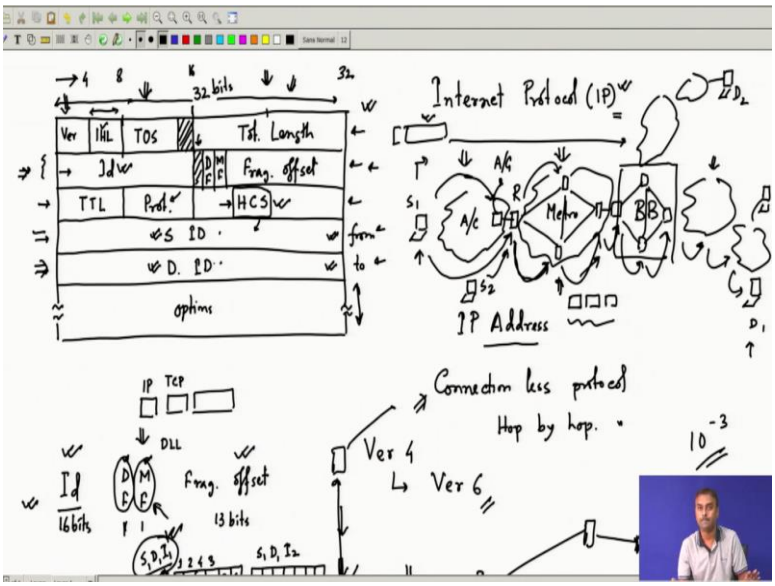
Communication Networks
Prof. Goutam Das
G. S. Sanyal School of Telecommunication
Indian Institute of Technology, Kharagpur

Module - 12
Network and Transport Layer
Lecture - 55
Internet Protocol contd

Start sir.

So we have started discussing the Internet Protocol and the IP layer. We have already started discussing the header. So, today what we will try to do in that header is some portion we have already discussed, and some portion is left, especially the addressing part and the fragmentation part. So, that is something we will be discussing next ok. So, let us try to see.

(Refer Slide Time: 00:53)

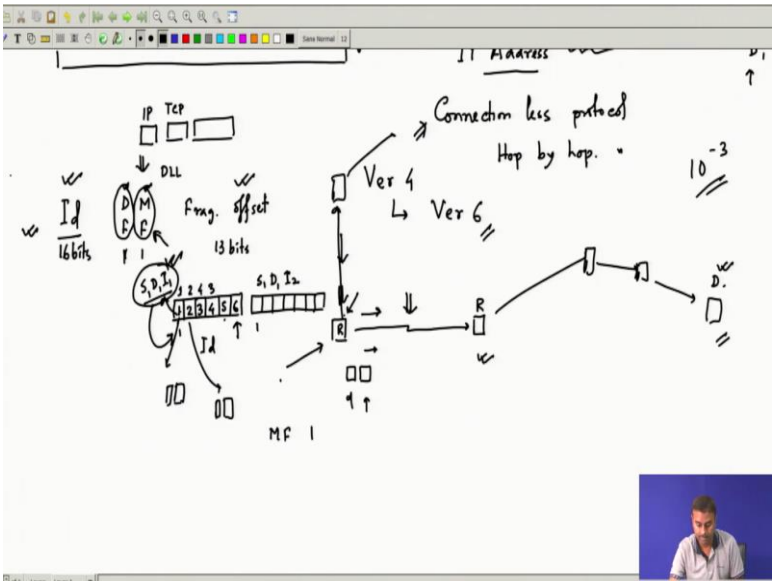


So, this was the header in the last class we were discussing about ok. So, in that header, if you see in the second row you have an Id ok. So, basically, you have an ID of 16 bits and then there are two fields where one particular bit is not being used. So, that is left for future use.

Then you have a 1-bit field of D F. So, that is called do not fragment. Then there is another bit called M F which is more fragmented and then there is something called

fragmentation offset ok. So, this is a this is 16 bit, 1 bit, 1 bit these two are flags and this is 13 bits ok. So, let us try to see how it operates.

(Refer Slide Time: 02:00)



So, what we are talking about is something like this a particular router. So, this is a router intermediate router that gets a long IP packet, it knows that from the routing table, it has decided that I have to forward it to the next router. This link's physical layer is not very reliable.

So, he has to make the packet size smaller or it might have other restrictions also because of that there might be a smaller packet size because this might operate on other network which has a restriction of smaller packets. So, this might also happen. So, if that happens then he has to there is a maximum size of packet length that can go through this link, then he has to actually fragment those packets of that size.

But, the problem is whenever this particular packet fragments it will generate multiple small fragments, but now these fragments when they come over here, this fragments has to be again regrouped with proper order.

So, this is something whenever you do fragmentation this is something which has to be facilitated. So, that facilitation is done by these bits. So, what happens? So, basically, any time you do fragmentation it might happen that he has multiple packets ok which are coming one after another and he has to definitely fragment all of them.

Now, a fragment from here and a fragment first fragment from here might be confused. So, he might jumble up these two things because these fragments might wait arbitrarily longer time in his because it is a store and forward. So, you first store them. So, multiple packets might be stored, all their fragments might be stored. Now, these fragments if they are not properly numbered and identified might get jumbled up.

So, therefore, first I need to have an ID of which packet I am fragmenting. So, every packet that you will be fragmenting has a unique ID that will be given over here with 16 bits. This 16-bit provides a unique ID for a particular packet which will be fragmented. So, this is something you will be always doing.

So, you provide an identification number for that particular packet. So, remember you might be thinking about who will be providing this ID. So, this router can itself provide that I no problem in that because or it might be also given by the source, that is also possible. That source can start putting ID because you might be thinking that this ID might get confused. Suppose this router is getting packets from one source and another source is also coming.

Now, if these two sources give the same ID to two packets will there be a problem? No, there will not be because there are IP headers means there is an IP header where this source ID and destination also is there. So, if I combine this source ID, destination ID, and ID these three things will be unique for a particular source-destination combination.

So, basically for every source destination, if every packet I keep on giving ID, one after another no problem with that. If they are means they are identified so they can be fragmented accordingly, the same ID has to be put in all subsequent IP packets. So, whenever you fragment all subsequent packets must have IP headers because otherwise you will not be able to do routing.

So, therefore, those same IDs can be copied. So, basically, this has a source ID, let us say S 1, destination ID D 1 and this has I 1 as the ID. This will have again S 1, D 1 and this will be I 2. So, subsequently all these packets here you will be putting these three combinations and putting the IP header.

So, all this small small fragments will have their own IP header, second fragment also will have its own IP header. So, you keep on doing that. So, the entire packet fragments

into multiple fragments and you add the corresponding IP header which is identified by a unique identifier source at the destination ID of course.

So, therefore, there is no possibility that anything coming from other sources and other destinations will be getting jumbled up with this ID because they will have they might have because each source will be uniquely generating ID. So, one after another in a sequential manner. So, when they generate I 1 followed by I 2 followed by I 3 this guy also puts I 1 followed by I 2, I 3. Even if they actually coexist in the same router they will not be confused because they have different source IDs and destination IDs.

So, with fragmentation, I will never have a problem. I will be able to identify which ones are for this source, this destination I will take all of them together, and then I will try to see who has I 1, who has I 2, and then accordingly I can I can actually manage them.

Now, the problem is what is happening? Now, all these packets now will have the same specifications S 1, D 1, I 1, but they might get jumbled up I can put this in 1st, this in 2nd, this in 4th, and this in 3rd. So, this will jumble up the whole thing. So, I need to identify their order also inside the fragment within one particular packet inside the fragments among those fragments what is their relative order. So, that is where this fragment offset comes into the picture.

So, fragment offset specifies where this particular fragment starts in the sequence. So, whether this is the 4th fragment, this is the 5th fragment or this is the 6th fragment. So, it will be able to specify that accordingly seeing those things ok you will be able to join these data together. So, fragment offset actually tells you that.

And what are these flags? This flag has something called do not fragment and more fragment ok. If you fragment there is a problem. What is the problem? If you fragment the packets packets will be forwarded very nicely. So, wherever one router does the fragmentation. So, now, onwards the packets have been fragmented. So, these small packets will be delivered through the next router to the next router to the next router finally, they will reach the destination.

Now, the problem is if the destination cannot defragment the packet. So, basically, if they are fragmented then you have to identify these things whose source, destination, and ID, and accordingly you have to club them. After clubbing them you have to see those

fragments offset, accordingly, you have to rearrange them and then construct the whole packet stripping off all the corresponding IP headers and then delivering it to the next transport layer.

So, doing these things sometimes some destinations might not have this facility. Therefore, the source will know about that. If he is transmitting the packet to the destination he will also know the capability of the destination. How does he know? We will understand that from our description of the transport layer protocol because the transport layer is end-to-end. They can first talk to the destination and construct a connection knowing the capability of the destination.

So, if that is the case because the source knows it, he might set this flag. If he sets this flag; that means, if he puts 1 into that none of the routers can if he sees that; that flag has been set, he will not be able to fragment that packet. Now, you might be asking what can he do? Because this is a wireless link, if he does not fragment that packet, most of the packet will be lost there will be erroneous.

And then he has to he might have to do it multiple times he might have to retransmit and never it will go through ok? So, what he can do? If he sees that that link requires this is the local link which requires a lower packet size, but the packet that has arrived that requires means that is requesting that do not fragment my packet.

Then you actually search for other routes. Do not take this route this might be the shortest route if you go via this the destination might be reached in a smaller number of hops or with a lower delay, but you can always redirect it into some other router and he might redirect and go through a longer path to the destination, but you do not have to fragment it.

So, basically, you choose those parts whose paths are more reliable and where fragmentation is not required. So, you can always do that. So, if the do not fragment bit is set then you can do these things. What is this fragment?

So, this is another thing that tells you one by one whether you have some more fragments that have to be appended. So, this helps the destination to actually defragment the packet. So, from the packet suppose you have constructed these fragments. So, this is my fragment 1, fragment 2, fragment 3, fragment 4, fragment 5, and fragment 6.

So, first five fragments you will be setting this more fragment to 1; that means, in that fragment whenever that packet arrives you also know there are some more fragments that are waiting. So, you have to actually wait for other fragments. Only when you receive the last one that is where the more fragment bit will be not set; that means, it will be 0. So, then you know that ok I have reached the final fragment. So, that should be it and I have got the whole fragments, and now I can join the packets together.

So, like this, as you can see this actually facilitates the entire network, all the associated routers and the destination along with what the source specifies this helps the entire network to handle the packets in between. User do not have to care whether in between he has a particular means link where it is wireless, where it is not that reliable.

I might have to make smaller packets. So, all those things the user does not have to care about, the user can just do these things knowing the destination and his capability, the user can just deliver this packet, and then the IP layer does this whole fragmentation part to manage the data delivery smoothly ok.

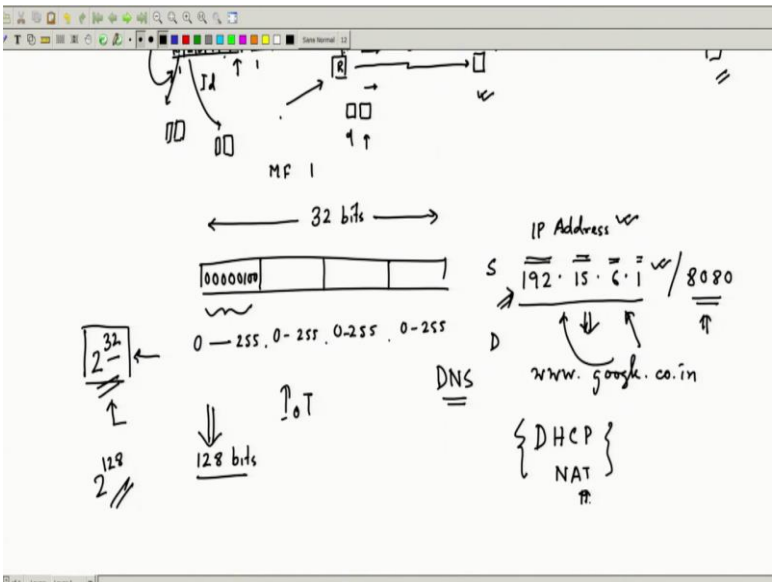
So, this is one way of appreciating how header specification and associated protocol; means, an associated algorithm that will be running that together helps source destination and in between nodes, helps some particular task of networking to be performed and that is exactly what is happening over here.

As you can see over here we have seen that if you have to fragment, then how do you fragment it? What are the specifications that you give to the fragmentation so that the other side without the intervention of the source can defragment the packet and generate the original packet? This has nothing to do with the data; data remains intact you just deliver it smoothly according to your network needs.

So, this is a network layer task that has nothing to do with actual data; actual data will remain intact whether you fragment it. Finally, you will have to exactly do the opposite thing to defragment it and make the intact means packet intact so that you can deliver it to the application there. So, this is something you could understand ok.

So, we have discussed fragmentation, next, we thought of discussing this id. Which are the unique IP addresses ok?

(Refer Slide Time: 14:25)



So, as you can see that is 32 32-bit address, this 32-bit address generally is subdivided into four 8-bit subsections. So, this is 8-bit. With 8 bits how many representations you can do? So, now we can see a decimal means whatever representation we will be having we can convert that into binary to decimal these things. So, you might have a particular realization as 0 0. So, it will be 8 bit 0 0 let's say 0 1 0 0, 1 2 3 4 5 6 7 8 ok. So, if you see this, if you take the corresponding means decimal representation accordingly you will be able to put the decimal representation ok.

So, what will be the overall decimal representation that can happen with this 8-bit? So, if all 0 it will be 0, if all 1 it will be 255 right with 8 bit. So, 0 to 255 some representation unique representation you will be able to build up ok. So, what they do each one will have this 0 to 255 each 8-bit.

And you put them just for representation that IP addresses are put in this with dot-decimal format. So, basically, I can write 192 dots 15 dots 6 dots 1; that means, it is a 32 bit, the first one has a binary representation equivalent to decimal 192, the second 8-bit has a binary representation equivalent to decimal 15, the third one 6 and fourth one 1 so on.

So, basically, if I give that accordingly we will be able to always calculate the equivalent binary 32-bit representation, but for our understanding, it is generally represented IP

addresses are generally represented you might have seen that this kind of IP representation, IP address representation you might have seen that ok.

So, this is how IP addresses are being represented. It will be always this with three dots and four decimal values. They will be ranging from 0 to 255 and whatever unique values you will be providing over here should be the unique ID. So, each packet as you have seen will have a particular source ID and a destination ID, where you want to send that ID you need to know.

So, basically, whenever you are sending a packet, of course, you will know your own IP address and that should be unique globally. So, you must have a unique ID nobody else is sharing that ID and your targeted destination is his IP address you need to know ok?

So, whenever you are connecting if you are doing let us say some of the older application layer protocols like FTP or let us say you do means this kind of means remote login and all those things always you will be seeing that supposed server you are trying to log in. So, you always give the IP address of the server that is exactly what you do.

Generally what we do; we give the IP address of the server along with we also give a port ID. Sometimes we write 80 or something like that ok. So, you might be seeing that whenever you are doing FTP or you are doing some other remote login. So, you generally put this IP address and a port.

This port is the TCP port, we will talk about that when we will be discussing about TCP. But, generally, this is what we will have to specify. So, basically, the transport layer and IP layer are both the things you will have to give then only you can connect to the remote this one.

So, the destination you want to connect then you have to specify his unique ID. If you do not know the ID then you cannot connect. You might be asking then how do we we do not know the IP of whenever we do this HTTP search or HTTP suppose we log in to a Google page we do searching we do all kinds of things.

There we do not specify IP addresses, then how this particular things networking goes on, is another thing. Even though you do not specify the IP addresses you specify

something else. Suppose whenever you have to go to a Google page you have to type that www dot Google dot if you are in India suppose you write co dot in.

As you can see this is also typed with a dot. So, almost similar to IP address and this is what happens we will later on we will see that there are means instead of having these numbers people have actually come up with corresponding names so that people can remember those things.

So, there is something called DNS or domain name server, and then this DNS inquiry and all those things from which we can actually adjust these IP addresses or from the name we will be able to find the corresponding IP addresses. So, whenever we type this it has a corresponding IP address. So, do not think that IP addresses we do not require. So, whatever in your http this one you are typing it has a corresponding IP address.

So, you are actually whatever you are typing you are accordingly if you write for facebook, if you write for let us say Twitter, or let us say whatever site YouTube always will have an underlying IP address. So, whatever name you give from that a corresponding unique IP address is there it will connect to that server only ok.

So, sometimes they might have multiple servers and then you go to a particular server and then it distributes it locally internally. So, you might not understand, but generally, it is always the unique IP ID of the other server that you or your targeted destination.

This will always happen. So, the IP address is always like this, destination ID you will have to specify otherwise no communication can go through. So, that is why always every header even if you fragment them these IDs should be there. Who is the source and who is the destination this should be always specified this 32-bit will always be there.

Now, because with 32 bit how many machines you can construct or how many machines you can have globally? 2 to the power 32 at max, this is a very big number of course, but still because these days almost every means machine has its own Id like we are we are going to the era of internet of everything right.

Internet of things we have already seen IoT; that means, everything every small device starting from your coffee maker to your refrigerator to your television everybody will be connected to the internet and they will have their own Id ok. There will be thousands of

sensors installed everywhere and all those sensors will have their IP address. So, if you have to give this many unique IDs then that is 2 to the power 32 is not good enough.

So, that is why IP from IP version 4 we have come to IP version 6, where these IP addresses are 128 bits. So, with this, we will be having a huge possibility of 2 to the power 128 unique IDs. So, that will be billions of this one and that will cover all the devices that you can think of right now probably.

So, that is why probably means that was the major drive going from IP V4 to V6 because IP V4 was not capable of handling that many global ID generations. So, remember there are authorities that actually reserve the right to distribute IP addresses because it has to be globally unique.

So, anybody can locally decide ok I will take this IP address you cannot with your own whim decide that I must have this IP address cannot do. Your organization or your service provider whomever you are taking the service from must reserve those IP addresses from the body authority who is responsible for distributing IP addresses.

So, you have to first reserve those things and then that has to be distributed among those users. So, this is very essential and you have to keep in mind that is what happens for IP addresses. These are very precious you have to actually mean they are costly also you have to pay for those addresses. Whatever you will be blocking you cannot just like that as many addresses you want you cannot block them because whenever you are taking these addresses those addresses are lost they cannot be distributed.

So, less number of devices can now request IP addresses as you can see. So, that is why there is a crunch of IP addresses. As we have talked about that already IP V6 has already come to the rescue it will come up with a bigger number of IP address pools because it has increased the IP address field.

But, before going over there as we have also talked about networking is all about adaptation and backward compatibility. So, there you will see we will talk about those solutions also you will see there are solutions where with a restricted number of IP addresses how you can handle them ok.

So, we will talk about this there is a protocol called DHCP Dynamic Host Configuration Protocol you will we will also talk about NAT these are all part of how you can actually with a restricted number of IP addresses how you can still manage it. Sometimes you can reuse it, but in a very different fashion we will talk about that later with NAT probably whenever we talk about this NAT we will be talking about that.

So, these are the things which will be happening. Even though we have a restricted number of IP addresses there are possibilities of handling it in a more intelligent fashion. You will also see how allocating IP addresses to a particular area helps in making the routing decision or helps make this routing decision a little bit easier.

So, next class, we will be discussing that that is more of subnetting how it is being done. So, how do you create a subnet, how do you create a subnet mask to facilitate the router to do routing in a faster manner? So, all those things we will see it is all about this handling this addresses.

So, that will be our next target. We will see how IP addresses have been classified it has also been classified and how they can be handled in a better manner. So, this is something which will be discussed in the next class.

Thank you.