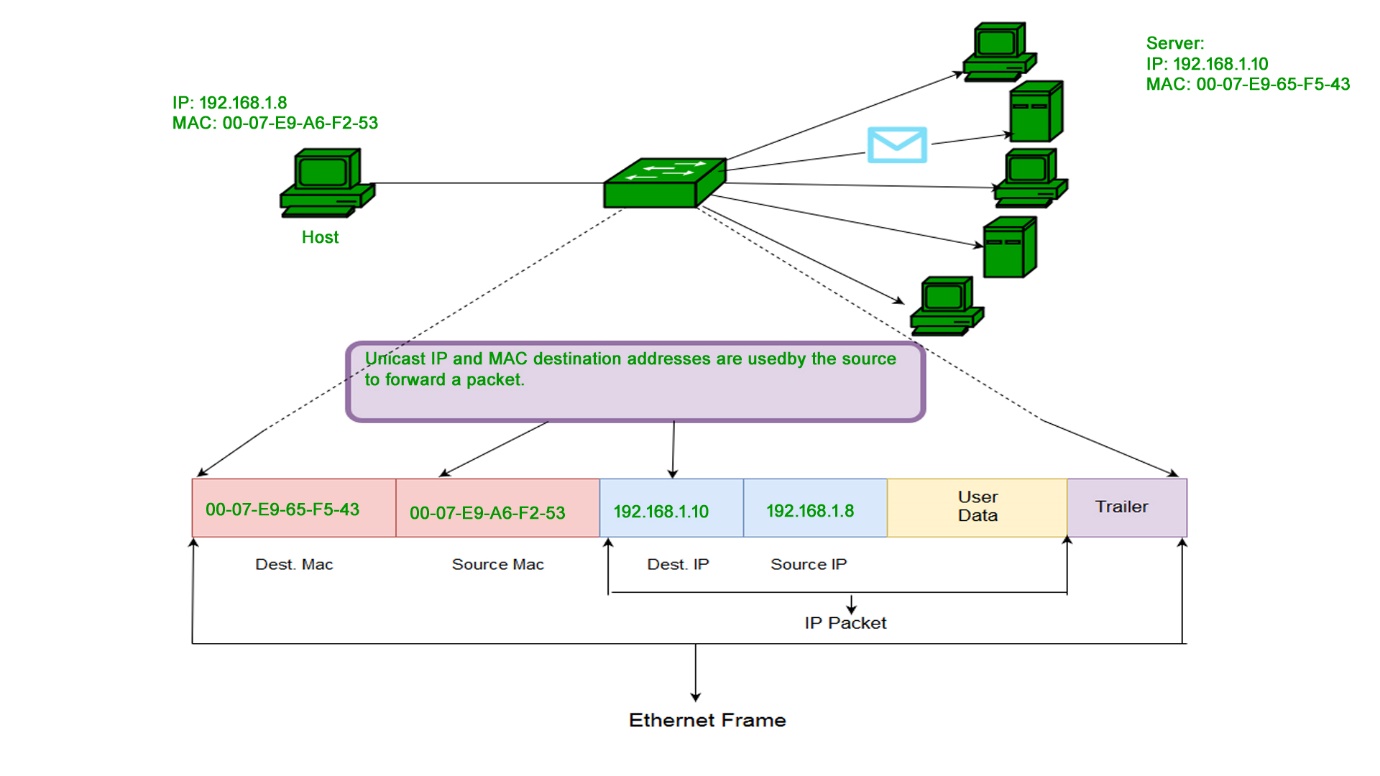
**Command:**

ifconfig -a  
ip link list   
ip address show

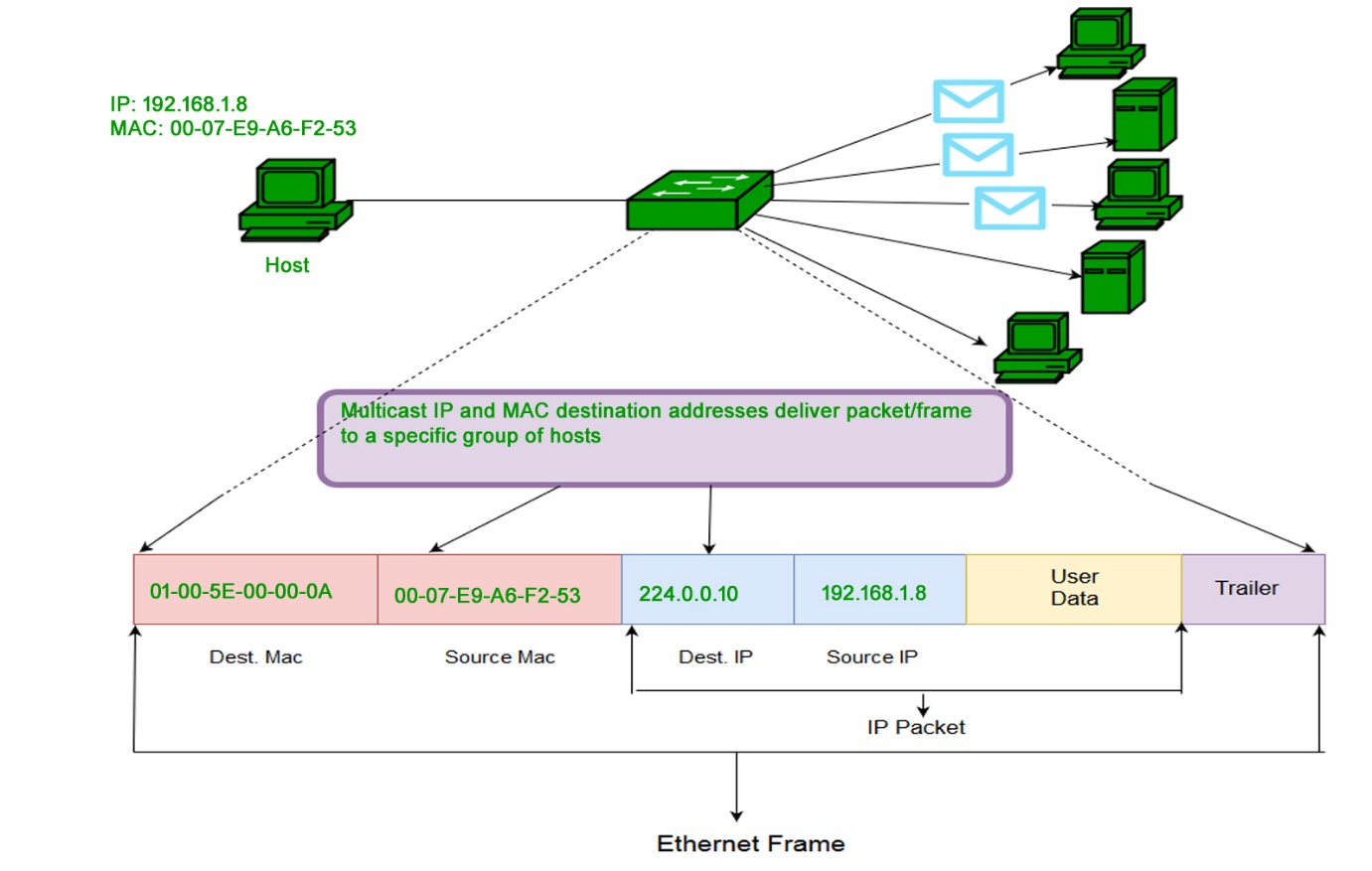
**Note:** LAN technologies like Token rings and Ethernet use MAC Addresses as their Physical address but there are some networks (AppleTalk) that do not use MAC addresses.

## Types of MAC Address

**1. Unicast:** A Unicast-addressed frame is only sent out to the interface leading to a specific NIC. If the LSB (least significant bit) of the first octet of an address is set to zero, the frame is meant to reach only one receiving NIC. The MAC Address of the source machine is always Unicast.



**2. Multicast:** The multicast address allows the source to send a frame to a group of devices. In Layer-2 (Ethernet) Multicast address, the LSB (least significant bit) of the first octet of an address is set to one. IEEE has allocated the address block 01-80-C2-xx-xx-xx (01-80-C2-00-00-00 to 01-80-C2-FF-FF-FF) for group addresses for use by standard protocols.



**3. Broadcast:**Similar to Network Layer, Broadcast is also possible on the underlying layer( Data Link Layer). Ethernet frames with ones in all bits of the destination address (FF-FF-FF-FF-FF-FF) are referred to as the broadcast addresses. Frames that are destined with MAC address FF-FF-FF-FF-FF-FF will reach every computer belonging to that LAN segment.

### Broadcast

## What is MAC Cloning?

Some ISPs use MAC addresses to assign an IP address to the gateway device. When a device connects to the ISP, the DHCP server records the MAC address and then assigns an IP address. Now the system will be identified through the MAC address. When the device gets disconnected, it loses the IP address. If the user wants to reconnect, the [DHCP](https://www.geeksforgeeks.org/dynamic-host-configuration-protocol-dhcp/) server checks if the device is connected before. If so, then the server tries to assign the same [IP address](https://www.geeksforgeeks.org/what-is-an-ip-address/) (in case the lease period has not expired). In case the user changed the router, the user has to inform the ISP about the new MAC address because the new MAC address is unknown to [ISP](https://www.geeksforgeeks.org/internet-service-provider-isp-hierarchy/), so the connection cannot be established.

Or the other option is **Cloning**, user can simply clone the registered MAC address with ISP. Now router keeps reporting the old MAC addresses to ISP and there will be no connection issue.

## Characteristics of MAC Address

The Media Access Control address (MAC address) is a unique identifier assigned to most network adapters or network interface cards (NICs) by the manufacturer for identification and use in the Media Access Control protocol sub-layer.  
An Ethernet MAC address is a 48-bit binary value expressed as 12 hexadecimal digits (4 bits per hexadecimal digit). MAC addresses are in a flat structure and thus they are not routable on the Internet. Serial interfaces do not use MAC addresses. It does NOT contain a network and host portion with the address. It is used to deliver the frame to the destination device.

* MAC addresses are used in LAN (Local Area Network) environments to identify devices and allow communication between them.
* MAC addresses are burned into the hardware of a network interface card (NIC) and cannot be changed, except in some rare cases where the manufacturer has provided a specific tool to do so.
* The first 3 bytes of a MAC address represent the manufacturer ID, while the last 3 bytes represent a unique identifier assigned by the manufacturer.
* MAC addresses are often used in conjunction with [ARP](https://www.geeksforgeeks.org/how-address-resolution-protocol-arp-works/) (Address Resolution Protocol) to resolve IP addresses to MAC addresses for communication on a LAN.
* Some operating systems, such as Windows and [Linux](https://www.geeksforgeeks.org/linux-vs-unix/), allow you to view the MAC address of your network adapter through a command prompt or network settings.

## Advantages

1. **Uniqueness:** Each MAC address is unique, which means that devices on the network can be easily identified and managed.
2. **Simplicity:** MAC addresses are easy to configure and manage, and do not require any additional network infrastructure.
3. **Compatibility:** MAC addresses are widely used and supported by a variety of networking technologies and protocols, making them compatible with many different systems.
4. **Security:** MAC addresses can be used to restrict access to a network by only allowing devices with authorized MAC addresses to connect.
5. **Fault-tolerance:** In case of hardware or software failure, a device can be easily replaced without affecting the network, as long as the new device has the same MAC address as the old one.
6. **Multicasting:** MAC addresses can be used for multicasting, allowing a single packet to be sent to multiple devices at once.
7. **Efficiency:** MAC addresses allow for efficient communication on the network, as they enable devices to quickly and easily identify and communicate with each other.
8. **Lower network overhead:** MAC addresses reduce network overhead by allowing devices to communicate directly with each other without the need for additional routing or addressing.
9. **Ease of troubleshooting:** MAC addresses can be used to troubleshoot network issues by identifying the source of problems and tracking network activity.
10. **Flexibility:** MAC addresses can be used to support a variety of network configurations and
11. topologies, including peer-to-peer, client-server, and hybrid models.

## Disadvantages

1. **Limited address space:** MAC addresses are 48-bit numbers, which means that there is a finite number of possible MAC addresses. This can lead to address conflicts if multiple devices have the same MAC address.
2. **Spoofing:** MAC addresses can be easily spoofed, allowing unauthorized devices to gain access to the network.
3. **Inefficiency:** MAC addresses are not hierarchical, which can make it difficult to efficiently manage large networks.
4. **Static addressing:** MAC addresses are typically assigned at the time of manufacture and cannot be easily changed. This can be a disadvantage in situations where devices need to be reconfigured or replaced.
5. **Limited scope:** [MAC addresses](https://www.geeksforgeeks.org/mac-address-in-computer-network/) are only used for identifying devices within a local network segment, and cannot be used to identify devices outside of this segment.
6. **Hardware-dependent:** MAC addresses are tied to the [network interface card](https://www.geeksforgeeks.org/how-to-install-a-network-interface-card/) (NIC) of a device, which means that if the NIC fails or is replaced, the MAC address also changes.
7. **Lack of encryption:** MAC addresses are sent in plain text, which can make them vulnerable to interception and eavesdropping.
8. **No inherent security:** While MAC filtering can be used to restrict access to a network, MAC addresses themselves do not provide any inherent security features.
9. **MAC address collisions:** In rare cases, MAC addresses can collide, which can cause network disruptions and make it difficult to identify and manage devices on the network.