1️⃣

Networking Notes by Rushi Padhiyar

Clear Text Protocols:

- > It transmit data over the network without any encryption.
- > Easy to intercept, modify and eavesdrop.
- > No alternative to this protocol and only used on trusted networks.

Cryptographic Protocols:

- > Uses encryption while transmitting the information
- > Useful in transmitting private information.
- > Harder to eavesdrop and modify the transferring information.

Q. What if we need to use clear text protocol on untrusted network?

- > We'll use a cryptographic tunnel i.e VPN
- > VPN uses cryptography to extend a private network over a public one.
- > Extension is made by performing a protected connection to a private network.
- > It is the same as being directly connected to a private network.
- > When connected to VPN, we are actually running the same protocols of the private network.
- > We can use a packet sniffer like Wireshark also.

Q. What is Wireshark?

- > A network sniffer tool.
- > We can see the data transmission over the network to and from our PC.

Protocols

- Machines talk to each other via protocols
- Protocols are in general rules for communication.

Packets

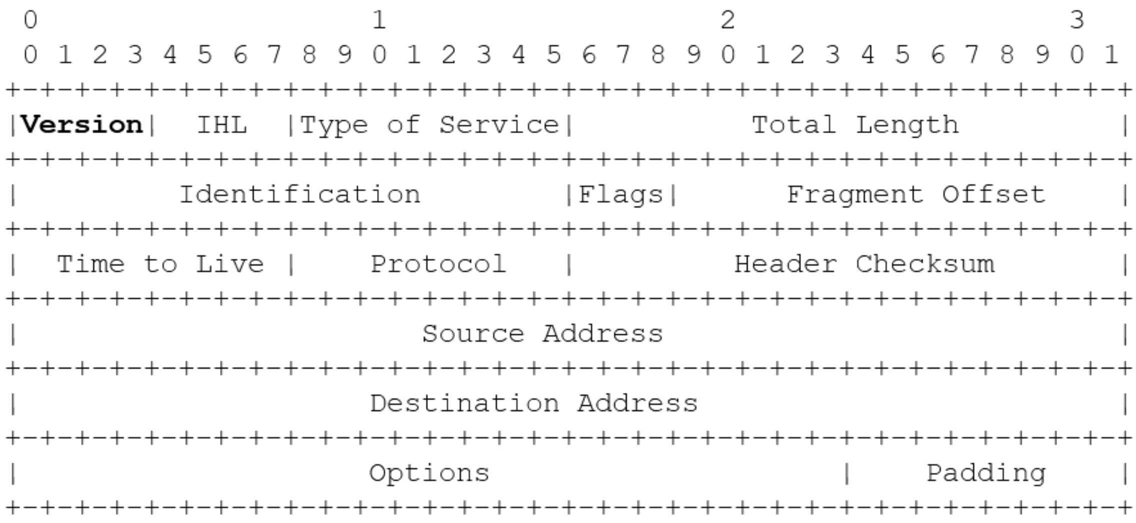
- The main goal of networking is to pass on the information to each other. They are done via packets.

- Packets are the stream of bits running in the form of electric signals in the physical media like LAN and in the form of air in the Wi-Fi network.

- Each packet consist of 2 main thing.

1. **Header**: Ensures that receiving host correctly executes the payload and handle overall communication.
2. **Payload**: Actual information / content.

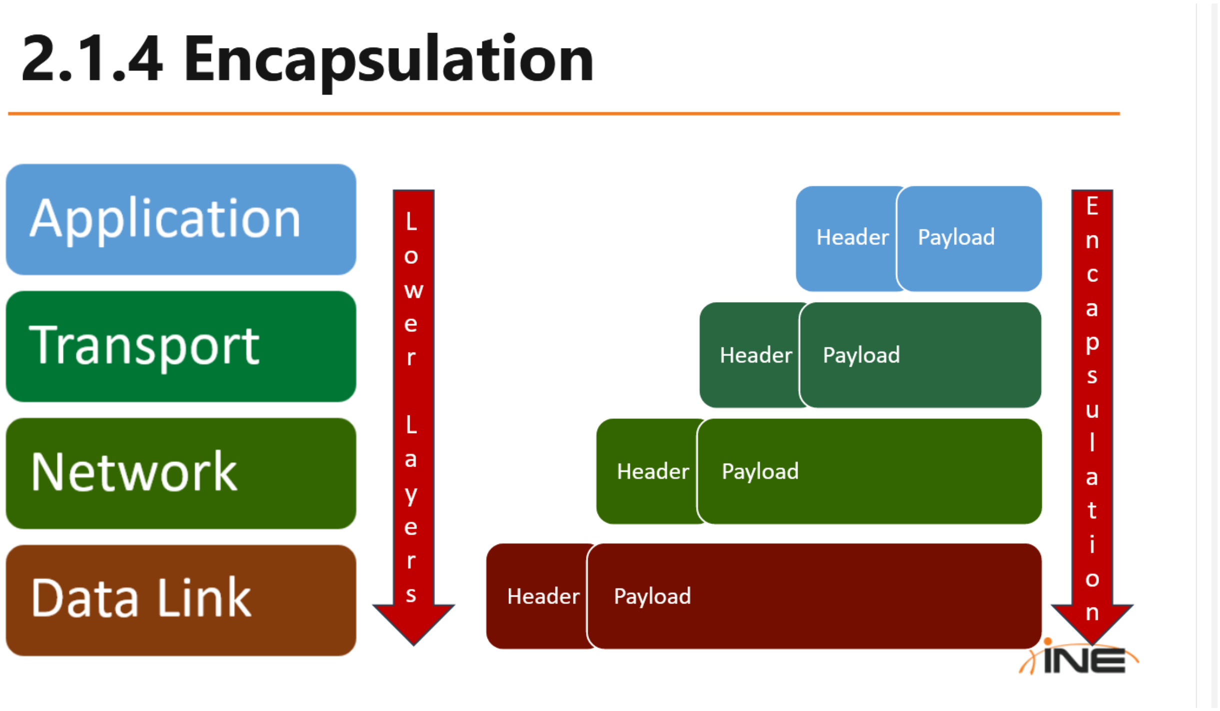
- IP header is at least 160 bits long



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Encapsulation:

- We know that there are 7 layers in the OSI model which can be said as the 7 protocol layers.
- In same way we have 4 layers of the TCP/IP model and it has the 4 protocol layers. The protocols are executed in such a manner that the Header and Payload of the first layer becomes the payload of the second one and the same goes for every layers of the mode. This is called ENCAPSULATION.



2.1.4 Encapsulation

IP - Internet Protocol

Protocol that runs on Internet Layer.

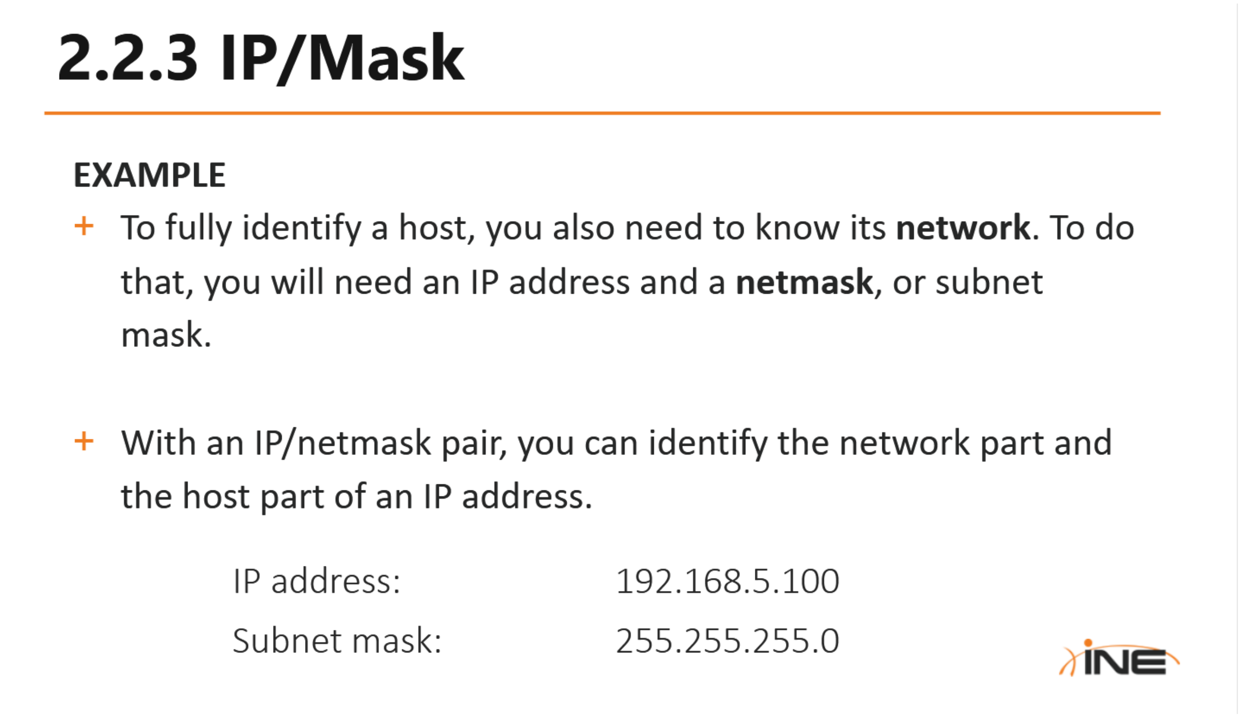
IP delivers the datagram to the hosts involved in the communications and it uses the IP address to identify the host.

IPv4 consists of 4 bytes or a octet

e.g. 73.5.12.132

Total Range: 0.0.0.0 - 255.255.255.255

To identify the host we need an IP address and a netmask (subnet mask)



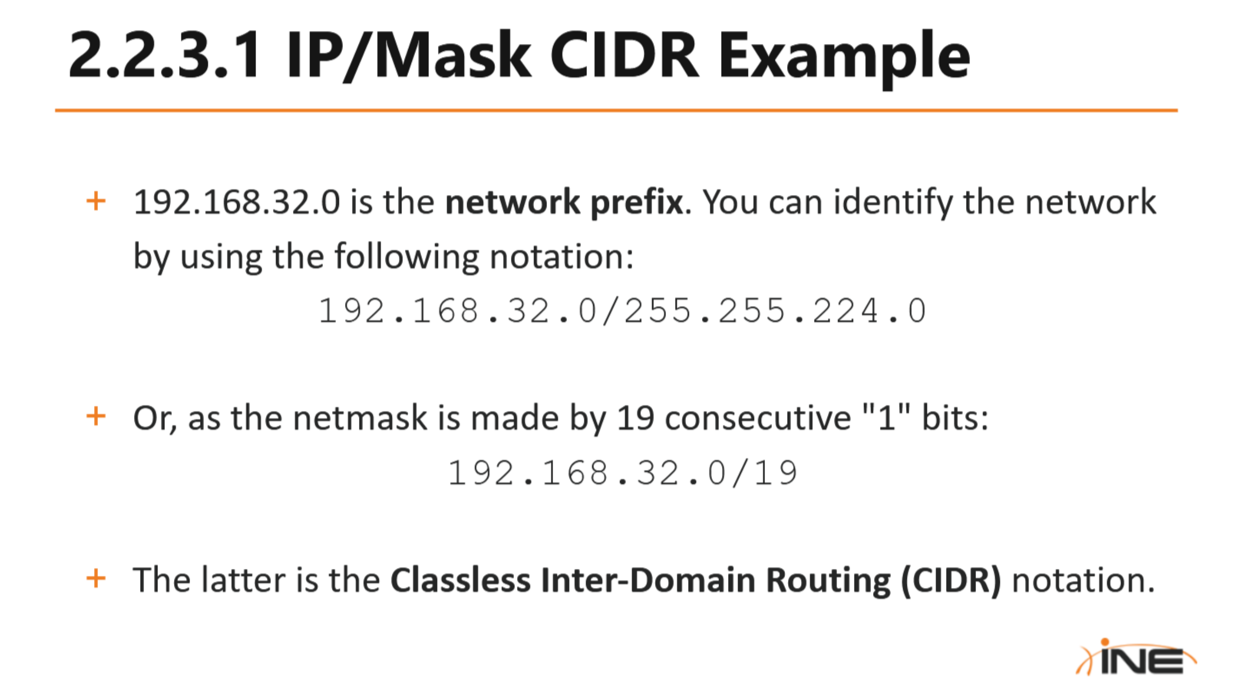
2.2.3 IP/Mask

EXAMPLE

- To fully identify a host, you also need to know its **network**. To do that, you will need an IP address and a **netmask**, or subnet mask.

- With an IP/netmask pair, you can identify the network part and the host part of an IP address.

IP address:	192.168.5.100
Subnet mask:	255.255.255.0



2.2.3.1 IP/Mask CIDR Example

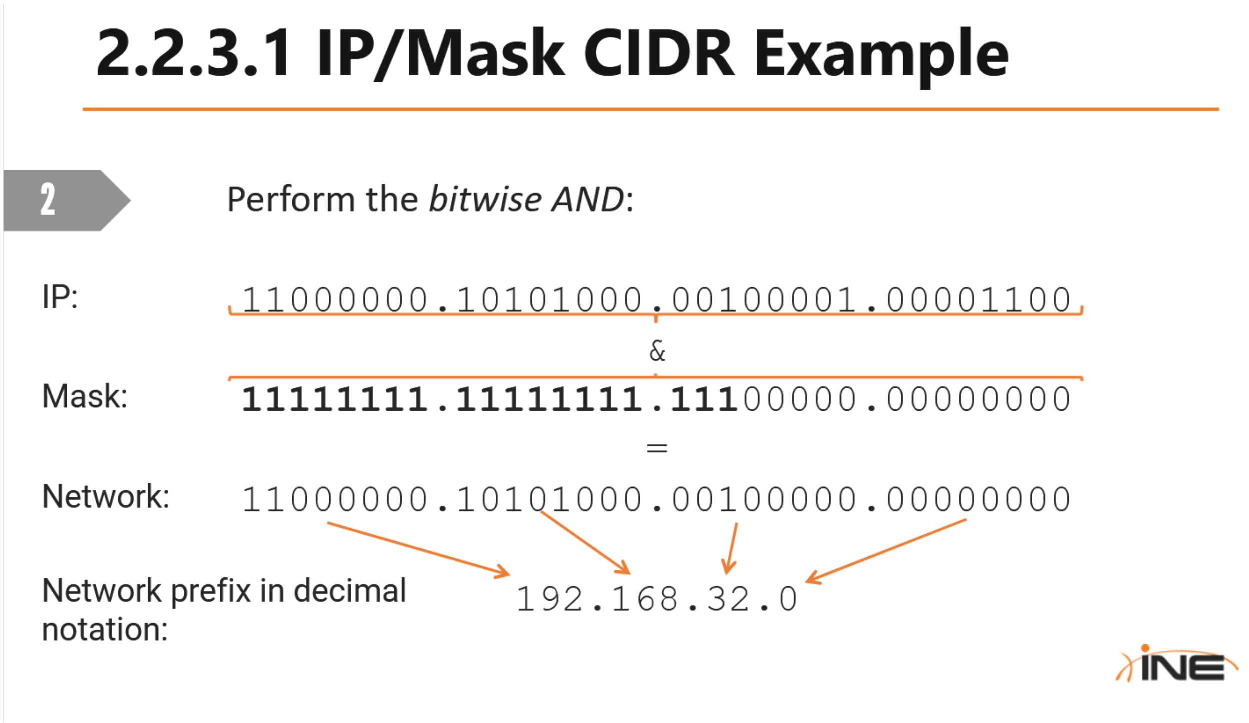
- 192.168.32.0 is the **network prefix**. You can identify the network by using the following notation:

```
192.168.32.0/255.255.224.0
```

- Or, as the netmask is made by 19 consecutive "1" bits:

```
192.168.32.0/19
```

- The latter is the **Classless Inter-Domain Routing (CIDR)** notation.



2.2.3.1 IP/Mask CIDR Example

2

Perform the *bitwise AND*:

IP: 11000000.10101000.00100001.00001100

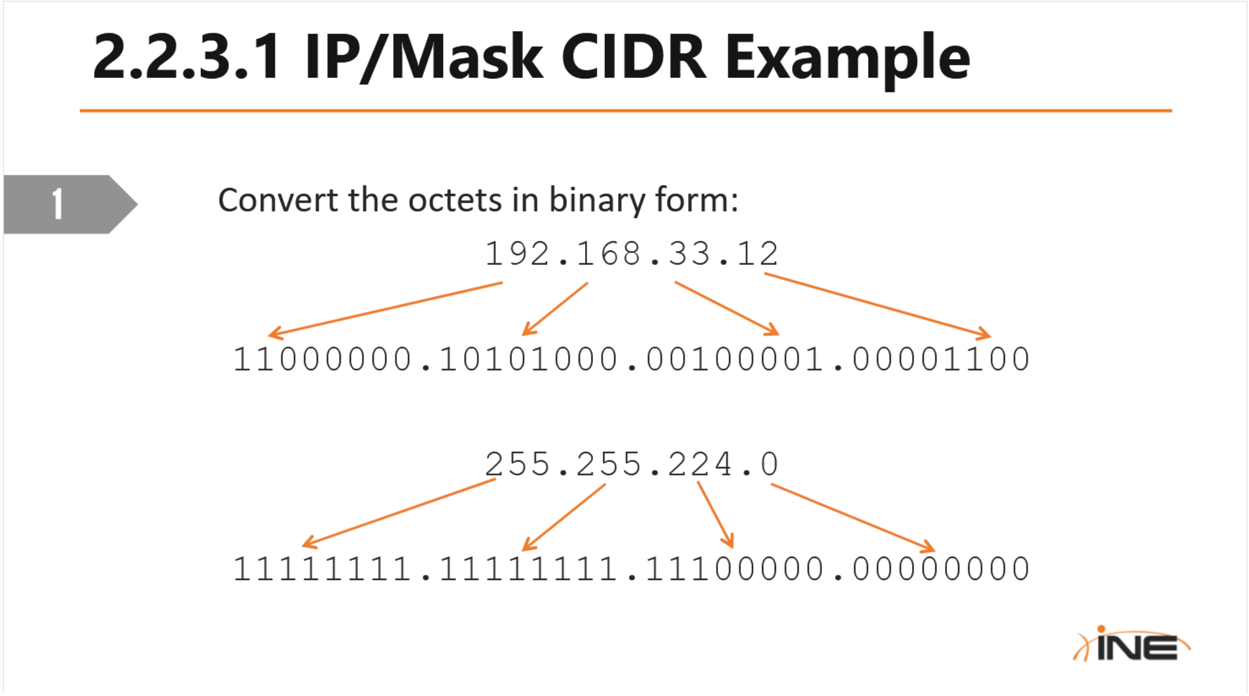
&

Mask: **11111111.11111111.111**00000.00000000

=

Network: 11000000.10101000.00100000.00000000

Network prefix in decimal notation: 192.168.32.0



2.2.3.1 IP/Mask CIDR Example

1

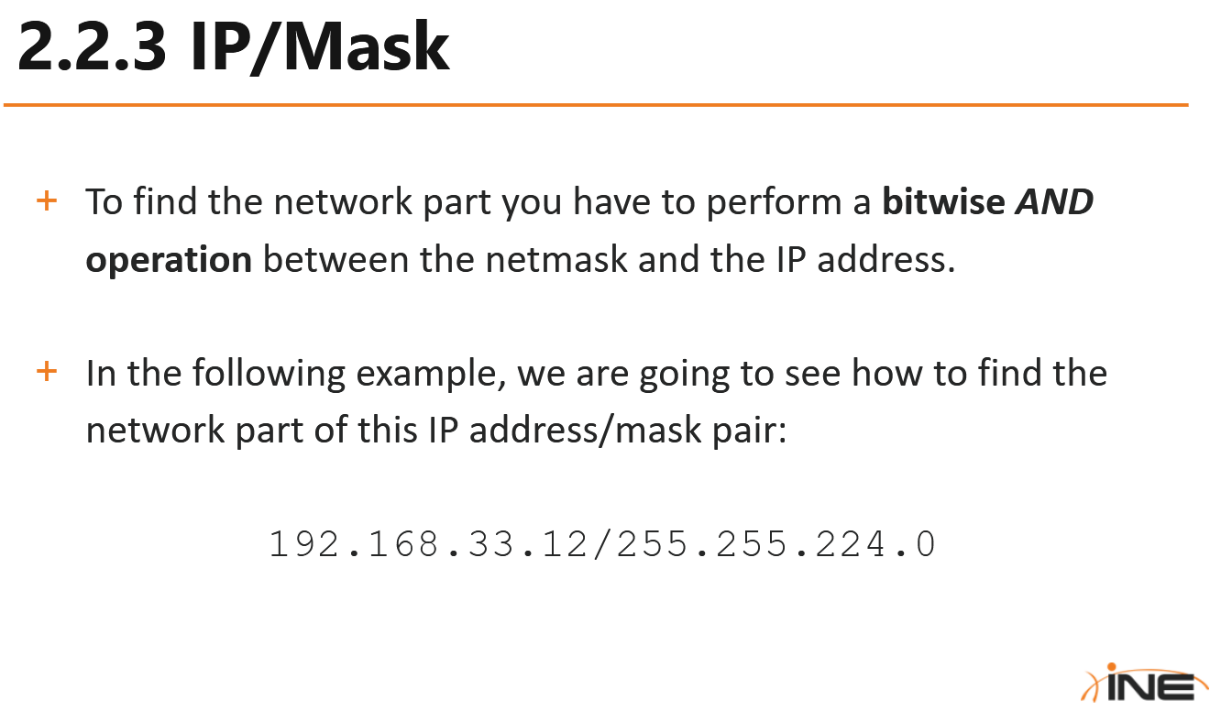
Convert the octets in binary form:

192.168.33.12

11000000.10101000.00100001.00001100

255.255.224.0

11111111.11111111.11100000.00000000



2.2.3 IP/Mask

- To find the network part you have to perform a **bitwise *AND* operation** between the netmask and the IP address.

- In the following example, we are going to see how to find the network part of this IP address/mask pair:

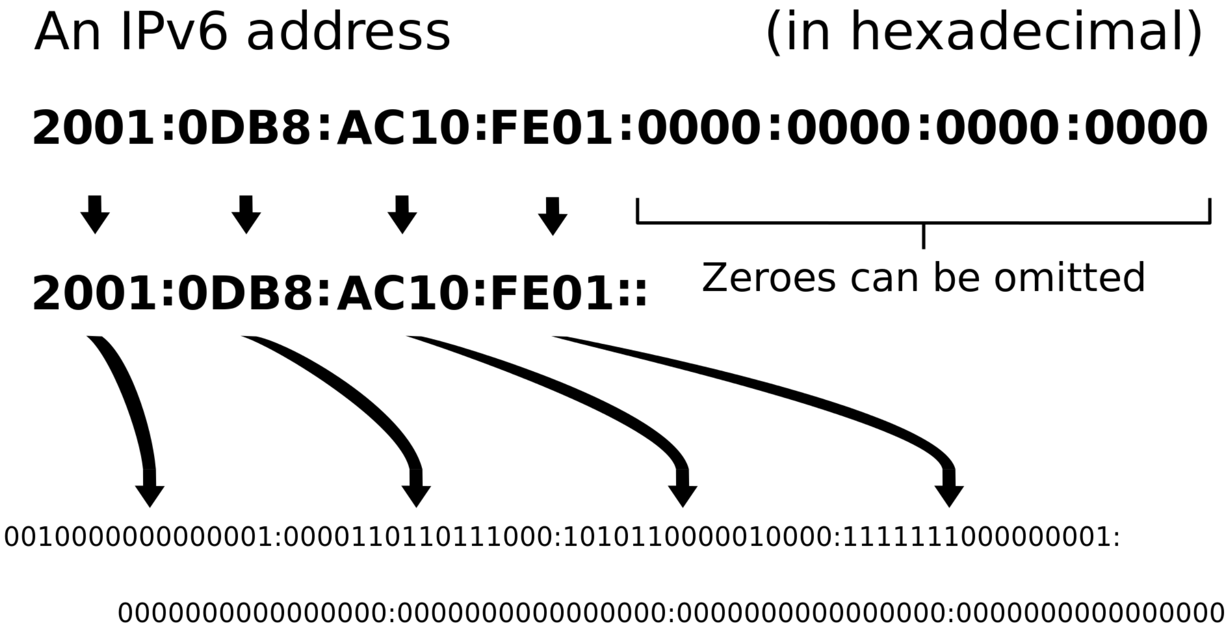
```
192.168.33.12/255.255.224.0
```

IPv6 Address:

IPv6 addresses consists of 128 bits which means 2^128 bits of the IP address.

16 bit hexadecimal numbers separated via a colon :

eg.



An IPv6 address (in hexadecimal)

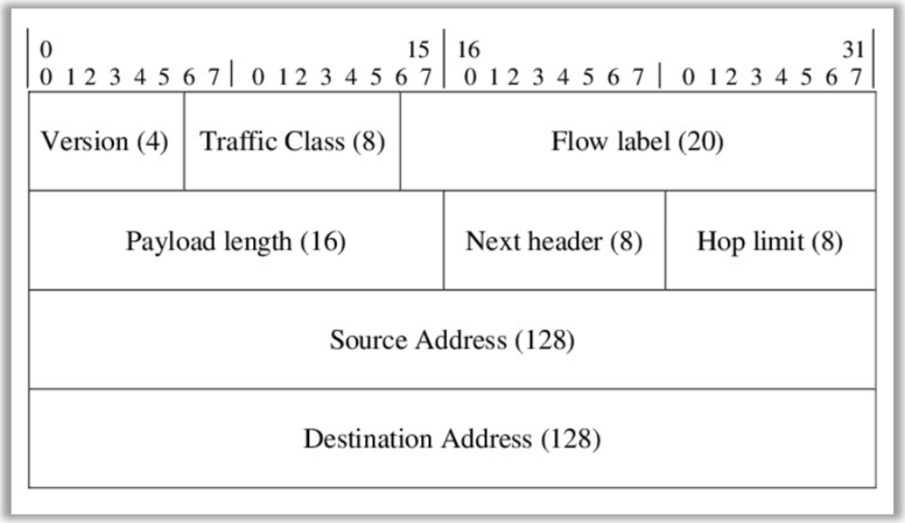
2001:0DB8:AC10:FE01:0000:0000:0000:0000

2001:0DB8:AC10:FE01:: Zeroes can be omitted

0010000000000001:0000110110111000:1010110000010000:1111111000000001:

0000000000000000:0000000000000000:0000000000000000:0000000000000000

2.2.7.1 IPv6 header

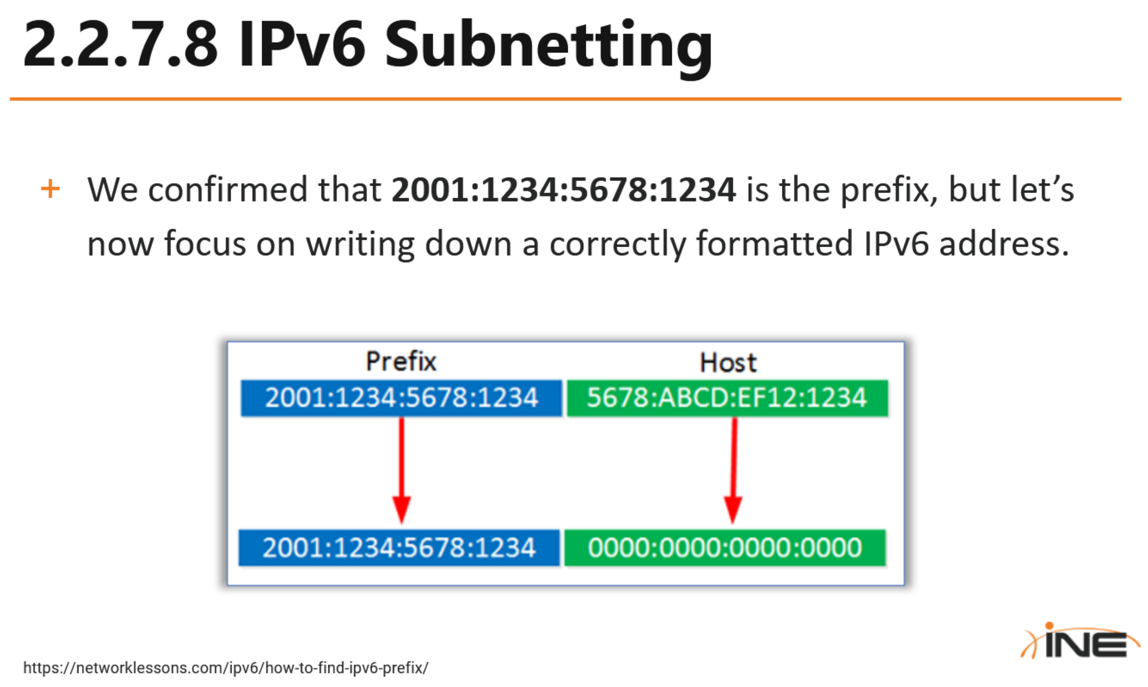


0 – 15		16 – 31	
Version (4)	Traffic Class (8)	Flow label (20)	
Payload length (16)		Next header (8)	Hop limit (8)
Source Address (128)			
Destination Address (128)			

https://www.researchgate.net/profile/Dragos_Truscan/publication/31596630/figure/download/fig5/AS:340685715722244@1458237212506/IPv6-Header-Format.png

The subnetting in IPv6 works like this:

Dividing the 128 bit into 2 parts of 64 bits each and the first 64 bit is the prefix part and the last 64 bit part is the host part.

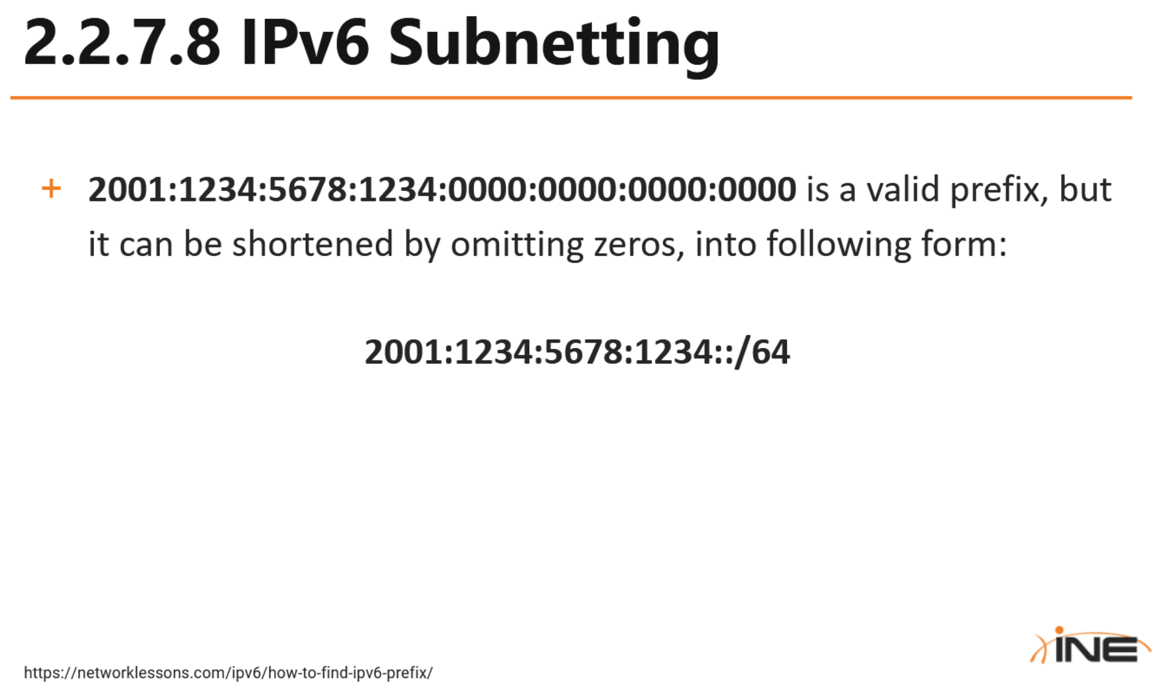


2.2.7.8 IPv6 Subnetting

- We confirmed that **2001:1234:5678:1234** is the prefix, but let's now focus on writing down a correctly formatted IPv6 address.

Prefix	Host
2001:1234:5678:1234	5678:ABCD:EF12:1234
2001:1234:5678:1234	0000:0000:0000:0000

https://networklessons.com/ipv6/how-to-find-ipv6-prefix/



2.2.7.8 IPv6 Subnetting

- **2001:1234:5678:1234:0000:0000:0000:0000** is a valid prefix, but it can be shortened by omitting zeros, into following form:

2001:1234:5678:1234::/64

https://networklessons.com/ipv6/how-to-find-ipv6-prefix/

Routing

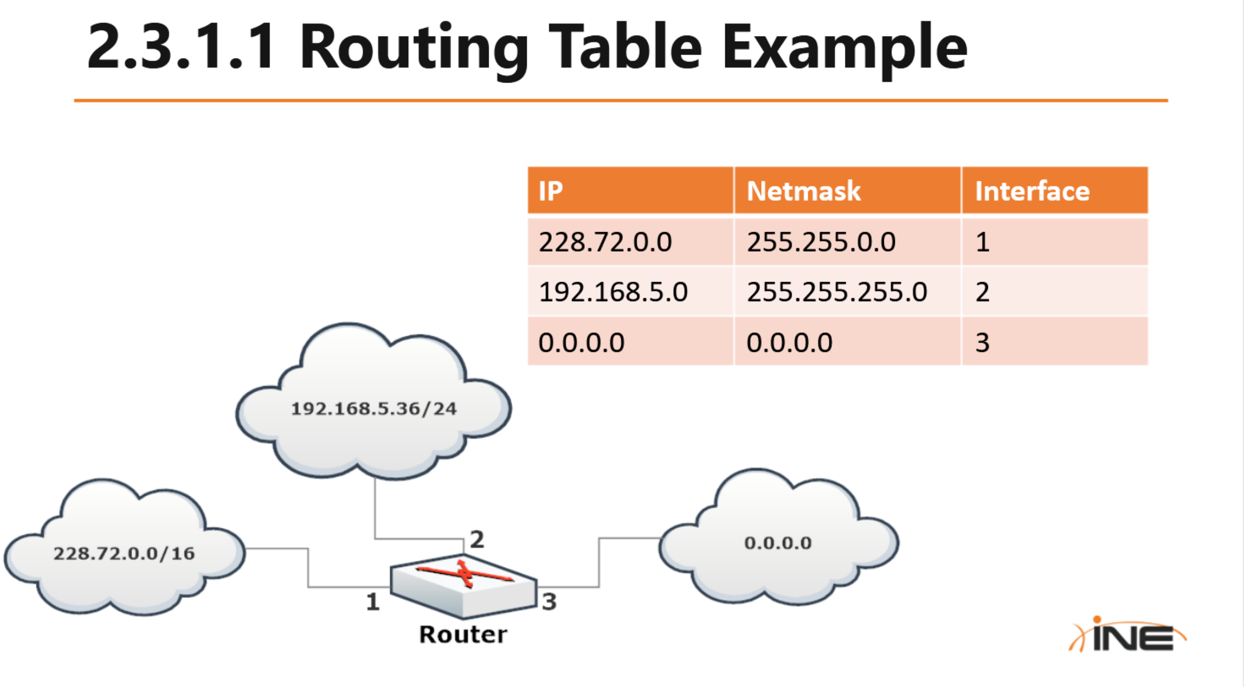
This is mainly used to perform network traffic inspection.

Packets are routed to each other via routers. They forward the datagrams from one network to another as many networks are connected via routers to each others.

What Routers do?

→ It inspects the destination address of every incoming packet and then forwards it through one of its interfaces.

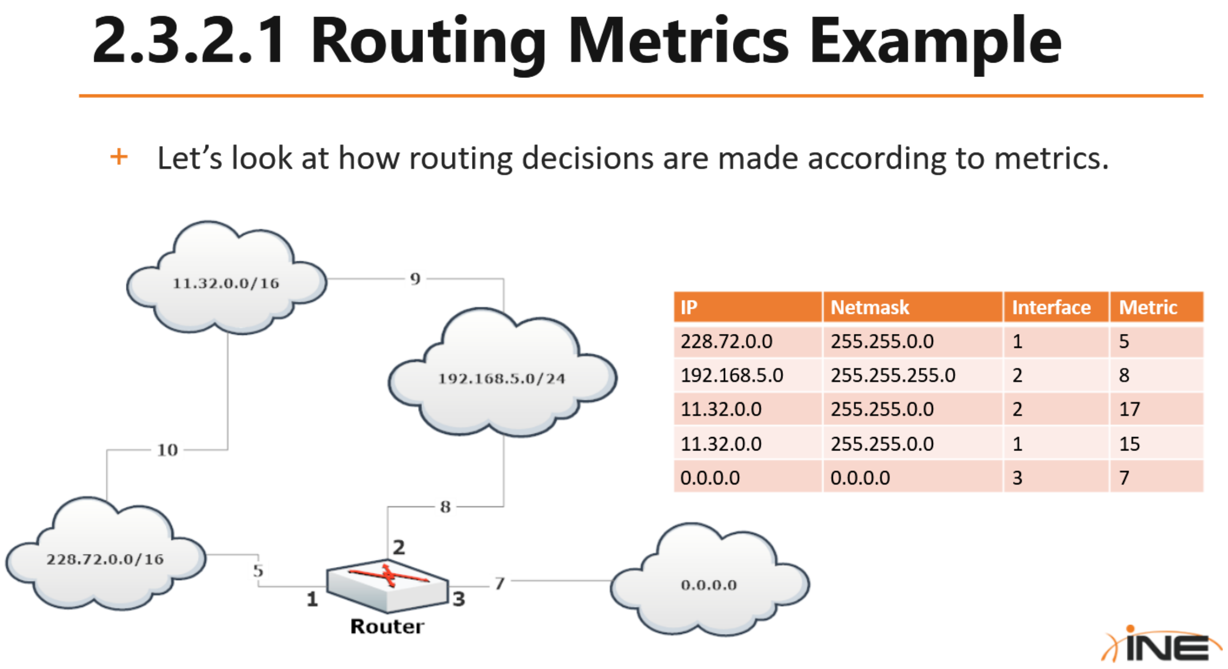
Routing table is used to find the IP to interface binding



2.3.1.1 Routing Table Example

IP	Netmask	Interface
228.72.0.0	255.255.0.0	1
192.168.5.0	255.255.255.0	2
0.0.0.0	0.0.0.0	3

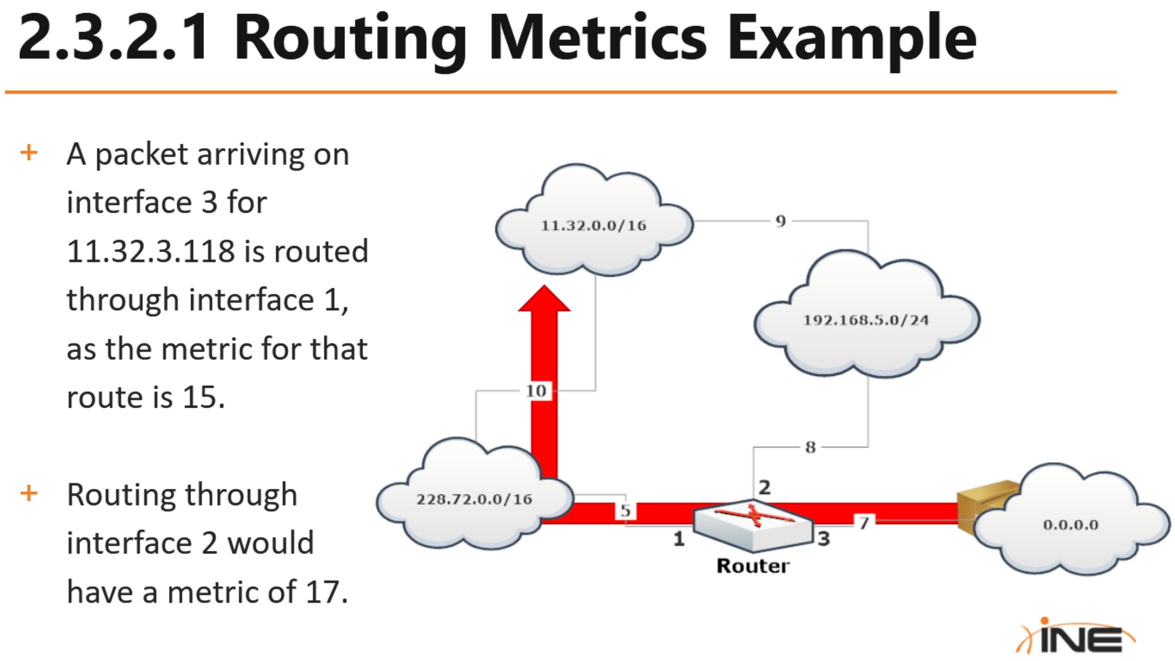
If two paths has same number of hops, the faster route will be selected.



2.3.2.1 Routing Metrics Example

- Let’s look at how routing decisions are made according to metrics.

IP	Netmask	Interface	Metric
228.72.0.0	255.255.0.0	1	5
192.168.5.0	255.255.255.0	2	8
11.32.0.0	255.255.0.0	2	17
11.32.0.0	255.255.0.0	1	15
0.0.0.0	0.0.0.0	3	7



2.3.2.1 Routing Metrics Example

- A packet arriving on interface 3 for 11.32.3.118 is routed through interface 1, as the metric for that route is 15.
- Routing through interface 2 would have a metric of 17.

Here if Packets of interface 2 has to go to routers then the metric will be 15 because of incoming 10 and outgoing 5.

To check Routing Table of your device a command is :

- Linux: `ip route`
- Windows: `print route`

Link Layer Devices and Protocols

Why to learn it?

→ To know the basics of

MAC Spoofing

MITM Attacks

Sniffing techniques

MAC Address → Media Access Control Address

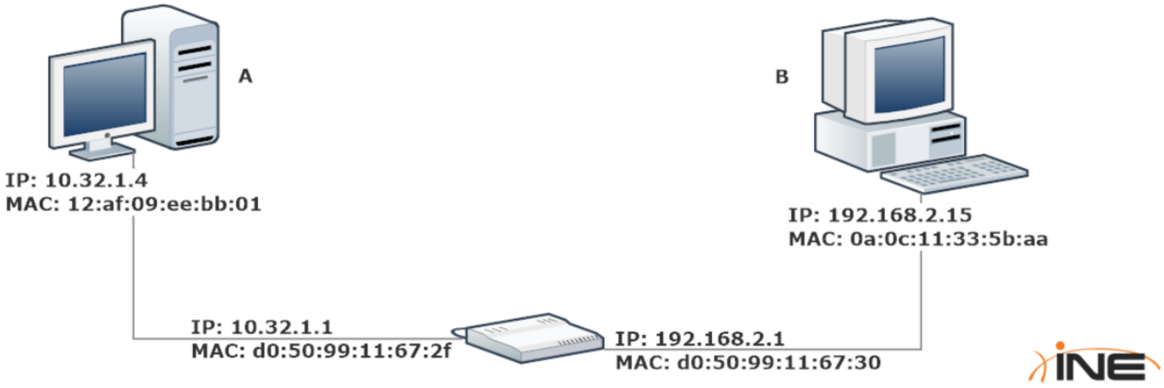
1. Physical Address of the device
2. Used to identify the network card of the device
3. 48 bits long (6 bytes)
4. Hexadecimal format.

Command:

Windows → `ipconfig`

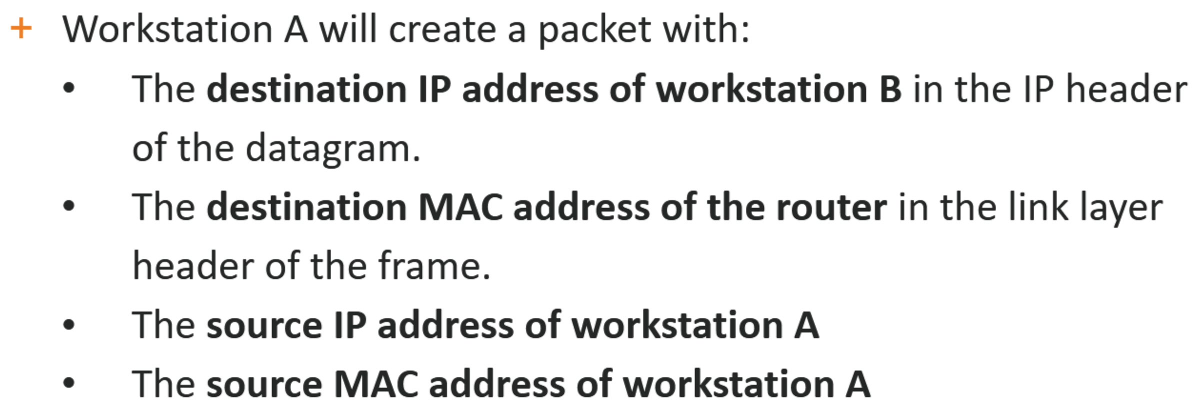
Linux → `ip addr`

- If workstation A wants to send a packet to workstation B, which IP and MAC addresses will it use?



- The router will then take the packet and forward it to B's network, **rewriting the packet's MAC addresses**:
  - The **destination MAC address** will be B's
  - The **source MAC address** will be the router's

- The router will not change the source and destination IP addresses.



+ Workstation A will create a packet with:

- The **destination IP address of workstation B** in the IP header of the datagram.
- The **destination MAC address of the router** in the link layer header of the frame.
- The **source IP address of workstation A**
- The **source MAC address of workstation A**

Relation between and IP Address and MAC Address in the communication:

- If you want to send a letter to your friend then you need to know his/her home address (IP Address) and the address of the nearest Post Office (MAC Address) to drop the letter.

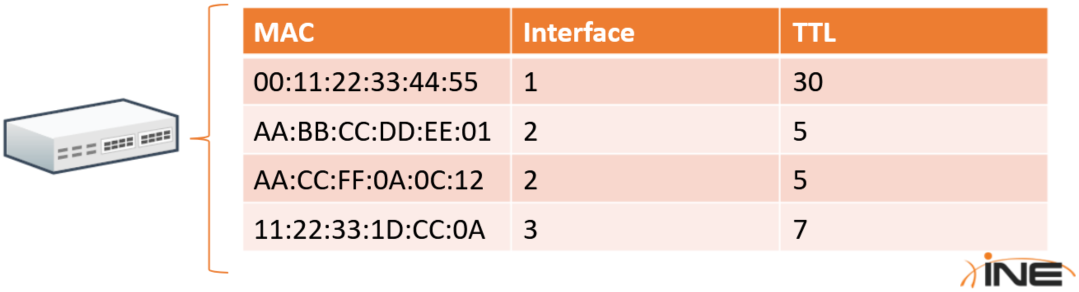
> 💡 FF:FF:FF:FF:FF:FF is the Broadcast MAC Address

The frame with the broadcast address is delivered to all the hosts in the local network.

Switches

Switches works with the MAC Addresses unlike the Routers that works with the IP Addresses.

1. They have a forwarding table like routing table which have all the information about the MAC address forwarding. It is also called CAM Table (Content Addressable Memory)

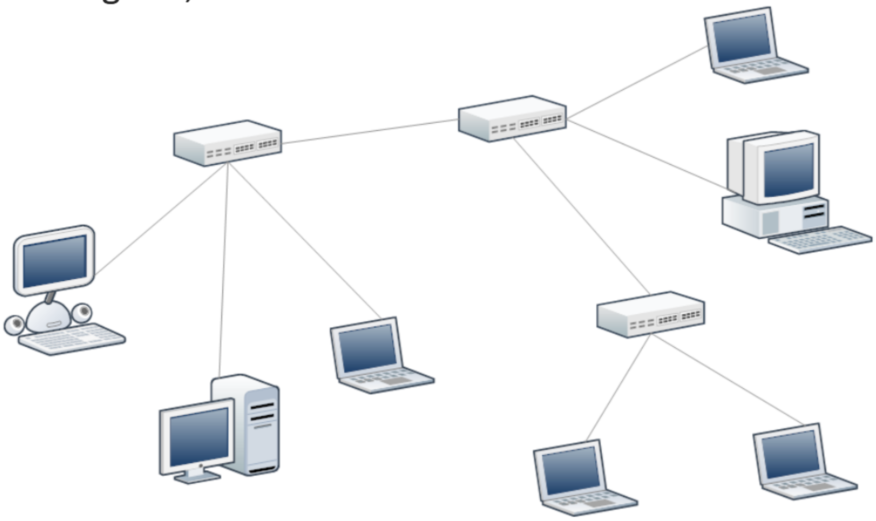


MAC	Interface	TTL
00:11:22:33:44:55	1	30
AA:BB:CC:DD:EE:01	2	5
AA:CC:FF:0A:0C:12	2	5
11:22:33:1D:CC:0A	3	7

2. Smallest switches have 4 ports and Corporate switches have 64 ports
3. 1Gbps is the common speed of the modern switches.

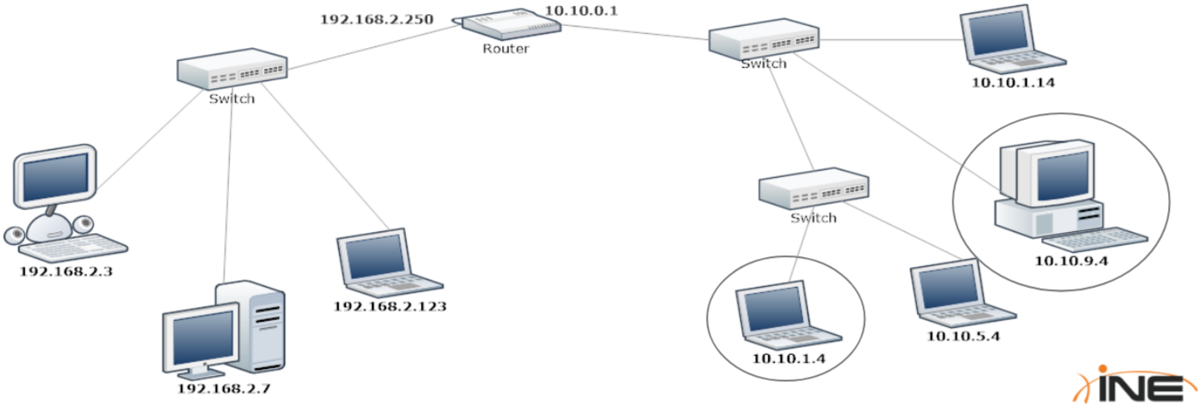
2.4.5.1 Multi-switch Network

+ In this diagram, all the machines are **on the same network**.

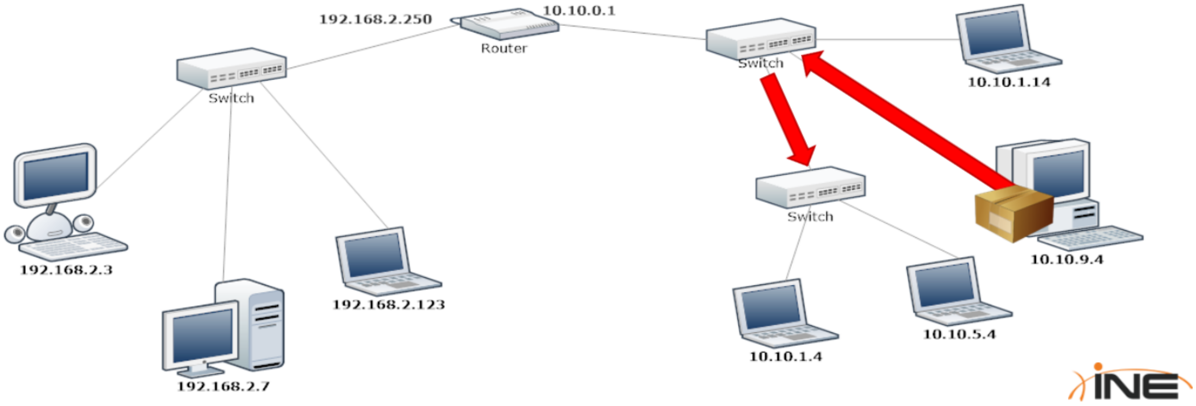


Now let’s see how the message is transferred from one PC to another in the interconnected network.

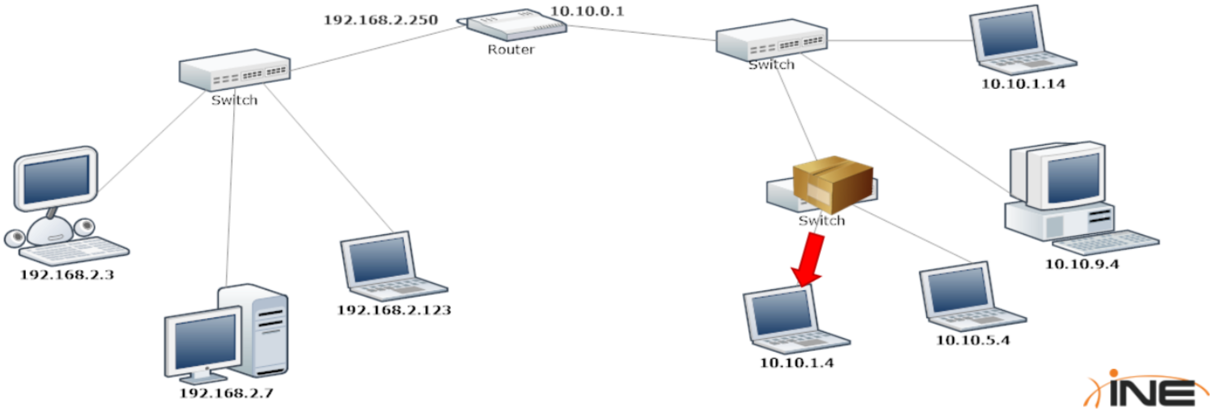
- What happens if 10.10.9.4 wants to send a packet to 10.10.1.4?



- The first switch receives the packet, performs a look-up in the CAM table and forwards it to the next switch.

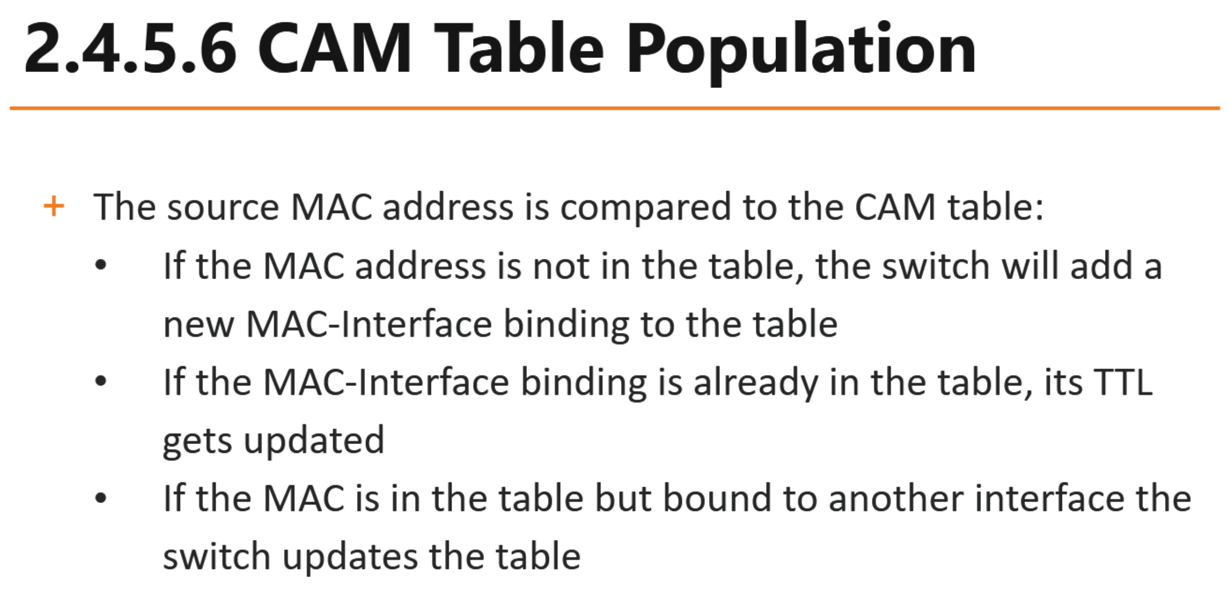


- The second switch forwards the packet to 10.10.1.4.



CAM Table

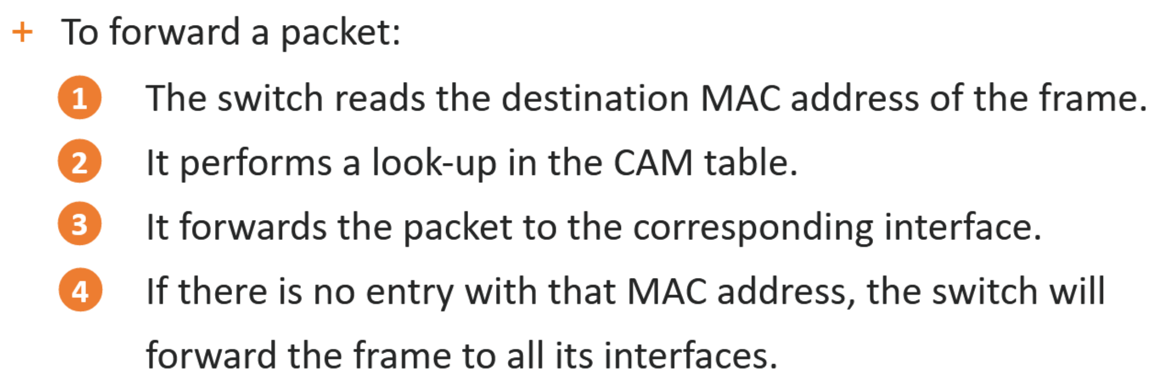
A CAM table or forwarding table contains the MAC Address and on which Interface its connected (switch) and the TTL(Time To Live) that determines how long an entry will stay in the table because the CAM table is of finite size.



2.4.5.6 CAM Table Population

+ The source MAC address is compared to the CAM table:
  - If the MAC address is not in the table, the switch will add a new MAC-Interface binding to the table
  - If the MAC-Interface binding is already in the table, its TTL gets updated
  - If the MAC is in the table but bound to another interface the switch updates the table

How the forwarding works?



+ To forward a packet:
  1. The switch reads the destination MAC address of the frame.
  2. It performs a look-up in the CAM table.
  3. It forwards the packet to the corresponding interface.
  4. If there is no entry with that MAC address, the switch will forward the frame to all its interfaces.

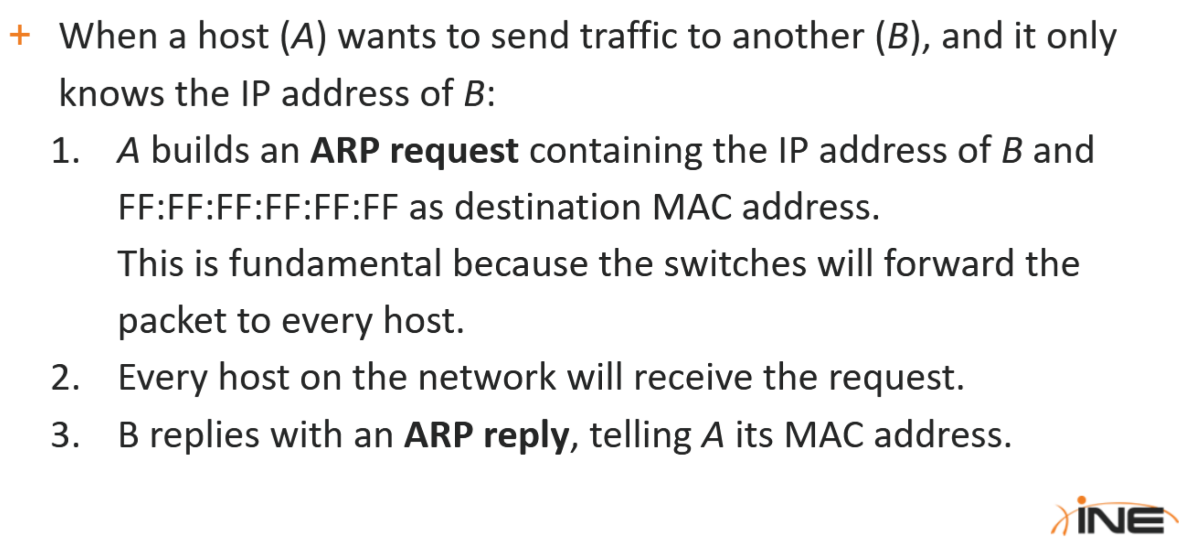
ARP → Address Resolution Protocol

When you want to send the letter to your friend but you don’t know the home address of the friend. The letter can’t be send.

In the same way a host needs to know both the Destination IP and Destination MAC Addresses.

The concept of ARP helps the host to build the correct IP Address - MAC Address pair building.

This is how the ARP works:



+ When a host (*A*) wants to send traffic to another (*B*), and it only knows the IP address of *B*:
  1. *A* builds an **ARP request** containing the IP address of *B* and FF:FF:FF:FF:FF:FF as destination MAC address.
     This is fundamental because the switches will forward the packet to every host.
  2. Every host on the network will receive the request.
  3. B replies with an **ARP reply**, telling *A* its MAC address.

‘A’ will then save the IP - MAC binding in it’s ARP cache. Further traffic to ‘B’ will not need a new ARP round.

But the ARP also have the TTL which means it will be erased once the system shut down or the expiry time is crossed.

Command:

Windows → `arp -a`

Linux → `ip neighbour`

TCP & UDP

Why to study them?

TCP Session Attacks

Advanced DoS Attacks

Network Scanning

TCP and UDP are the protocols by which the packets are transported in the transport layer.

TCP

1. Transmission Control Protocol
   a. Guarantees packet delivery
   b. Connection Oriented as it must establish a connection before transferring the data.
2. Most commonly used protocol on the internet.
3. Slower because of the assurance it carves.

UDP

1. User Datagram Protocol
   a. No Guarantee of the packet delivery
   b. Connectionless.
2. Faster than TCP
3. Used in VoIP and Video Streaming

UDP connections can tolerate the packet loss whereas the TCP connection can’t tolerate any packet loss.

Ports

They are used to identify a single network process on a machine

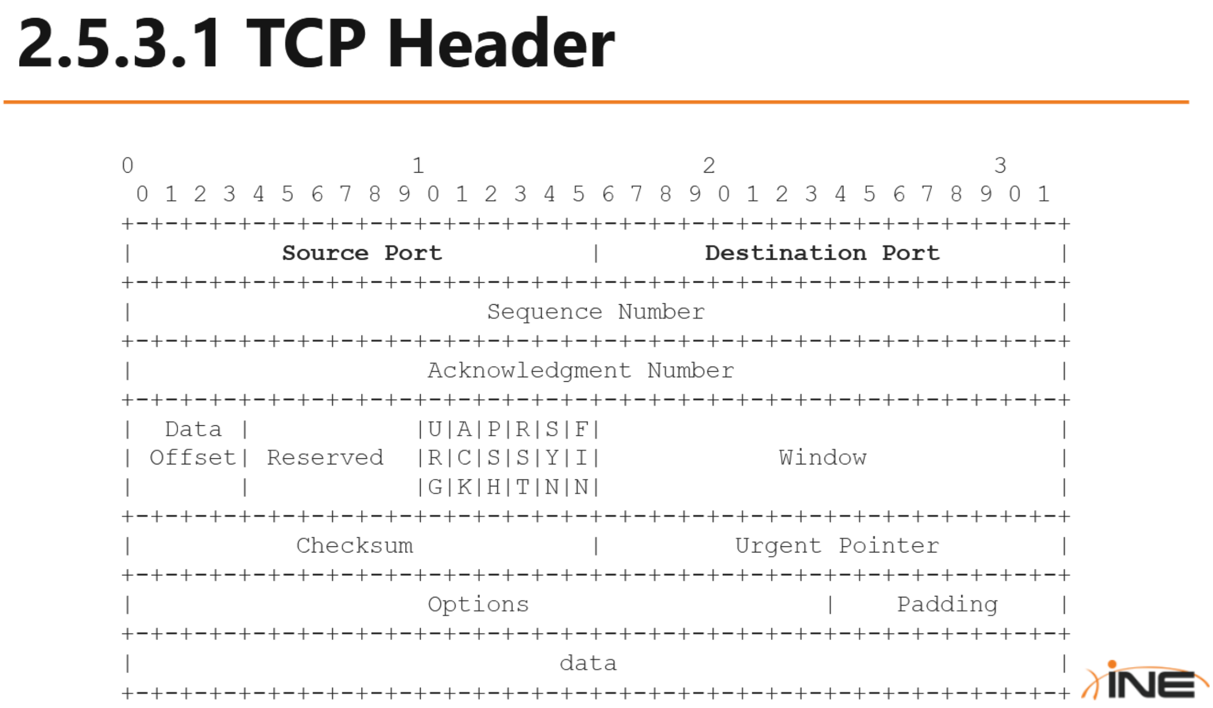
Format:

<IP>:<Port>

Example:

192.168.0.0.1:443

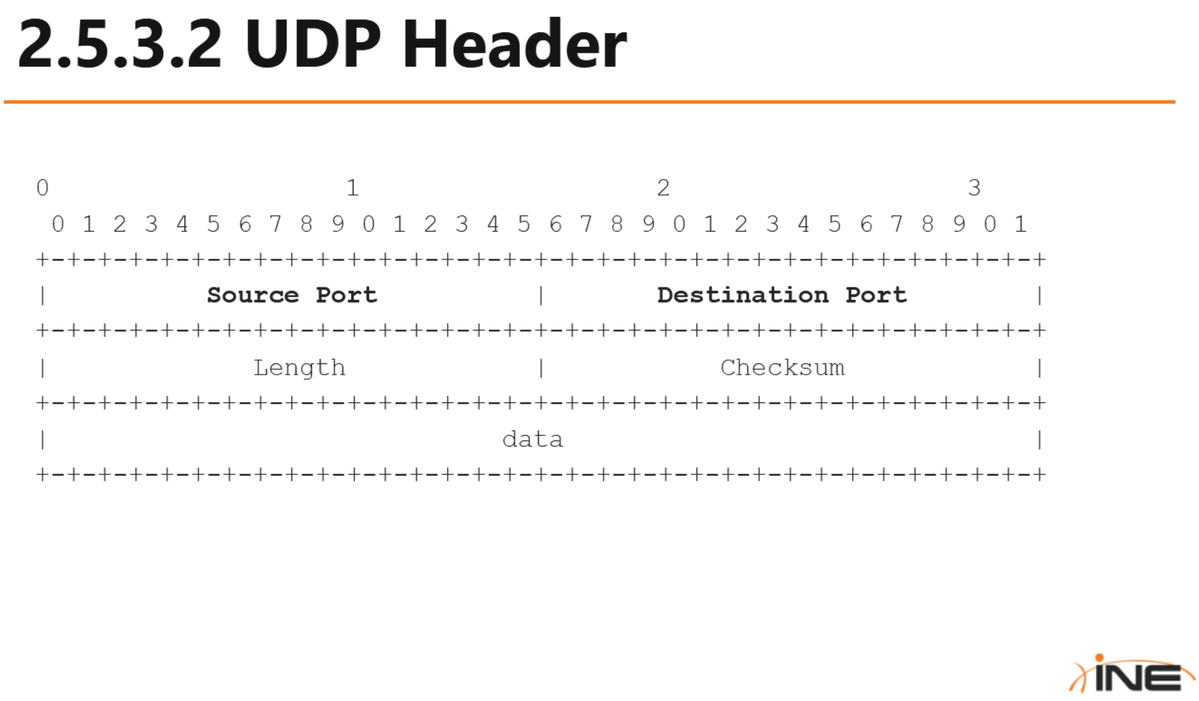
TCP Header has the following type of port binding:



2.5.3.1 TCP Header

```
0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port          |       Destination Port        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Acknowledgment Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Data |           |U|A|P|R|S|F|                               |
| Offset| Reserved  |R|C|S|S|Y|I|            Window             |
|       |           |G|K|H|T|N|N|                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |         Urgent Pointer        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

UDP Header has the following type of Port Binding:



2.5.3.2 UDP Header

```
0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port          |       Destination Port        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Length             |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

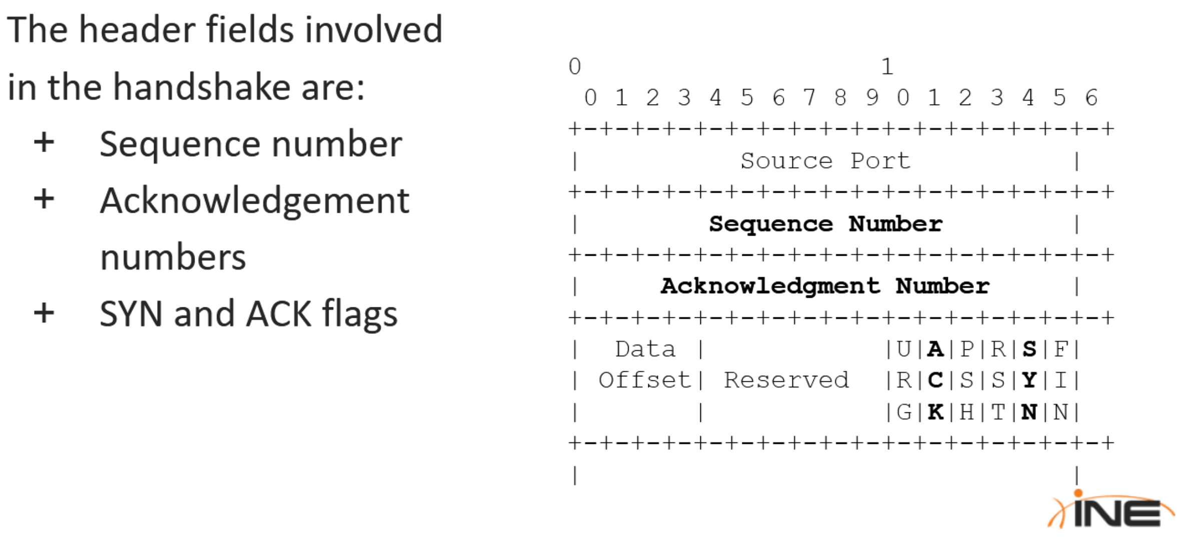
Command:

To check the listening ports and current TCP connections on a host we can use

Windows → `netstat -ano`

Linux → `netstat -tunp`

TCP 3 Way Handshake:

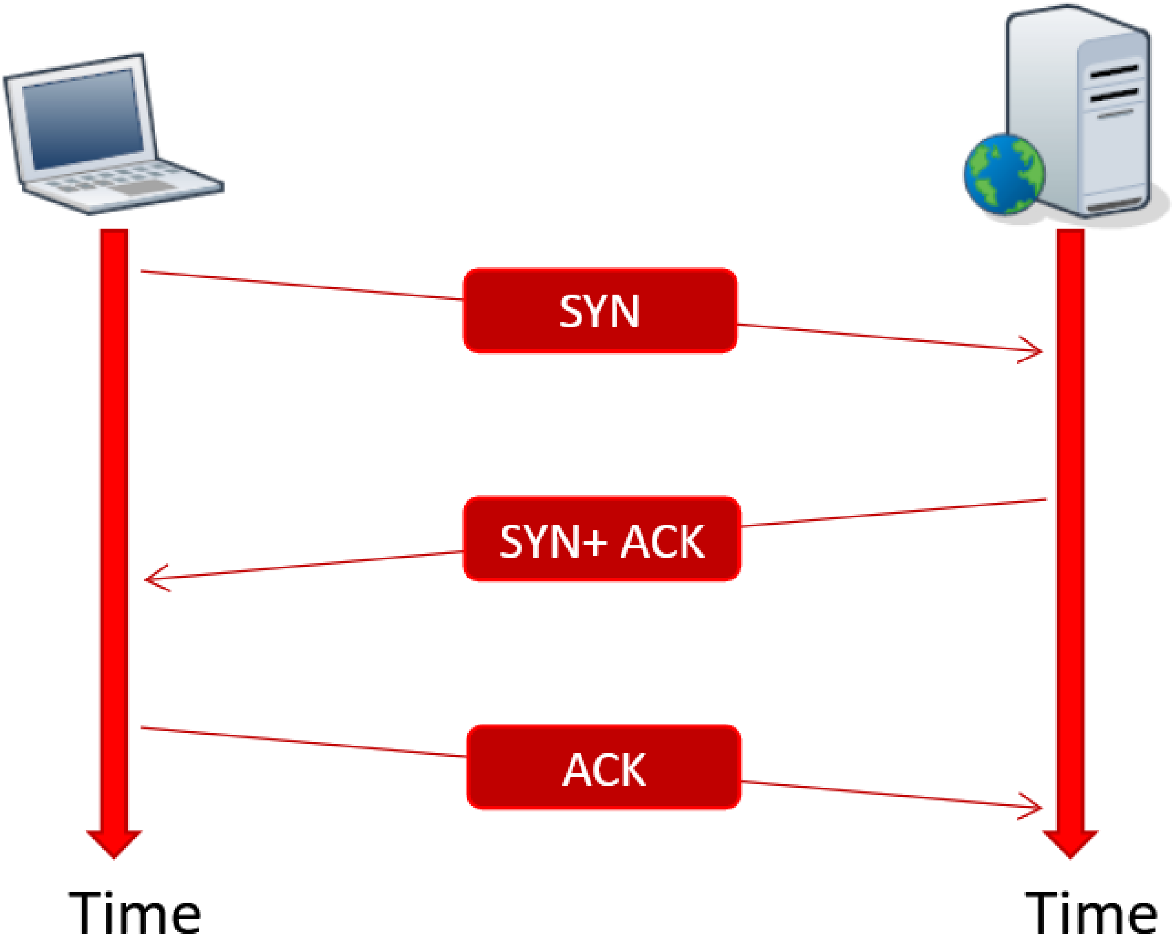


The header fields involved in the handshake are:

- Sequence number
- Acknowledgement numbers
- SYN and ACK flags

```
0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Sequence Number          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Acknowledgment Number      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Data  |          |U|A|P|R|S|F|
| Offset| Reserved  |R|C|S|S|Y|I|
|       |           |G|K|H|T|N|N|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                 |
```

TCP 3 Way Handshake:



PART - II will be dropping soon..stay tuned for that 😉

AUTHOR: Rushi Padhiyar a.k.a *Cyph3rRyx*