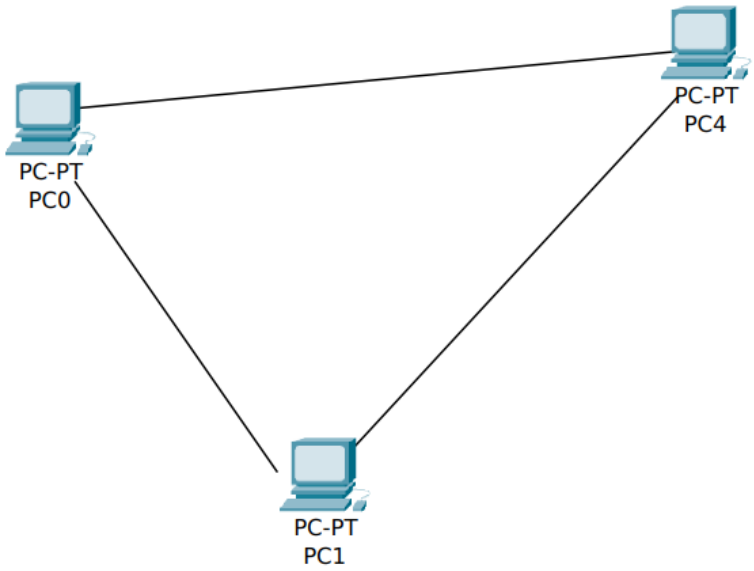
You might be wodering why IP addresses were born in the first place right? Well the answer to this question is not simple, and I am going to show you in depth why ip addresses came into pictures, So lets start...

In the olden days, in the era where there were very very very very few computers, I.e only researchers holding the computers, they typically used the computers just to go on with the research and computation, I.e the main task of the computer. They didn't needed the networking or communicating with other computers at all,

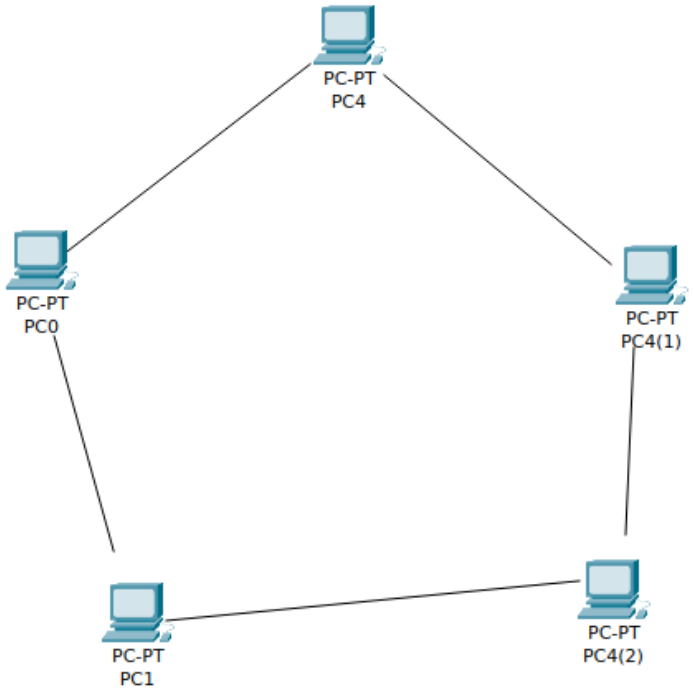
Later networking ivolved as researchers started collaborating on multiple computers connected to each other, where they developed simple protocols to be able to communicate with each other, via simple cables connected with each other, like,



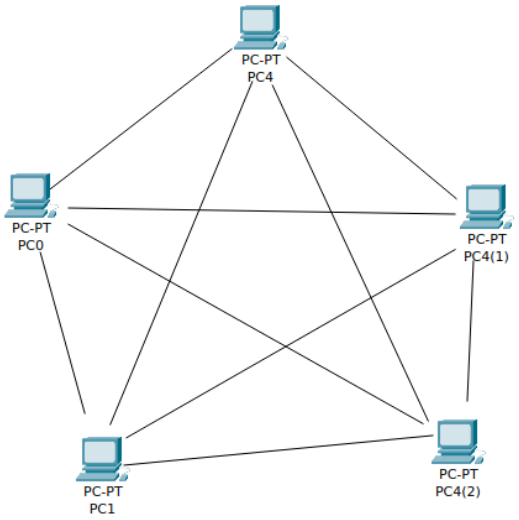
Here, as you can see, the computers are simply connected to each other with simple cables from their interfaces, I.e here pc0 will be having two interfaces to be able to connect to two computers, suppose eth0 and eth1 and this is for every computer, and each interface comes with its own MAC address, so using the simple protocol which involved usage of MAC addresses which were assigned or burned onto the NIC by the manufacturer was used, and there was no problem at all with this approach,

Later the network grown, and various topologies were introduced, like bus topology where the frames were broadcasted on the line and it was(and still) the responsibility of the interface to read the destination MAC and see whether that frame is for that computer or not, if it doesn't match with the computer's MAC, it would simply discard it, later another topology called RING topology was introduced,

here, the slight modification to the protocol was made here, instead of pc's interface dropping the packet, if the packet isn't for it, it would simply forward it, like this,



Here at that time, mesh topology was introduced, this was a good approach to eliminate redundant network traffic to nodes to whom packet doesn't belong, however this will also increase the interfaces on one computer available, so it is infeasible aand not scalable approach

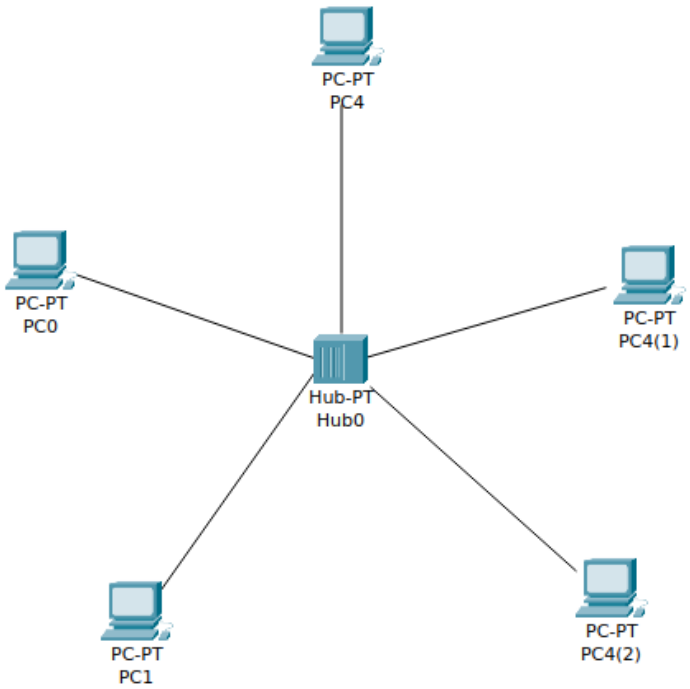


Here, the work was going well with the MAC addresses without a problem, but as we all know, we cannot know or there is no way to infer or at least imagine What MAC adddresses other pc will be having if a mac address of one pc is given, right?, consider the above pictures and you will understand.

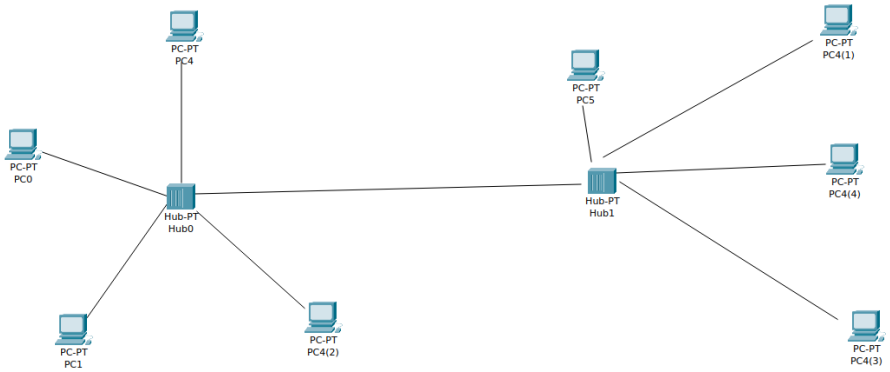
Now there was a need to have a central device which will co-ordinate and send packets on behalf of computers, as mesh topology had limitations which are obviously impossible to recover from end-pc point of view, isn't it?

Then a HUB was developed whose work was simple, to BROADCAST frames coming to it, ofcourse broad cast to all other connections except one from it is come, then pc's who don't have dest. MAC as theirs, would drop and the pc which has, simply get the packet,

But of thought it right, isn't it it would create unncessary traffic in the network, that is right, but it was feasible in such very small network,



Now as you know even if we use a single hub, it would be limited to certain number of end-pcs to be connected, so why not connect two hubs together, with this approach, what would happen, is that, from one hub's end, if a pc sends frame to the hub, the hub will send the packet to its own network, and also send the frame to the other hub, and then that next hub would send the frame to its' nework ,lik this,



`

And to increase the network, just connect one more hub, and then this would cause unnecessary traffic in other hub's network right, so very destructive approach, but fine with this small scale network isn't it?

Here, understand that devices are still communicating through layer 2 protocol, I.e using only MAC addresses

Sometimes there was a big distance between these hubs, so a REPEATER (with two interfaces) was created which would simply repeat the frames coming from one interface and send it to other interface with amplified signal, I don't think I need to give the diagram right?

Later to address the issue of HUB, SWITCH was created which was an intelligent device and which could work on layer 2 than HUB which worked at layer 1

Here, the workflow of Switch is simple, it would make a TABLE in which it will note down which interfaces are active and on which interfaces which MAC addresses(pcs) are connected, but when you first establish the switch in the network, it doesn't know the mac addresses of pcs and exactly on which interface one is connected right, so it learns from the initial frame distributions, where it first broadcasts the frames coming from one end, and as it knows that if a frame is coming from interface A, then the source MAC address from that frame is actually connected on the interface, so it would update the table accordingly, and when a computer replies from that reply as well it can infer by looking at the source MAC and then mapping it with the interface, so some initial broadcasting is required, but once it has completed its table,

It will forward the frame to only that interface where that particular MAC address is configured, but if a different MAC address is found, it will again broadcast the frame in hope that it is possible that a new computer has joined on some remaining interface or a computer is replaced with different one,

So please keep this in mind as this is required for us

So, please again understand, the switch will note down a table like this:

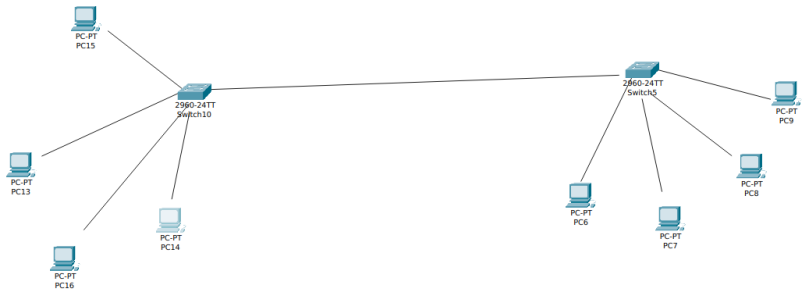
INTERFACE(abrev.	MAC (abrev.)
FA	14
53	DF

So here pc having mac address 14 is connected to the interface FA of switch

This is fine where the computers are directly connected to the switch, fine.

Later as switch is replacing the hub, there comes important thing, how to join two different networks of switches, as previously we saw a network of hubs joined using repeater, but It was just a repeater not useful for normal purposes, as immedialy the hubs were got replaced by switches due to their problem

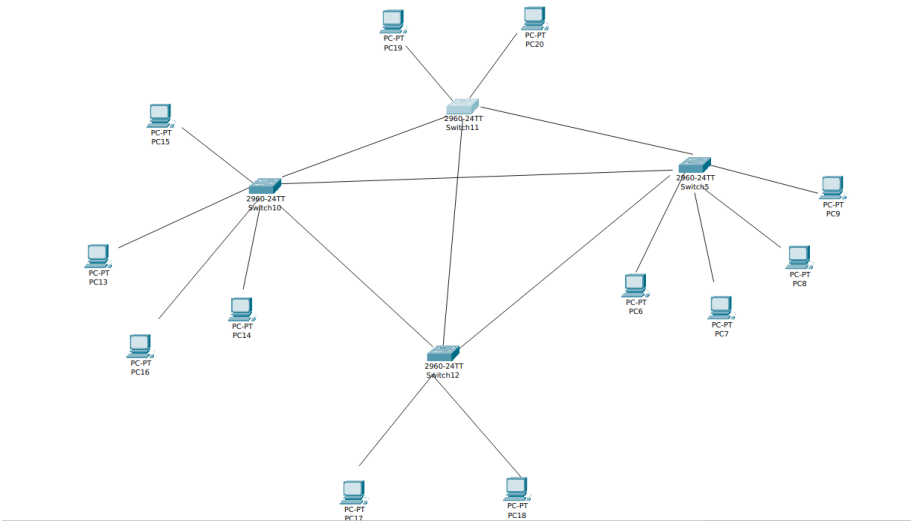
While in the case of switches, to connect long distanced switches, repeater was still used, but as you are going to know to connect two switches in local it is not needed so consider the scenario:



Here, as you can see, two switches are connected to each other, suppose inteface X of first switch is connected to interface Y of 2nd switch, so here, you can consider if both the switches are just installed, they don't have the so called table which maps the MAC addresses to the interfaces,

So what will be the mapping for the interface of 1st switch connected to 2nd swich?

Well this is where the real problem begins, here after considering this problem, first protcol makers didn't want the network to be flooded with broadcast messages, so broadcasting when MAC not found is not solution here right, well you might say, if the MAC address is not found to be on any interface the switch should be configured to route the frame to a default interface, here X, and the problem is solved and also the same for the 2nd switch which will send the frame on the default interface I.e Y when no MAC matches with the table configurations right, well this only worked when only two switches are considered and are connected to each other, so you would configure to send that frame and you will think to make some changes in the switch's firmware, but wait, is that change in the firmware gurantees that this will be useful to scale up the system, as only then change made will be considered successful. But there is a catch, what would you do in this case,



How would you configure default interface where your switch is connected to more than one switch?

You won't be able to say where exactly the MAC addressed' pc lies right?

So default interface fails here,

Lets consider other successful scenario which was indeed popular to some extent without modifying much of the switch's workflow, that is,

Append the MAC addresses from other switches to the interface where the switch is connected to that another switch,

BUT BEFORE THIS, LETS RESOLVE A DOUBT HERE, why mac address of switch is not coming here,as you can see when configuring default gateway or interface, we can see the interface of other switch is confgured on first switch, using the MAC of other's switch's interface, but well, this doesn't work, let me tell you,

Who generates the frames? End-devices right, and not the switches, so here it means that the frames can never contain the MAC address of an interface of switch anyway, so how does we configure default interface, we configure it manually above,I.e it should send it to that direction, as it can never know the actual mac address of the switch coming into play here, as it can only infer the MAC addresses from where the frame is coming I.e on which interface and what is its source MAC address, so as you know the workflow of the switch, it will initially broadcast the frame until it knows the addresses on the interfaces and will update the table accordingly,

Fine, so when a frame comes from another switch only then this switch will be able to recognise in case what if multiple frames are coming after one another, will it update continuously and then when a particular suppose 54 MAC is configured as last one, and then we generate frame for 56 and it resides in

the another switch's network, obviously this switch will again broadcast the message as it will consider only the latest configured MAC address on the interface and not all, this is the problem

i.e suppose, switch A here is connected to switch B, and switch B has MAC address table,

INTERFACE	MAC
54	FA
12	45
34	23
DF	98 (a pc that previously sent frame and thus this switch was able to detect and configured it, if another would have sent after that, its MAC would be here)

So here what you can do is simply copy the first 3 lines' MAC addresses and assign them to the single interface of switch A (12)

So table of switch A would look like :

INTERFACE	MAC
23	98
45	1A
12	FA,45,23

So here what you did is, when in the network of switch A, a frame comes with dest. MAC of 98, it will be forwarded to interface 23, and when comes for MAC of FA, it will be forwarded to interface 12, I.e to the next Switch, (WITHOUT KNOWING THAT BEHIND THIS INTERFACE THERE IS A SWITCH AS I ALREADY TOLD YOU THAT MAC ADDRESSES OF SWITCHES NEVER COME INTO PICTURE ANYWHERE)

Yeah, it will be working fine right then, as you can connect then multiple switches with this as well, and some interfaces of the switches will contain more than one entries for the MAC addresses,

But you might be thinking how this can actually be implemented, well it can be done manually but as we already said as switch initial works, it will do the same for above scenario and learn the entire topology and MAC addresses where are they and all when some broadcasting is done, as after this, it will be able

to infer that is multiple MAC addresses' frame are coming from a single interface, there might be a switch which is handling these multiple end-devices so it will append these MAC to the interface, and this will be done until it seems to be fine right, well this is quite good but what if in a switch B's network computers grow to suppose 100, then obviously single switch cannot handle so there would be another or more switches that would be connected to each other, while each one having appended addresses in their table as even are they in same network,

But then problem comes on switch A, as it is connected to one of that switch, and considering behind this switch, there are multiple end-devices, it would then need to append the MAC addresses of each device when it encounters a frame on this interface with this MAC as source MAC, but here,

The initial configuration will take time as there are much more end-devices, lets consider this is also fine, but there is also a problem of deadlock, where Switch A is connected to switch C and switch B, also switch C is connected to switch B, here,

When they are all started, suppose a computer in switch C's network initiates a frame, with source MAC X and destination as anything suppose Y, and suppose this destination Y is present in B, then when switch C broadcasts this frame, it will be forwarded to both switch A and B, here, switch A will be able to identify that the interface T (suppose) connected to switch C is indeed having a MAC address X, so it will append an entry into that interface T, and and as it doesn't contain the pc with this mac address, it will try to broadcast it again, as you know, it simply works on input output principle, if input doesn't get any path, it will broadcast to all output, and here, rather than sending it to the switch C as it was the input, it will send it to switch B, and what switchc B will receive is frame with source MAC X and dest MAC Y, but wait switch B already received such frame from switch C right, and with this, it configured MAC X on suppose interface W, and again it is getting the same frame from interface Q from switch A with source MAC of X, so isn't it double interface for same MAC,

Hence you cannot create a cycle of switch, but yes can definitely create hierarchy

Fine, so as you can see if an organization is using hierarchy of switches and there are hundreds of machines are there behind there main switch, I.e switch that is connected to outer ones, then other switches first need to wait for the broadcasts then do broadcasts then receive MAC then append ,

And this is for just one organization, if there are multiple and want to communicate to each other, woah!

More broadcasts and difficult to scale up this system right?

As each switch in the network need to have info about if this MAC comes, on which interface to pass on this frame, and with this if new organization joins even if we try to do manually, more difficult as there is no mechanism in MAC addresses that will let us create and infer something about them,

Try to think it, is there any way to make it better no way!

You might say if there is a relationship in the computers behind a switch, which we can consider and use that to forward the frame then it would be possible, but wait, what would you do with that?

The existing protocol will not let you do that as its workflow is straightforward if MAC comes and available on interface, send to it, else broadcast it and infer where it is available, well even if you are

able to find some patterns even it is impossible, this will break the normal frame distribution then, I.e you need to change the whole protocol right?

Well you might think this is working well right in local network, where there are limited computers and doesn't have problem here, as when you are going beyond the local network to a more broader network of switches, infering from switches and their network is the most daunting task, both manually and automatically and non scalable.

So this the time where the MAC and its protocols were made limited to internal and local network.

So this protocol works well behind a single switch right? Right yes? Well you are definitely right!

So scientists and other researchers found that we need to establish some relationship between computers of an organization to establish them different identity than other computers worldwide, yes you heard it right, worldwide.

So Here the creation of Classful Addressing started !!!!!!!!!!!!!!!!!!!!!!!!!!!!

With this, the creation of IP addresses, a device that understands this protocol and routes (yeah!) on the basis of ip addresses was needed, so yeah you are right, Routers were made to handle this which would work at level 3.

Fine, now lets think how it would be configured in those days,I.e in the era of classful addressing.

Well what was happening is that when an organization require internet connection to be able to connect with others, they needed to find suitable range according to their needs, I.e suppose they have around 100 computers that need to be online, so they would consider class C IP range, suppose 194.20.20.0, and the IP generated from these are needed to be configured from the router organized by the organization.

Before actually utilizing the IP addressing schemes, supported operating systems were created first

********** IP CONFIGURATION IN THE ORGANIZATION **********

Now lets first talk how this router would initiate the talk between itself and the switch which is connecting multiple end-devices to each other here comes real work, you know in those days the IP addresses needed to be manually configured on each PC right,

So after manual configuration of IP addresses, suppose we configured the IP address for a pc and the pc now wants to communicate to the pc in the same network, I.e

Suppose pc with ip 194.20.20.4 wants to communicate with the pc with ip 194.20.20.10

How? As right now, it is NOT configuring the MAC address as it was doing before, I.e before we were giving MAC addresses of the destination pcs we want to connect to right, but right now, we don't know the MAC addresses of the computers in our network!

And there is a reason to this, as we already know having an IP address, we need to talk to the outside world right, then why not use the same to talk to the internal network, why remember both the IP and MAC address of the computers which are there in the local network right, this is to maintain consistency

Let me explain, suppose you are going to create an application which is requires internet, now what if you needed to tell it whether you are in the same network of the computer you want to connect to or in different network ,and according to that if needed to give IP or MAC, then it creates inconsistency right, so why not just use IP addresses to communicate even behind the Switch?

Well switch?

Well there needs to be created some protocol by which switch can understand where to forward the frame right? As it is our main device which is going to transfer the actual frame to the computer.

But how do we do that? How do we map IP address to MAC address?

Wait what I said? Mapping? Why?

Well understand this, lets take above scenario, PC A wants to communicate with PC B and has its IP address know to it, but know MAC address is known for this IP address right? No MAC address right?

Really? Yes,

PC A doesn't know the MAC address as it doesn't need to know, what the PC A needs to do is simply make and ARP (Address Resolution Protocol) Request and broadcast this request!

YES YES YES, you heard it right! Broadcast this request in this network, well how? You might know in earlier days specifying destination MAC address as 00... I.e only zeros, it means the frame needs to be broadcasted, and switches were known this frame, I.e in that case they would not refer the table for finding the interface, they would simply broadcast the message on all the interfaces,

So here the frame passed by this computer A has source IP as 194.20.20.4. and source MAC as suppose X and dest.IP as 194.20.20.10, and dest. MAC address as zeros

The wrong PC getting this request would simply drop this frame, as it will not be its IP address, even though It can infer the IP to MAC mapping of the originating PC!

Well suppose right pc gets this, and when it sees its IP address, it understands it is for us and sends response to only this pc by setting up the dest. MAC and dest IP of this source PC with its source MAC address, and also notes the mapping of the source PC

Now of course according to the workflow of switch, it will transfer the frame to the source PC, and then when source PC gets this frame, it will understand that it is the response to its ARP request, then what it will do is that after this it will save the mapping of IP to MAC and then do the normal request and response of the data that it wants to send and receive using the protocols

Here you need to understand that even if they are in the same network they are still using IP addresses even after successfully knowing each other's MAC addresses right, well this is to maintain integrity of the system, as even when the frame has info of IP addresses, it doesn't affect MAC addresses and thus, switch is able to route the frame correctly.

Now why did PC A made ARP request and why not simply sent it to the router?

How did computer knew that destination is in our network itself?

Well it is according to the Classes, if the destination IP address belongs to the same network as the source IP, I.e in this case source IP was 194.20.20.4 and destination was 194.20.20.10, and programmatically it is definitely possible that if source and dest reside in same network, make and ARP request and then proceed with the MAC address mappings,

i.e if both resides in the same network, it source PC will first try to find IP to MAC mapping for that dest IP, if found it will directly make that request with the MAC address else make an ARP request to find it first then send the request.

Fine, first clear a doubt, a router as we all know right now, also has IP address right?

Then suppose the PC finds that the dest. IP is not in its network then?

Well what it will do is it will set the MAC address of router !!

Yes you heard it right, as that is the only gateway for connecting to outer world right?

So if the PC doesn't have the MAC address of the router, as usual it will do the ARP request right?

And as the router is connected to the switch, it will simply also get the ARP request and will respond to it as discussed here,

After this, the PC will get the MAC address of the router and will do what?

Will simply make the required request containing source IP and source MAC of it, dest. IP of the original resource, and most important dest. MAC address of router, to be able to successfully escape from the switch right?

Correct!

Ok the request coming to the router to make it to the external world contains what?

Source IP and source MAC of original PC

Dest IP and dest MAC of router

Now what will the response will contain?

It will contain Source IP of real resource , source MAC of previous router (don't worry we will cover)

Dest IP of original PC , MAC address of actual router

Now when router receives this packet, it knows that the packet is indeed for its network as that is why it came there, (will explain why it came there)

Then router will see that the Dest. IP is indeed the IP address which belongs to its network, then router will find the IP to MAC mapping in its table, and then it modifies the packet and adds dest. MAC of original requester according to table mapping by examining the dest. IP of the incoming response, and then makes source MAC of itself and then transfers the response to the switch which then as usual sends to the appropriate computer in this case original requester.

This is the entire workflow we seen BEHIND THE ORGANISATION, now lets go beyond...

BEYOND ORGANIZATION WORKFLOW:

Beyond the organization entire world sees the router of the organization as the router of the organization I.e the ip address of the router or the network of the organization,wait

But what does it mean that the organization has the IP address range,

Well that means that when organization buys the IP address range, it actually pays for the configuration that is done in the main routers in that vicinity, you mean it right, where the organization's router is going to get located matters, and in that vicinity, whatever is the first router is, it is then connected to the organization's router via some mechanism I.e typically through optical fibre and that router in the vicinity is thus able to communicate to organization's router through this interface, but is that the only connection made? No, this router is connected to multiple other routers in the vicinity in order to ensure that connection is made even if any of the router is got down, and not only that, to also route packets through other path if the current path through a router is congested or busy, this makes routing easier and manageable,

So we got to know that the routers in the vicinity wires up with the organization's router in order to route packets, now these routers to which the organization's router connects are main routers and do not contribute to any other organization's internal routing, I.e they are not actually involved in any internal routing at their end,

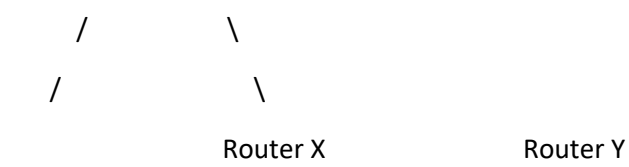
This concludes that our organization's router cannot be used for further routing purposes as it is obvious, if we use, it would congest the router with other packets as well, which will definitely congest the internal routing because of this, and organization hasn't paid for this hastle right, so this oraganization's router is called as END ROUTER which typically ends the packet's route and routes the packet internally using the MAC addressing by looking in the table or by routing the packet to the subnet (WE WILL TALK ABOUT THIS LATER)

With this set, as the vicinity routers are part of bigger routers' network, the nearby and thus virtually all routers update their routing table to route the packet coming with that ip address to the next hop accordingly with the routing protocols like BGP etc. , but exactly how?

Lets dive deeper,

When we wire up the vicinity routers with our organization's router, what happens is that those routers are able to communicate with router router, I.e lets take a scenario,

Suppose router A is our organization's router, and router X and Y are in vicinity, we now wire up them like,



WIth this approach, suppose interface x(router X) is connected to interface a(router A) and interface y(Router Y) is connected to interface b(of router A)

With this set, how can they communicate?

Well, it seems somehow complicated at first, as these routers are typically connected directly and to our knowledge they are routers right? Meaning layer 3 devices made to work only on layer 3 and don't care about layer 2 at all right, this means that they don't use layer 2 protocol while transmitting packets within themselves?

Well we are in the classful era of addressing and at those times, routers didn't have major hierarchy of routers, ok, lets make that aside,

First we already know that layer 2 devices I.e switches do not consider the MAC addresses of themselves as well as other switches as that doesn't make sense ovbviously right, while routers, the layer 3 devices also do have MAC addresses for their interfaces which is obvious, but they are indeed work like normal computers, I.e they are getting considered by the switch, as the ROUTER is the END POINT for the internal PC right, and also router has the capability to change MAC addresses (not IP in case of classful)

And then send the packet in or out the network

Now first understand that routers are also type of hardware and so they have NIC cards right?

Right?

So those NICs should work exactly like our normal computer's NICs should work isn't it?

In case of our normal computer meetup using MAC address, we only put the recipent's MAC address right, which was the original receiver which should receive the packet right, so our end goal was to send the frame to the computer we intend to, right? So that is the reason the MAC addresses of any interface of the Switch was not considered and didn't required at all right as the NICs of switches are different and

work differently as normal NIC would do as normal NIC would drop the frame if it doesn't see dest. MAC to be its MAC. as our aim was to transfer the frame to that device which can understand the frame and can process it, and be able to send us the respose right?

But here in case of routers even if they are directly connected they do have NIC's right?

And these NIC's work like the normal pc's NIC would work, I.e if they see that dest. MAC is not theirs' they would drop it, so what does a typical router does, it encapsulates the ip packet into the ethernet frame or any other protocol if being used and then sends the frame over the wire to the next router, and the NIC of that next router gets this and ensures that frame is indeed for itself and then strips the frame headers and extracts the ip packet and performs further routing.

And in case router doesn't know the MAC address of the next hop which is connected on the interface X, suppose, then it would simply send the ARP request on that interface only, and when response comes get that MAC address and store it, and then use that to send the frames created using that MAC address.

Now, we understood how a router typically sends a packet to next router right?

But how does it infer to which router to send the packet to?

Yes using one of the protocols we mentioned above, I.e what would the typical routing table would contain is simply routing info for that ip address like this,

IP(here Network)	IP to forward the packet to	Interface
194.20.20.0	40.20.20.1	eth0

Here, the IP actually gets down to its network from which it originates from, I.e in this case it is 194.20.20.0, and the ip to forward to, Is the ip address of the next router to forward the packet to right?

Yes, and that router is available on the interface eth0

Now what would the current router will do, yes, it will simply find in its ARP table whether the MAC address for that IP address is there or not, if not, it will make and ARP request and get the MAC address,

And once have, it will simply encapsulate the packet in the ethernet (or if other protocol in use) frame and sends through that interface.

Now you have understood how that routing is actually working right.

But wait, why is that router having the IP address, and Is that IP address routable?

Here routable meaning if I am sitting here and made a request to that IP 40.20.20.1, will that request reach that router and router will recognize yah it is indeed for me or it is just the IP address for internal routing between routers and my request would land on some other organization's router is available?

The answer is,

Yes these routers have indeed publically routable IP addresses and yes, when I send the packet intended to that router, it will indeed reach that router, the question is why is it needed as we currently don't see any need for the router to need to have any public ip address right?

Because the main work of the router is to just get the incoming packet and find the right next hop for the packet right? Isn't that true?

And even the routing table updation protocols like BGP or some other typically only requires communicating with peers only to update the routing table isn't it?

So why do they need IP addresses in the first place as this work can be simply done with MAC addresses isn't it?

lets consider a scenario, where a router gets a packet from suppose another router, and then it strips the frame header and get pure ip packet, then it looks up in its routing table where to send that packet, in that routing table suppose the entry shows that this packet needs to be forwarded to the router having ip suppose X available at interface eth0, and then the router would encapsulate this ip packet as it is with the ethernet frame containing its MAC as source MAC and dest. MAC as next hop's i.e next router's MAC, so where does the ip address of these two routers came into picture here, it seems like they don't require ip addresses, as the routing table can just have information about on which interface the packet needs to be sent and the MAC address of who is connected on that interface, isn't it?

And simply using that MAC address to send that packet by encapsulating that packet into ethernet frame and sending it, isn't it?

Well this seems pretty right,

That seems fine, and it means that even the routers doens't require internal ip addressing mechanism, I.e ip addressing scheme between themselves which will not defer the real world, ip addressing scheme as I told above regarding packet transmission to the router, even that isn't required,

THEN WHY ROUTERS HAVE REAL PUBLIC IP ADDRESSES?

The answers lies in the configuration management and security, well understand why networking is done, in order to share data, manage data and manage systems, wait what, manage systems?

Yes, as you probably would know in those olden days of classful addressing, protocols like telnet and so were extensively used to manage computers remotely and securely(somehow) using their publicaly available IP addresses, so aren't these ROUTERS a type of computing device?

The answer is yes, of course, these routers run mini operating system often in the form of firmware which provides lots of functionalities and even remote login and configuration with some protcols, but why? The reason is suppose a country needs to ban temporarily most the connections going to a specific IP address, how would network engineers would do that? By manually going to those routers?

Of course no, by remotely managing them to include some firewall rules in those routers to allow them to work upon those rules and configurations.

And this also helps the already implemented ARP protocol that we previously used in internal network to find out who with has this IP and the actual holder would response with its MAC address isn't it?

So this can indeed be used over here in case of routers as well, and when we configure the router with that public ip, indeed we cannot assign that ip to any other.

So with this you might have got the reason why to use public IP addresses for the routers which are actually not part of any organization or so.

WITH ALL THAT SET, LETS MOVE INTO THE INTERNAL NETWORK OF ORGANIZATION AGAIN!

Now after some time period, the organizations were finding the single level I.e single router and multiple pcs behind it to be inefficient, but why?

They wanted a more robust solutions where they would require sub-networks within that network to manage them efficiently, but why?

Let me clarify this, when you have only one router and need to wire up the switches from that router, it is daunting task and more importantly, you need to stack up the switches in order to fulfill the actual interface port requirements and managing this infrastructure was cumbersome and daunting task as multiple switches needed to be stacked up on one another (actually stacking here means connecting via a single cable so that it other can get access to the router),

This is indeed complicated as in case of 1000 hosts, around 50 switches are required in order to fullfill their requirements and stacking them up on one another Is I think a nightmare and maintaining is something which cannot be thought of right?

Well to address this issue, subnetting was introduced, where multiple internal routers could be used as subnets which will handle this issue, but please remember in those days, ip addresses needed to be manually configured INSTEAD of relying on a protocol like DHCP which didn't exist in those days.

So what happens in this subnetting is that some host address bits are considered in the network address bits and then routing decision was made

But wait,

Subnetting is done inside the network to configure and manage individual subnets right,

This means the packet coming from outside doesn't need to know anything about internal network infrastructure right?

Yes obsolutely!

When the packet comes at the main router, the router then has to decide to which subnet it belongs to,

Again before moving forward, I would like to tell that even this subnet information needed to be configured manually in each router.

FINDING THE RIGHT SUBNET

There are various programming Technique by which it can found which subnet the packet belongs to, obviously you can find on yourself,

Suppose network 194.20.20.0 is subnetted into two networks, So

194.20.20.0 to 194.20.20.127 the first subnet and

194.20.20.128 to 194.20.20.255 the second subnet,

And one programmatic way to find whether the packet ip address from that packet belongs to which of these is to simply compare the thresholds right, with 194.20.20.127, and if found less compare with right one I.e 194.20.20.0, and obviously more, then the packet lies within this subnet, then declare this packet to be of this interface and send through respective interface,

But what if we have hundreds of subnets, and the ip address the packet bringing somewhere midway of the ip range, so router would need to compare the ip to all the thresholds before this midway and if it actually belongs to last subnet, then redudant comparisions for single packet would cost tremendous time and which is not feasible the simple approach is with subnet masking

The subnet mask for the network from this router point of view (will explain why point of view) would be 255.255.255.128

And by simply doing bitwise anding, between subnet mask and the ip address, it will find the subnet address to which the packet needs to be send,

And this routing you need to handle manually I.e configuring routing table which contains information about the next router to forward the packet to,

As you probably know in the routing table of the external routers we talked about, we talked about strip down the incoming packet's dest IP to the network it belongs to and then find out whether an entry for that network is there in the routing table, and then sending the packet to the next hop, so according to it, our routing table would look like,

IP (sub-network) IP to send to Interface

194.20.20.0/25 194.20.20.2 eth0 (as 194.20.20.1 would be taken by main

Router as it takes the first ip from the first available subnet.

194.20.20.128/25 194.20.20.128 eth1

So in our case for suppose, a packet comes with ip address 194.20.20.44, the main router first do masking,

194.20.20.44

&

255.255.255.128

This will result into a subnet or subnetwork and with the information from the routing table,

It is clear that we need to transport the packet through the eth0 interface while the ip address of the router doesn't actually play any role here, but it is important for security purposes and tracing as well as for ARP.

After this the next router however configured can send the packet directly or can peform further subetting on its end, that is why subnetting is router specific or you may say specific router point of view, here this router can further divide the network into two using following routing table:

IP (sub-network) IP to send to Interface

194.20.20.0/26 194.20.20.3 eth0 (as 194.20.20.2would be taken by this

Router)

194.20.20.64/26 194.20.20.64 eth1

And yes other entries will not be there as this router is only going to have the ips in this range,

Then this would be getting to 2 more other routers which will eventually transfter the packet to the original host

When we configured the subnetting one at main router and next at another router, we hot vertical hierarchy of routers like

R

R R

r r r r

Like this,

But if we wanted to subnet into 4 at root level only then it would have been

R

r r r r

Like this,

Some organization require first approach to manage the routers efficiently, while the 2nd approach migiht work well for small ip ranged organizations, what about big ip ranges like class B and class A networks?

Well managing them with only one hierarchy like in 2nd case if infeasible right?

Even if you did thousands of subnets, connecting them to a single router is not a simple task, or somewhat impossible as well, so we do require the first vertical hierarchicall topology of routers to configure for larger networks like this, by going to divide half and sometimes more, to manage the network efficiently.

Now another question arises here, when a pc sends a packet from this internal network to the

External network how does router transfer this,

When the packet comes with dest. IP of something different, the routers are normally configured to lookup in their routing table to find out best possible next hop for the packet to send it, but if that packet doesn't belong to this network, the routers in the internal network I.e subnetwork are configured to send the packet up the hierarchy until the root router isn't reached and the the root router would do its job.

In other case I.e the pc wants to communicate to the ip in the same network, the routers are configured in that way until the matching entry is found, I.e those packets would be destined upwards in the hierarchy until suitable path isn't found,

i.e the subnet mask for the packet will be calculated at each router, and if there is no match found I.e no matching subnetwork found, it will be transferred upwards in the hierarchy, if the ip address is from that network, then indeed it will match a subnetwork at some point, at most at the root level router, and it is impossible to not find it there when packet has come this so far, and I think entirely different ip addresses which are destined outside the network are transferred like this, so that it doesn't find a proper subnet at the root level router as well, then router finally sends it outside according to its routing table, it finds where to send it,

This is how the organization's router work, when other routers in the vicinity send the packet to this router on the basis of their routing table, for that network, when this router gets this packet, it has already an entry in its routing table for its network and the interface regarding it which goes into the internal network, and indeed when it checks the packet's destination ip and finds it belongs to the network which is available on this interface, it simply sends the packet to this interface, but here at this point, it doesn't know to whom to exactly send this packet to right,

As if already subnettings was done, then in the routing table of this router, subnetted networks would be present and then according to the right one, the packet would be sent to the internal router with the normal flow, as this router would know the MAC address of the internal routers and those internal routers' ip address would be there in the entry of the subnets,

But if subnetting is not present, and the router identifies that it is indeed the part of the network, there is no other way to find out that pc with this ip and find its MAC and send that packet directly through the interface available, as the routing info will indeed be present in this router, but just network to interface mapping and not any other ip address, so this signals that there is no more router involved here and directly need to resolve this dest IP to MAC and send it accordingly.

This is how to organization's router I.e main router is configured, it has routing table entry for its own network to an interface, but with empty ip address, showing that there is no more router involved and send that accordingly, as if ip address was there, it means that some router is involved in the front, and it then passes it to it, it might be just another router or network's internal router as well.

As you already know how does a router work right?

What it does is it simply takes the incoming packet, sees its dest. IP

Lets say it is class C address 194.20.20.4

As you probably know that the classes are created according to the first 3 bits considered, so what router does is,

Simply checks the first 3 bits of the ip address and determines its Class

In our case the first three bits would be 110

Ok, what after finding the class?

Each router is configured with subnet mask of the default class,

i.e for Class A, the subnet mask is 255.0.0.0,

For Class B, the subnet mask is 255.255.0.0,

For Classs C, the subnet mask is 255.255.255.0,

Right?

Yes, and after this, router simply perform bitwise anding between the ip address and the subnet mask of its corresponding Class

Why? To simply get the network address or simply network of the IP to which it belongs to,

So in our case,

194.20.20.4

&

255.255.255.0

This bitwise anding would result out 194.20.20.0 right?

And that's it! This is our network right?

Yes, and now What router does is simply checks for this entry in the router Table I.e entry for this network,

If it finds the entry, it finds the interface to forward the packet to and the ip address of the router connected to that interface, and with that it can find the MAC address and simply forward it to that interface,

Fine,

Now note this point as it is going to be important in the CIDR.

CLASSLESS INTERDOMAIN ROUTING (CIDR)

Woah! Pretty long though! We have reached the CIDR concept, Well there is nothing much fancy here, as we already have seen lots of stuff already,

And we are living in the era of CIDR, where you no need to use the concepts of Classful addressing and so,but wait, in the era of networking,

Classful Addressing is Foundation and,

CIDR is a building

So you got the point.

So, here in the CIDR, what happens that organizations need to buy the ip addresses only according to their need and from any range, I.e not at all limited to 10.... or 192... or anything,

You can take ip address from almost any available ip address and tell them how much public ip you require, right?

Yes, Now lets work with an example,

Suppose I wanted to buy hundred ip address for my Organization and I love ip address to be in 10s range, and can even buy hundred ip address from any range I.e 190sor 192s or 193s or 120s or literally from anywhere if they are available !

Suppose I told the IANA (Internet Assigned Number Authority (I guess it is correct)) that I want hundred ip address from 10s range,

Now first of all, when I told you that I want hundred ip addresses, no matter CIDR will provide me those, but not exact!

Yes even though it is CIDR, it is bound by subnets, and it cannot simply assign an organization the exact number of ip though it tries its best to make that possible,

But here, we will be getting 253 (out of 256 one for network defining, one for router itself and one for broadcasting)

These ip addresses are which which I can actually use for my purposes, (and this can go lower if I do subnetting in my network)

So suppose I got the ip range 10.10.4.0,

Right? Really right?

Yes of course!

Don't go into Classful addressing!

This is just a simple ip range,

If I have been allocated ip range 10.10.4.0

Obviously, ip range for me is 10.10.4.0 to 10.10.4.255

So, when I bought a router, I allocate it the ip address 10.10.4.1 (mostly) to it

And the router is then going to get connected to the other routers in the vicinity and thereby they are going to get configured to router any packet that is destined to my network to my router.

But wait, how does the vicinity routers be able to define that the packet is indeed mine,

Well understand,

The vicinity router needs to have entry like this in its routing table,

IP (network)	IP send to	Interface
10.10.4.0	40.10.10.2	eth0

Suppose a packet containing dest. IP 10.10.4.22 comes to a vicinity router, how does the router find the network of this ip?

As there is NO Class! then how?

Yes, with the Subnet mask! But then how?

Actually when an IP address comes to a router, the route matches that ip address with the longest prefix in the routing table, wait what does this mean?

Lets take an example here,

First lets take a simple example, where we require around 50k ip addresses, and we choose this network,

198.40.0.0

Here 16 bits are for network and 16 are for hosts,

And a router would configure this network as it is with the prefix I.e 16

i.e 198.40.0.0/16

Now suppose a packet 198.40.23.40 comes to this router and the router has following routing table:

IP (network)	next hop	Interface
10.4.10.0/24	something	something
10.10.0.0/16	another	another
198.42.15.0/24	different	different
198.40.0.0/16	ofcourse	ofcourse

Here according to the longest prefix matching rule, when we compare the incoming ip

198.40.23.40 with 10.4.10.0, not even bits in the first octect match, so discard it

Next suppose 198.42.15.0, here it matches upto the 14th bit as the 15th bit would be 1 in this network case as there is 42 ,and in case of our ip it would be 0, so 14 bits matched

Next we consider 198.40.0.0, here, entire 16 bits match up here, as the first two octects as you can see are common here, so the router thus determines this is the network of this packet and sends this packet to the interface.

Now lets take a difficult example

FIRST OF ALL UNDERSTAND THAT IN CIDR, everything works according to our need,

So if we say our requirement is so and so, and that leads us to have suppose hosts bits to use upto 9 bits I.e according to our requirements I.e suppose we require around 500 publicly routable ip addresses,

Then it is obvious that instead of 8 bits for the host address which would address upto 256 hosts(max)

We need 9 BITS in order to address around 512 bits close to our requirements

So we just make sure that our network will have 9 bits as host bits, and as a result the prefix to our network would be 23/ right, yes!

But here, which network we choose, well the combination can be anything

xxxxxxxx.xxxxxxxx.xxxxxxx0.00000000/23

Here the x positions can be in any combination, and those are the combincations which lets us have our desired network with 9 bits for host,

Here,

Possible networks within 198.20.something.0 would be

198.20.0.0/23 (of course this is a valid network)

198.20.254.0/23

Here if a packet with dest ip 198.20.255.34 comes, it will go to this network, as according to prefix rule!

What about

198.20.60.0/23

And ip range from 198.20.60.1/23 198.20.61.255/23

This means that any other organization cannot have ip range in that segment, you got it right

This is fine isn't it?

This is how the CIDR works, and the most imporant thing here is that the routers in the vicinity of the organization's router just get this routing info by using some protcol and then simply as line like

IP network	ip(of router)	Interface
198.20.60.0/23	40.23.1.2	eth0

Here, when a packet arrives at this interface with ip 198.20.60.145 or with 198.20.61.23,

The router starts finding out the longest prefix in the routing table to match up with this, and of course, our above entry matches the most I.e upto 23rd bit and as this will be the only longest prefix found to be matched with the ip address, the router will then consider that this packet belongs to this network and then forward this packet to that interface to that router

Fine,

If I ust bought network 10.10.1.0/24, then an organization requiring this entire ip range as 10.0.0.0/8 can't buy this ip range, right, as i have some portion of it,

So This works like this.

The IANA needs to be very careful about allocating ip addresses in certain ranges right, as they might get a client in future which require a bigger range of ip addresses and just allocating ip range to already existing clients here and there would cause issue laterwords

The Internet Assigned Numbers Authority (IANA) and regional Internet registries (RIRs) play a crucial role in allocating IP addresses to organizations. They follow certain guidelines and policies to ensure efficient allocation and management of IP address space.

When allocating IP addresses, they need to consider the future growth and requirements of organizations. They aim to allocate IP address ranges that minimize conflicts and allow for scalability. This involves carefully planning and assigning IP address blocks to different entities, taking into account factors such as the size of the organization, their projected growth, and their specific needs.

By following these allocation practices, the IANA and RIRs try to prevent situations where overlapping IP address ranges cause conflicts or hinder future allocations. However, it's important to note that IP address management is a complex task, and occasional challenges or conflicts may arise. That's why

organizations are encouraged to plan their IP address requirements carefully and communicate their needs to the appropriate authorities to ensure proper allocation.

Now managing the packet on the organization's router comes into play,

When the vicinity router sends the packet to the organization's router, this router sees the dest IP and checks its routing table where to send the packet,

The router may have single or multple entries based on the number of routers it is connected to internally or subnetting done at this router,

In our case suppose ip packet 198.20.40.134 comes to the network 198.20.40.0/24, then the internal routers if are only two I.e only the subnetting done at this level is two, the ip packet will get matched with the longest prefix again

Here the two subnets would be 198.20.40.0/25 and 198.20.40.128/25

The ip address will match the first one with 24 bits and the other with 25 bits, so the router will send it to the second network

And suppose if the network is 198.20.40.0/23

I.e total 512 ip addresses,

On this router if the same packet I.e 198.20.40.134 comes,

And if the router has two subnets there,

The subnets would be

198.20.40.0/24 and 198.20.41.0/24

Here the router can use subnet mask to determine where the packet should go, as there is not need of longest prefix matching here

So here our subnet mask would be 255.255.255.0

Then do anding and get the right network and send the packet that is all!

Well you might think why we are able to use subnet mask here, the answer is simple, as we have to deal here with only one network I.e ours right,that is why subnet mask is fixed

But when we consider other vicinity and farther routers, they have to deal with multiple networks and they have bunch of networks in their routing table with varying network bits, so there obviously cannot be a single subnet mask for all !

While checking the ip address with the multiple entries,

Suppose 10.1.20.0/24 network is there and

10.2.0.0/16 network is there in order for router to understand where to send the packet to,

And suppose IP 10.1.20.4, here you are not getting any feel of using subnet mask right?

Of course as there are multiple networks which might start with something like this right?

Here the obvious way comes it match until you get I.e longest matching prefix

Which in this case comes 10.1.20 here right?

Then do the normal workflow

So remember Subnet mask here in CIDR is only limited to internal organization's network, to split up the network in different subnets.