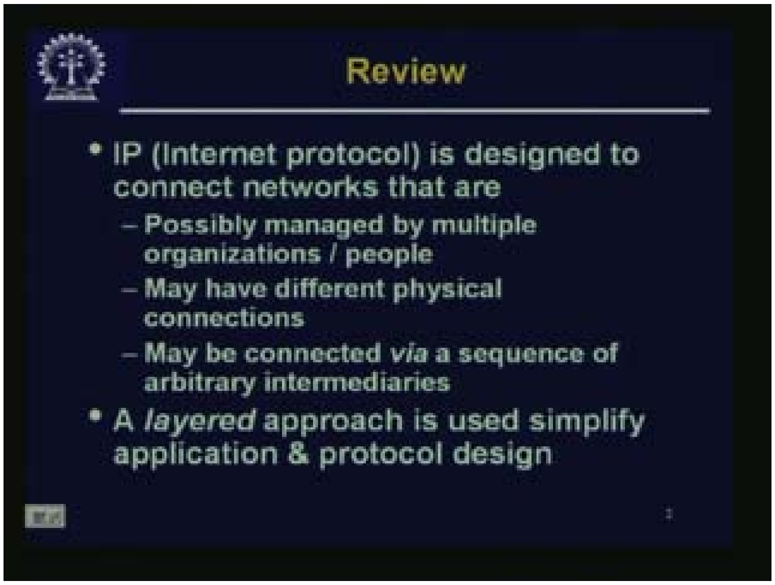
Computer Networks
Prof. S. Ghosh
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Lecture – 28
IP Version 4

Good day, we will talk about IP version 4 that is the internet protocol version 4. This internet protocol is really the network protocol of the entire stack and actually is at the heart of data communication. As it turned out that it became so successful that also other kinds of communication like voice, video, etc were also coming over to IP in a big way in many segments.

(Refer Slide Time: 01:26 - 02:50)



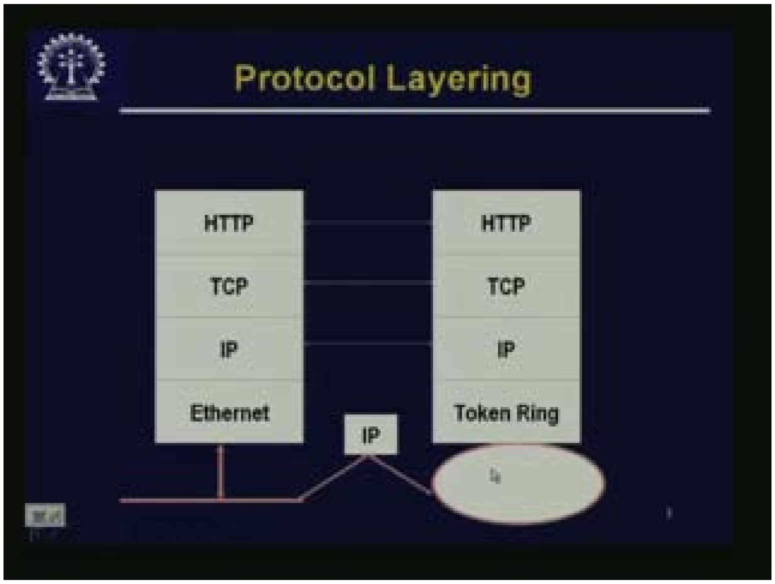
Review

- IP (Internet protocol) is designed to connect networks that are
  - Possibly managed by multiple organizations / people
  - May have different physical connections
  - May be connected *via* a sequence of arbitrary intermediaries
- A *layered* approach is used simplify application & protocol design

Today we will talk about the IP version 4. Just a quick review of the stack: The IP or the so called Internet Protocol is designed to connect networks that are possibly managed by multiple organizations or people. The internet we see today is a connection of network of networks. So it is of various networks and various networks are naturally owned by different organizations of people, managed by different people. But if they have to communicate they somehow have to come together and agree to one central network layer protocol and IP is that protocol, it may have different physical connections.

Naturally if there are different networks having different connections it may be connected via sequence of arbitrary intermediaries. Arbitrary intermediaries mean that when you are communicating from one computer to another these two networks also may not be directly connected they may go through other intermediate networks. So there may be number of hops before your communication reaches its destination. In the beginning we discussed about layered approach which is used to simplify the application.

(Refer Slide Time: 02:50 - 03:35 min)



Protocol Layering

HTTP

TCP

IP

Ethernet

IP

HTTP

TCP

IP

Token Ring

This is just an example. Let us say we have HTTP which is the protocol used in the application layer, it may use a TCP connection. TCP is another layer that is called the transport layer. TCP communicates with the IP which then may communicate with Ethernet. But please note that below the IP there may be a multiplicity of different data link layer protocols like Ethernet and token ring that communicates because of this Integrating Protocol IP which is common to both.

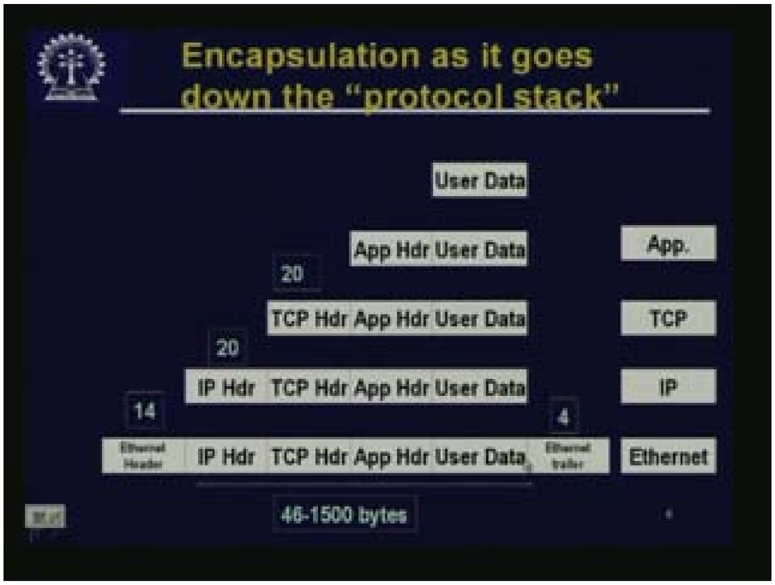
(Refer Slide Time: 03:35 - 03:44)

Hourglass Design

- Single protocol at network level insures packets will get from source to destination while allowing for flexibility

So it is a single protocol at network level that insures packets will get from source to destination while allowing for flexibility. We have the hourglass design like FTP, HTTP, TFTP etc, are different application layer protocols at the top. Then we have the transport layer and their two common protocols namely TCP and UDP. Both integrate to one single network layer protocol namely IP and this IP may connect to naturally different networks running different data link layer protocols.

(Refer Slide Time: 04:18 - 06:30)



Encapsulation as it goes down the "protocol stack"

User Data

App Hdr User Data — App.

20

TCP Hdr App Hdr User Data — TCP

20

IP Hdr TCP Hdr App Hdr User Data — IP

14 4

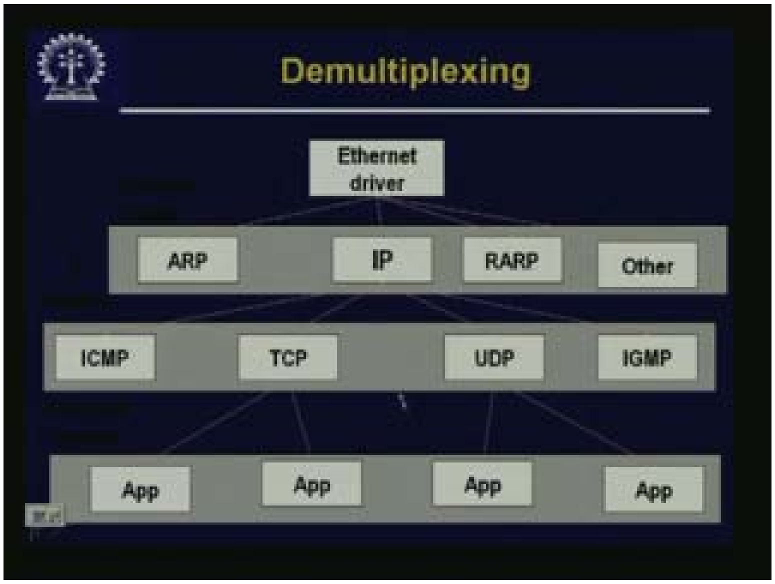
Ethernet Header IP Hdr TCP Hdr App Hdr User Data Ethernet trailer — Ethernet

46-1500 bytes

Just to look at the how the encapsulation goes once again, we had discussed this earlier. Suppose we have the user data being fed to some application then that particular application will have its own header. This header information is what is used for protocol between peers at the same layers. The application layer of this host will communicate with the application layer of the other host through this application header information and is passed to the transport layer TCP the transport layer protocol which is being used. Here the TCP header gets added and then it is passed to the network layer when it is coming down and when something is being sent where the IP header gets added at the network layer.

As far as this IP protocol is concerned this entire thing containing the TCP header, application header, user data etc is the payload. Similarly, for TCP the application header and user data together is the payload. So, whatever is inside the TCP does not really consider this. Similarly IP considers this entire thing as the payload and then it sends to the next layer which may be Ethernet as an example. Ethernet adds a header as well as the trailer. Some of the overhead we incur in this, for example, this TCP header is 20 bytes, IP header is 20 bytes, Ethernet header is 14 bytes and it has got a 4 byte trailer and then this entire thing is payload for the Ethernet which has a minimum of 46 bytes - 1500 bytes etc. This is how it comes when data is being sent and when data is being received it is in the other way, the first layer will take out these two headers and trailers, then the IP layer will take out the IP header, look at it and pass it up the TCP header and that will get stripped and finally the user data will reach the application layer.

(Refer Slide Time: 06:31 - 07:38)



Demultiplexing

Ethernet driver

ARP

IP

RARP

Other

ICMP

TCP

UDP

IGMP

App

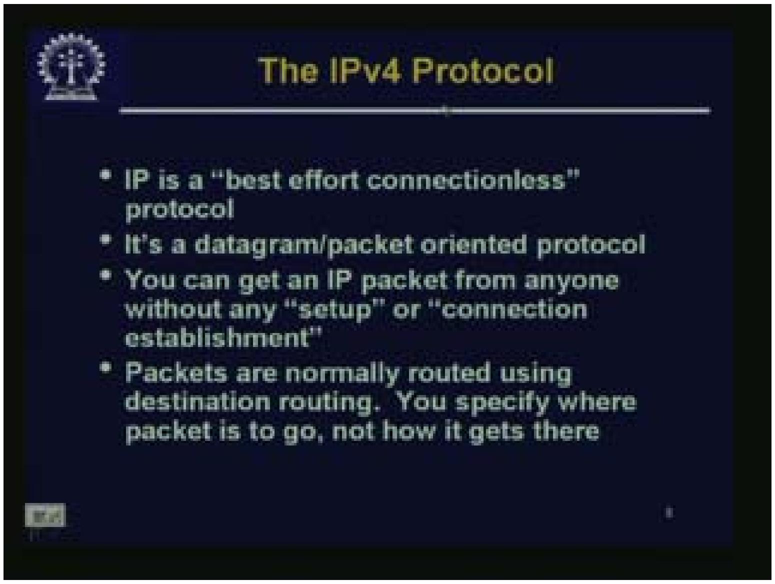
App

App

App

By looking at it the other way, suppose we have the Ethernet driver, this is really the lowest level in the tree that has been shown upwards. When something comes to Ethernet it has so many protocols in the next layer but these ARP is the address resolution protocol which is that given an IP address then finding the MAC address, this is the reverse of that as given the MAC address finding the IP address. IP addresses are used for communication within a same network but otherwise most of the traffic comes to IP. Above the IP layer this is TCP, UDP, IGMP used for multicasting and ICMP internet control protocol. Mostly the applications are for control and multicasting functions but the major part of the communication comes through this TCP and UDP and they connect to the various applications which may be running at the application layer and this how it is de-multiplexed.

(Refer Slide Time: 07:38 - 09:51)



The IPv4 Protocol

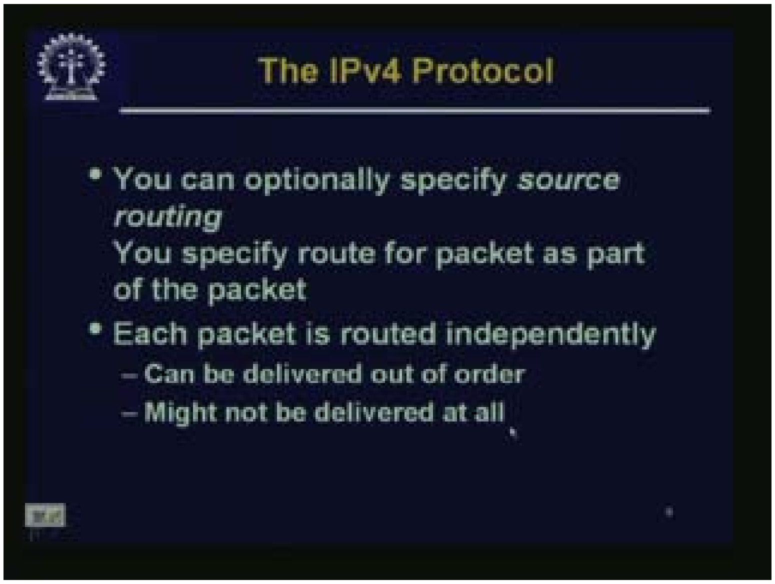
- IP is a “best effort connectionless” protocol
- It’s a datagram/packet oriented protocol
- You can get an IP packet from anyone without any “setup” or “connection establishment”
- Packets are normally routed using destination routing. You specify where packet is to go, not how it gets there

Just to summarize this IPv4 protocol IP is the best effort connectionless protocol. That means that the intermediate nodes will try to get your packet to the proper destination as best as possible but if it is not possible it will drop the packet. This does not give you any guarantee about the delivery of the packet at the other end but it is understood that all the nodes in between will make its best effort.

Suppose you consider a router which is somewhere in the network now it is receiving packets from so many sources and they are destined to many other different networks. It may so happen that because of the pattern of communication the router may get congested. So if it is congested as there are so many packets coming in then it cannot hold them in its buffer any longer so it is force to drop some packets. Although it will make the best effort but then this is not a guarantee. And then secondly this is connectionless.

You remember about connection oriented and connectionless protocols? For example, our telephone network was a connection oriented protocol but here this is connectionless. That means from the source to the destination there is no guaranteed physical or virtual connection between the two. And if you are sending a stream of packets from one source to the other some of the packets may take one route and then some take another route where all of them will reach their destination but they may get out of order. These are all part of the connectionless protocol. With a datagram or packet oriented protocol you can get an IP packet from anyone without any setup or connection establishment. Packets are normally routed using destination routing that means the destination is known and how to get to the destination is what is stored in the routing tables. You specify where the packet is to go now and not how it gets there.

(Refer Slide Time: 09:51 - 10:46)

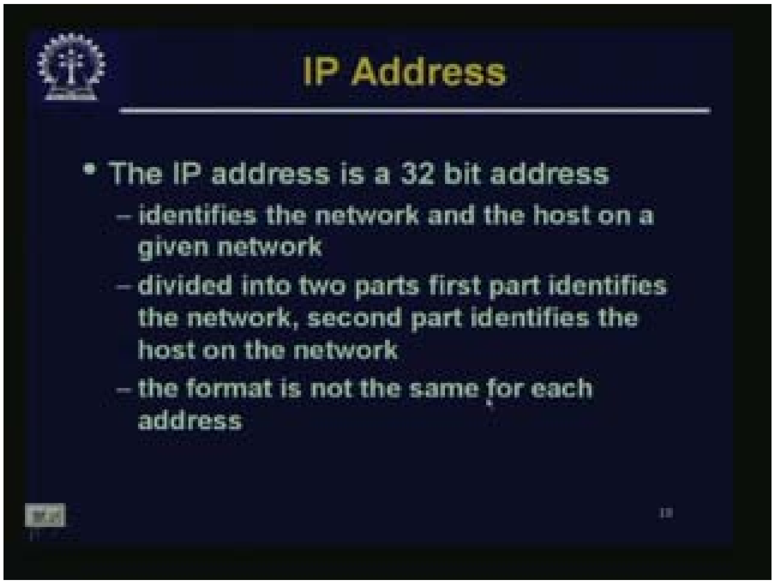


The IPv4 Protocol

- You can optionally specify *source routing*
  You specify route for packet as part of the packet
- Each packet is routed independently
  - Can be delivered out of order
  - Might not be delivered at all

There are some more parts. You can optionally specify source routing. In Internet Protocol there is a provision that in the source itself you specify that this is the route through which it should reach the destination. For example, if you remember our BGP uses source routing. But in general for internet packets it is possible to do the source routing but this is somewhat limited because the number of hops you can specify becomes restricted because of some limitations of IP version 4 protocol structure. Each packet is routed independently. That means they can be delivered out of order or might not be delivered at all.

(Refer Slide Time: 10:46 - 11:19)



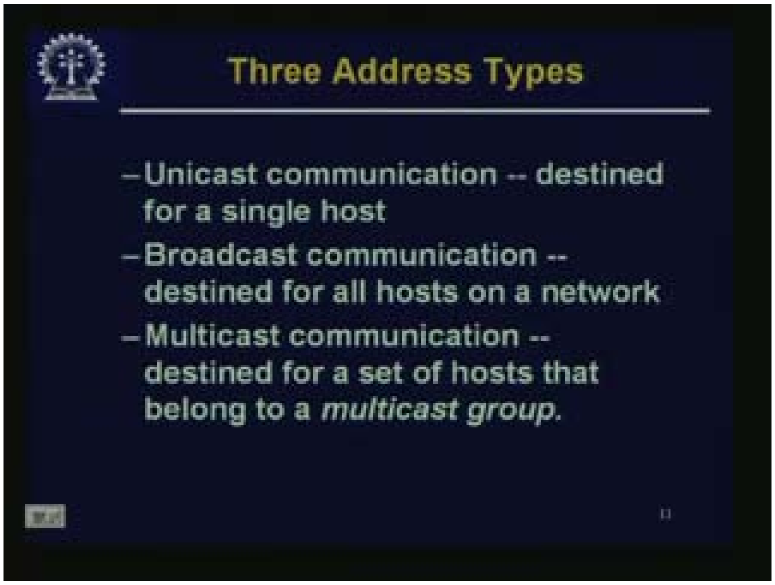
IP Address

- The IP address is a 32 bit address
  - identifies the network and the host on a given network
  - divided into two parts first part identifies the network, second part identifies the host on the network
  - the format is not the same for each address

Now we come to this important topic of IP addresses. The IP address is a 32 bit address that is 4 bytes which identifies the network and the host on a given network. It is divided into two parts, first part identifies the network and the second part identifies the host in the network. The form is not the same if the format is not the same for each address. So this is an important point to understand that when you are given an IP address it is a 32 bit address, these 32 bits are actually divided into 4bytes and each byte is usually read out in decimal separately. You can get an IP address like 144.16.19.23 where this 144 is the decimal equivalent of the binary string which is in the first byte of the address. Then we have 145.16 where 16 is the next byte and next byte contains the decimal binary equivalent of 16. So, if you convert this 144161923 separately into bytes which are the binary strings and put them together that gives you the 32 bit address.

Or in other words, the 32 bit address is usually read out this way. Now in this 4 bytes there is some part which specifies which network you are in and there is some part which specifies which is the host in that particular network. This is the same concept when you think of a postal system where you give the name of the town and then the name of the street or the house. But the people outside will not really know about the streets in a distant town. They will just look at the town name or the pin code and just simply send it there. Then those people will figure out as to where is that particular house or street in that particular town. So we have a network part which is usually important for people who are outside who are trying to route into this network, and then there is a host part which is important within the network. So once the packet has reached this particular network then it has to reach a particular host so this is where the host part comes.

(Refer Slide Time: 13:42 - 14:26)

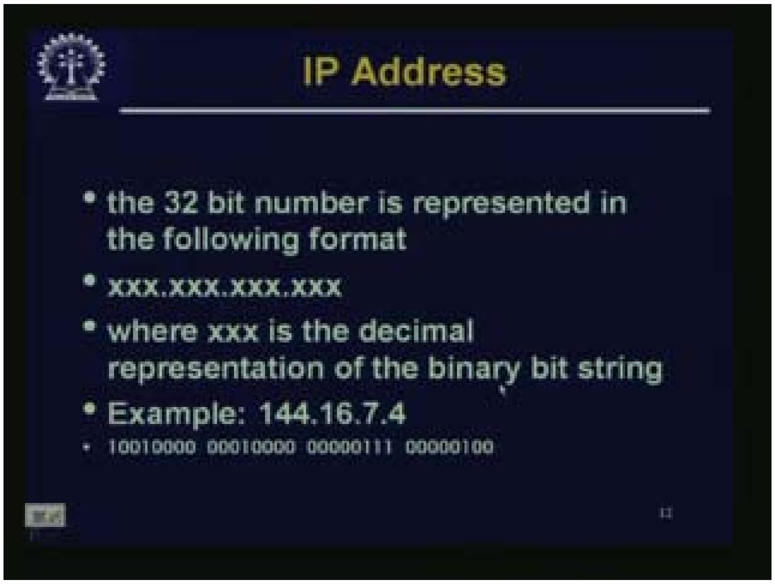


Three Address Types

– Unicast communication -- destined for a single host
– Broadcast communication -- destined for all hosts on a network
– Multicast communication -- destined for a set of hosts that belong to a *multicast group.*

There are three types of addresses in one sense. There are other classifications also. But one is that, it is for unicast communication. Unicast means it is destined for a single host, so, it is originating somewhere and is destined for a particular host. Otherwise there is broadcast communication which is destined for all hosts on a network. In a particular network if you want to send some message to everybody then you can use this broadcast communication. Or there may be multicast communication which is destined for a set of hosts. This is a subset of hosts which are in that particular network which belong to a particular multicast group. So, that is called multicast communication.

(Refer Slide Time: 14:26 - 15:18)

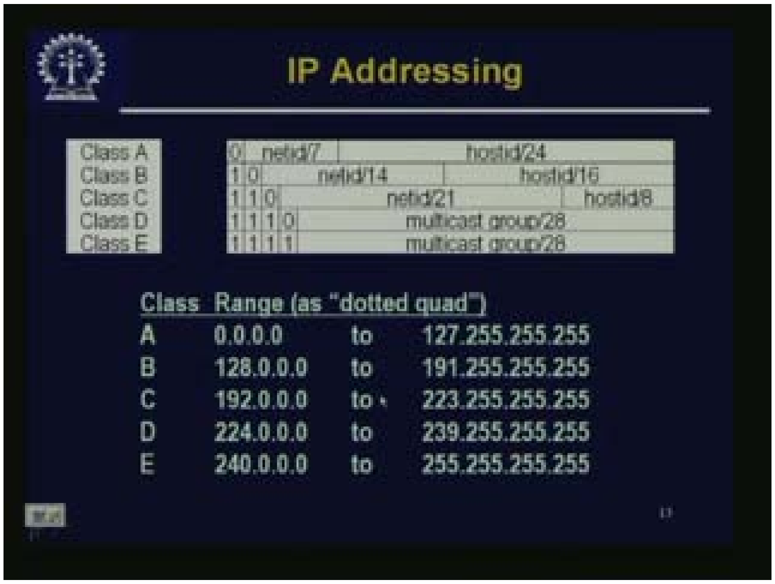


IP Address

- the 32 bit number is represented in the following format
- xxx.xxx.xxx.xxx
- where xxx is the decimal representation of the binary bit string
- Example: 144.16.7.4
- 10010000 00010000 00000111 00000100

As we have already seen, the 32 bit number is represented in the following format. It is actually something dot something dot something dot something where xxx is the decimal representation of the binary bit string. For example, 144.16.7.4 so if you see the first byte it has got a place value of 128 plus 16 which makes it 144. This is the binary equivalent of 144 or 144 is the decimal equivalent of this. So this is the IP address 4 bytes of it and the corresponding decimal equivalent of each of the bytes is given 144.16.7.4. So this looks like a valid IP address.

(Refer Slide Time: 15:19 to 20:51)



IP Addressing

Class A	0	netid/7	hostid/24
Class B	1 0	netid/14	hostid/16
Class C	1 1 0	netid/21	hostid/8
Class D	1 1 1 0	multicast group/28	
Class E	1 1 1 1	multicast group/28	

Class	Range (as "dotted quad")		
A	0.0.0.0	to	127.255.255.255
B	128.0.0.0	to	191.255.255.255
C	192.0.0.0	to	223.255.255.255
D	224.0.0.0	to	239.255.255.255
E	240.0.0.0	to	255.255.255.255

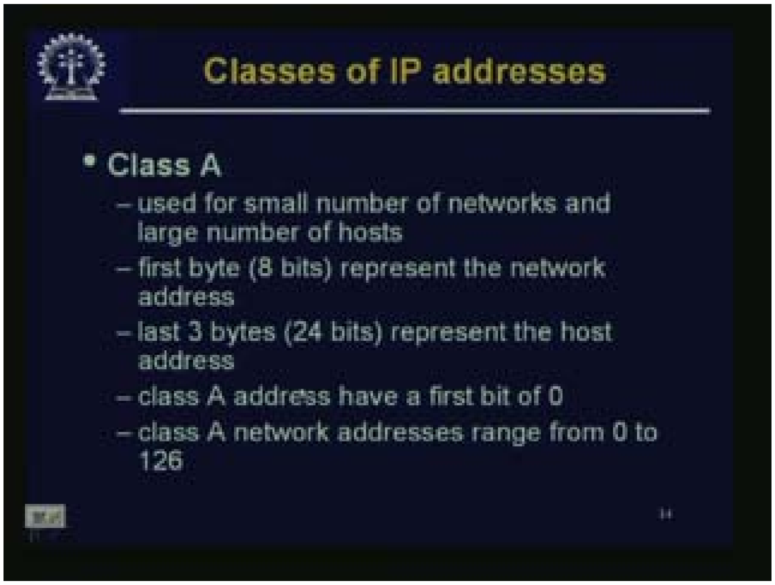
As for another classification of IP addresses, IP version 4 has five classes A B C D E where all class A addresses start with 0 so it has a prefix of 0, class B has a prefix of 10, class C has a prefix of 110, class D has a prefix of 1110 and class E has a prefix of 1111. So you can see that they all have unique prefixes and by looking at it you can make out what it is. Irrespective of what values are here you can make out which class it is. An IP address contains two parts, the first part is the network ID and the rest of it is the host ID.

Now just consider the class A address, the first byte is given to the network ID and of course of the first byte the first bit is 0 indicating that this is a class A address and then you have 7 bits left for network ID. So you can have at most 127 or 126 networks as a class A because we only have 7 bits to represent it. And in each of this network the host ID has 24 bits for specifying the address of the host in that network. So for 24 you have actually 2 power 24 which is about 16 million.

If your organization has a class A address which is very unlikely but if your organization does have a class A address, then this is one big network of that 127 networks world wide which has as many hosts as there in the network and how many hosts can be there? For the host address we have kept 24 bits which means that there can be a possibility of 2 power 24 or about 16 million hosts so there could be a network with 16 million computers in it. Nobody has such a big network and this is very unlikely to happen. This is how a class A address is specified.

Next we go to class B which is in the same way but here instead of 1 byte for specifying the network we have 2 bytes for specifying the network. Out of these 2 bytes the first 2 bits are 10 so that is already gone and the rest 14 bits are for the network ID. So 2 power 14 is about 16,000, you can have 16,000 different class B networks. And each network can contain 2 power 16 hosts which is about 64,000 hosts so these are also fairly big networks. There are networks which are of that order but you have only 2 power 14 or 16,000 such networks but you cannot have more. For class C 3 bytes are given for the network part and only 1 byte. So a class C network can have only 2^8 that is 256 hosts that is a real small network but you can have a large number of them namely you can have about 2 million such networks 2 power 21. Class D is for multicast group and class E is really experimental so they start with 1110. So after 1110 the entire 28 bits is for specifying which group. But this does not work very well so we will come to that later on. So if class A ranges dotted quad so A is 000.0.0.0 of course all zeros is not a valid particular address but anyway we are giving the outer limits of it to 127.255 etc. So, by looking at the first decimal number which corresponds to the first byte we can immediately make out whether this is a class A, class B, class C or class D etc address because class A address cannot be more than 127 and if the first digit or the first decimal number is less than 127 then you know that this must be a class A address. If it is from 128 to 191 then you know that this is a class B address and so on. So, by looking at the first decimal number which is the equivalent for the first the first byte is being dictated this way depending on the class of the address. That is how they come to a particular range of numbers. So, by looking at the first number we can know which class of address it is.

(Refer Slide Time: 20:52 - 21:21)



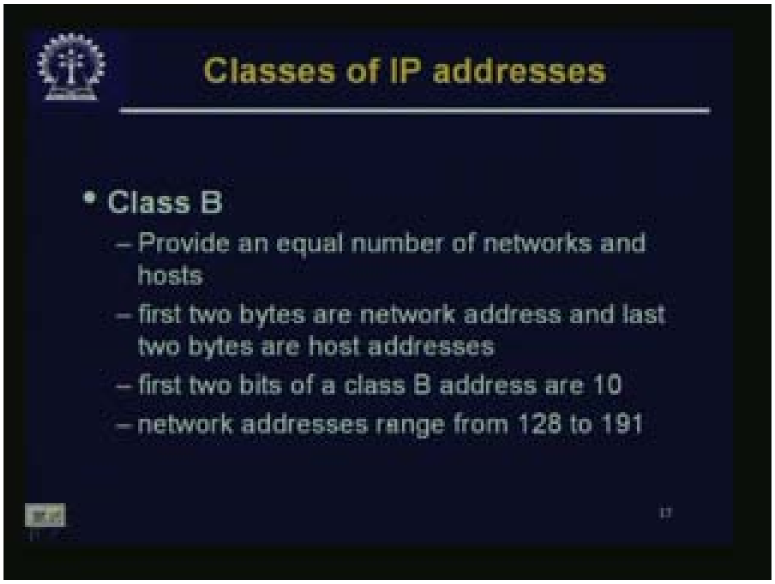
Classes of IP addresses

- Class A
  - used for small number of networks and large number of hosts
  - first byte (8 bits) represent the network address
  - last 3 bytes (24 bits) represent the host address
  - class A address have a first bit of 0
  - class A network addresses range from 0 to 126

14

So this class A is obviously used for very small number of networks and large number of hosts. First byte represents the network address and the last three bytes represent the host address. Class A address have a first bit of 0 and class A network addresses range from 0 to 126 and 127 is actually reserved for something else.

(Refer Slide Time: 21:21 - 21:49)

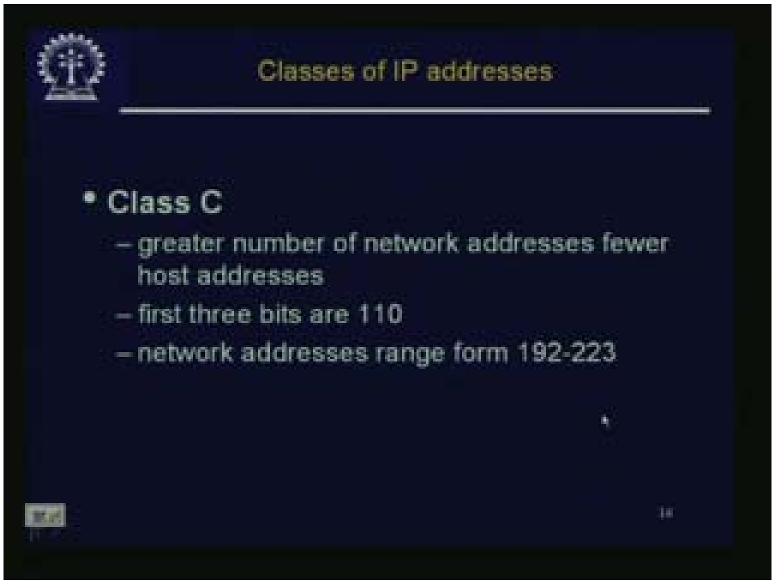


Classes of IP addresses

- Class B
  - Provide an equal number of networks and hosts
  - first two bytes are network address and last two bytes are host addresses
  - first two bits of a class B address are 10
  - network addresses range from 128 to 191

Class B provides an equal number of networks and hosts. First 2 bytes are network address and last two bytes are host addresses. First 2 bits of a class B addresses are 1 and 0 so you can only have 16,000 such class B networks and network addresses range from 128 to 191 that is the first decimal number for the first byte.

(Refer Slide Time: 21:29 - 22:03)

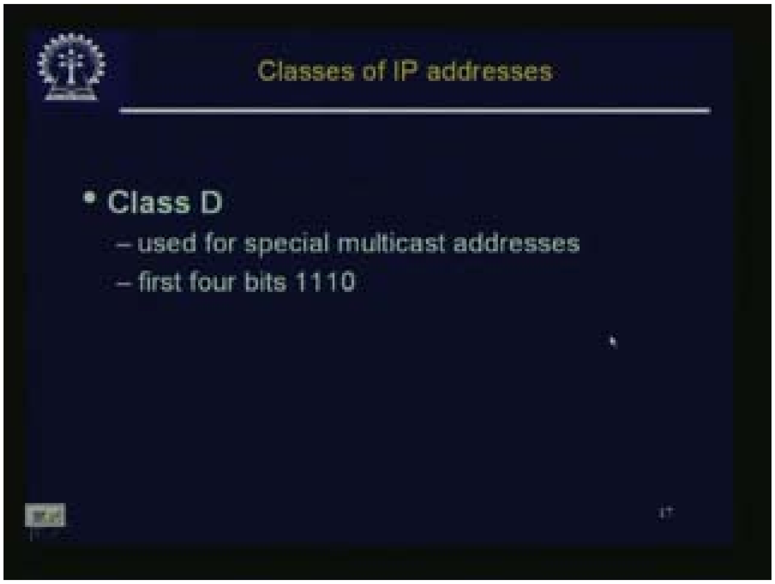


Classes of IP addresses

- Class C
  - greater number of network addresses fewer host addresses
  - first three bits are 110
  - network addresses range form 192-223

Class C is a greater number of network addresses, fewer host addresses, the first 3 bits are 110. So network addresses range from 192 - 223.

(Refer Slide Time: 22:04 - 22:11)



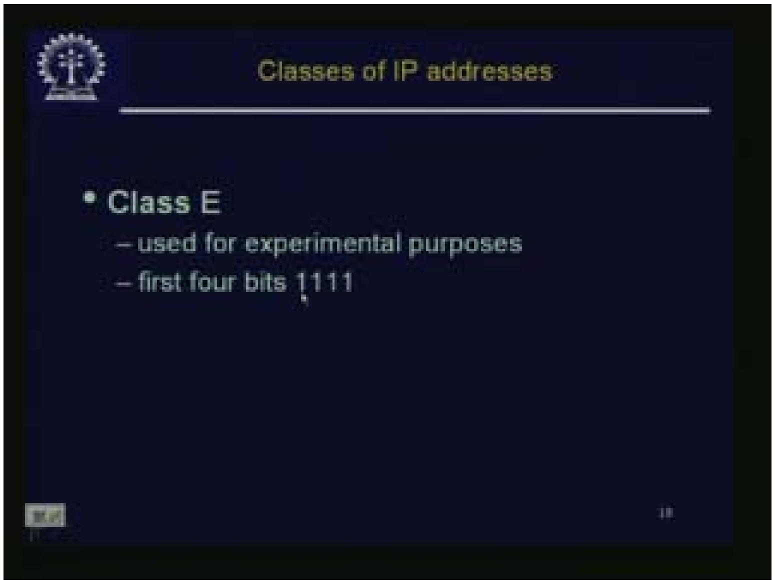
Classes of IP addresses

- Class D
  - used for special multicast addresses
  - first four bits 1110

17

Class D is used for special multicast addresses. The first 4 bits of Class D are 1110.

(Refer Slide Time: 22:12 - 22:21)



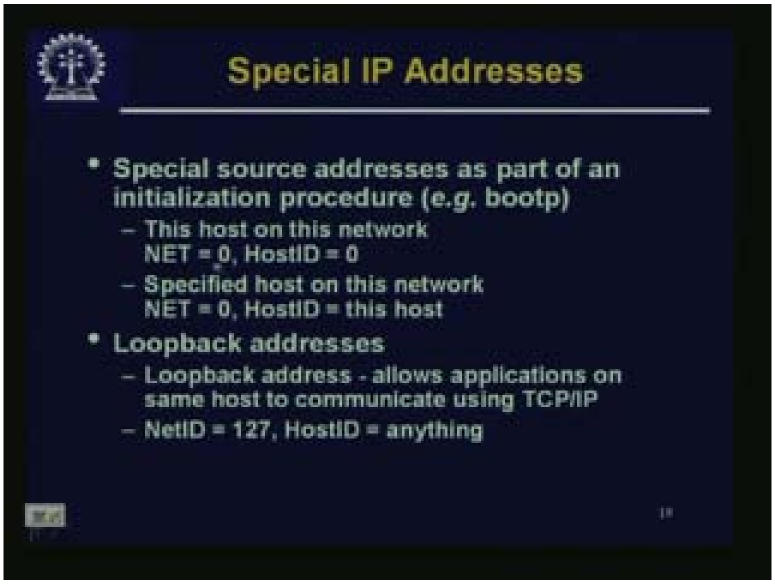
Classes of IP addresses

- Class E
  - used for experimental purposes
  - first four bits 1111

18

Class E is used for experimental purposes. The first 4 bits of Class E are 1111.

(Refer Slide Time: 22:21 - 25:42)



Special IP Addresses

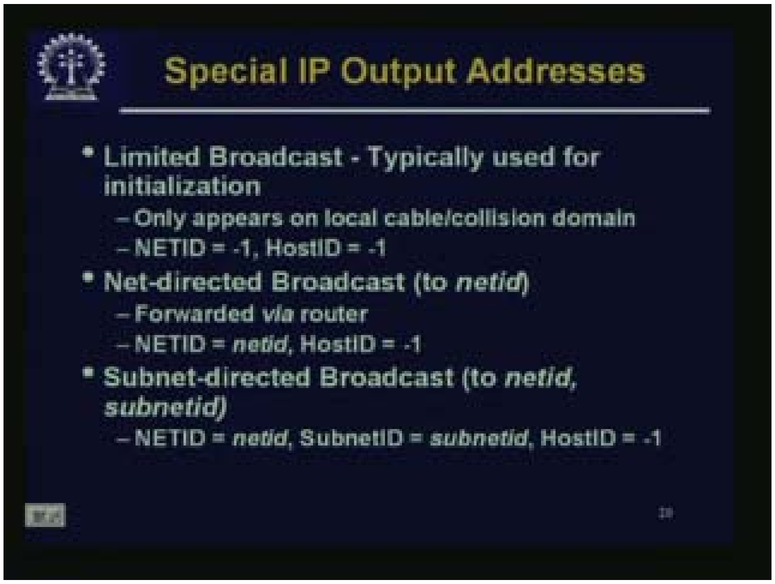
- Special source addresses as part of an initialization procedure (*e.g.* bootp)
  - This host on this network
    NET = 0, HostID = 0
  - Specified host on this network
    NET = 0, HostID = this host
- Loopback addresses
  - Loopback address - allows applications on same host to communicate using TCP/IP
  - NetID = 127, HostID = anything

There are some special source addresses as part of an initialization procedure for example bootp. This host on this network, the net part is 0 and host ID part is 0. That means when the net part is 0 depending on which class of address it is and suppose it is the class C address then the first 3 bytes are really the network part so it will be 0.0.0 and the host part is 0. So if you say 0000.0.0.0 it means that it is itself. So this is sometimes required for example in bootp protocol you require referring to yourself. Specified host on this particular network: So you keep the network part 0 and the host ID for this particular host that you want to specify, suppose this is a class C address, the host part is given only by the last byte.

Suppose the last byte has a decimal value of whatever maybe 130 or something. So if you say 0.0.0.130 that means whatever the network I may be in, in this particular network get me the host number 130. It is a specified host on this particular network wherever it is. Loop back address: The loop back address allows applications on the same host to communicate using TCP IP. Here the net ID is given as 127 and host ID could be anything which means the first byte is all 1 that is the loop back which is referring to this particular host.

Why do you require this? Suppose two different applications are running on the same host and they want to communicate with each other using TCP IP. But why do they require TCP IP if they are on the same host and when they could communicate directly? The point is that these two applications could have been hosted in two different hosts also. If this was the case then they would have to use TCP IP. So instead of writing two different versions, one for the case where both of them are in the same host and the other in two different hosts you write the same program and use the same TCP IP stack. The only thing is that you want to refer to the same host. If you are trying to refer to the same host using the same TCP IP how you would know the network address and so you would not be interested to include a hardware into that particular application. Therefore there is a way to refer to the same host which is by using 127 that is all 1s in the first byte.

(Refer Slide Time: 25:43 - 28:41)



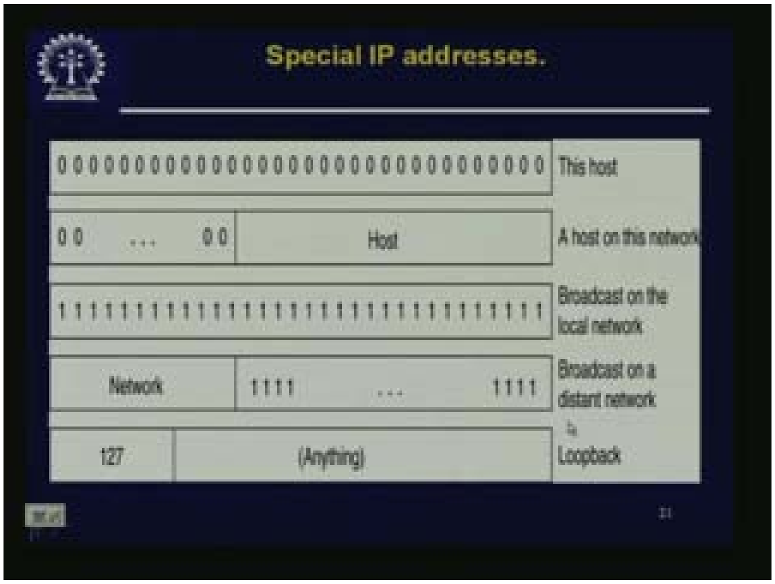
Special IP Output Addresses

- Limited Broadcast - Typically used for initialization
  - Only appears on local cable/collision domain
  - NETID = -1, HostID = -1
- Net-directed Broadcast (to *netid*)
  - Forwarded *via* router
  - NETID = *netid*, HostID = -1
- Subnet-directed Broadcast (to *netid, subnetid)*
  - NETID = *netid*, SubnetID = *subnetid*, HostID = -1

Then we were talking about three different types of addresses for unicast, broadcast and multicast. We have seen about unicast communication that is the network part and the host part, both you indicate and then you are talking to that particular host. We have seen multicast in class C addresses and mostly in class D so that was a multicast group. And then now you talk about broadcast. But here there is a caveat. In the sense that you are not allowed to broadcast to the whole wide world because if everybody or even a very limited fraction of people start broadcasting some message to everybody in the world then the entire network will be swarmed with broadcast because now in this age of internet millions and may billions of people are getting connected to the internet. So, if only a small fraction of them want to broadcast things to everybody that cannot be allowed because then the entire network will go down. So broadcasts are always limited and how they are limited and how the broadcast addresses are specified will be discussed now.

Limited broadcast typically used for initialization only appears on local cable or collision domain with net ID is – 1 that is all 1s and host ID is all 1s. This means that if you give an address which is all 1s it means that you want something to be broadcast in the local network wherever you are in. The net directed broadcast means you want to broadcast to a particular network. So this is forwarded via router. Now that particular network has to be mentioned so the network address part is to be specified giving the net ID, the host ID is all 1s so all 1s means to every body. But if you put all 1s in network part as well as all 1s in the host part it cannot mean that all networks and all hosts in all networks, that is not allowed. All 1s in the network part as well as all 1s in the host part means that it is for all the hosts in this particular network. And then there could be subnet directed broadcast. We have not talked about subnet as yet which we will do. There will be a subnet ID which is really carved out of the host part of the address and then host ID part will be all 1s.

(Refer Slide Time: 28:41 - 29:14)

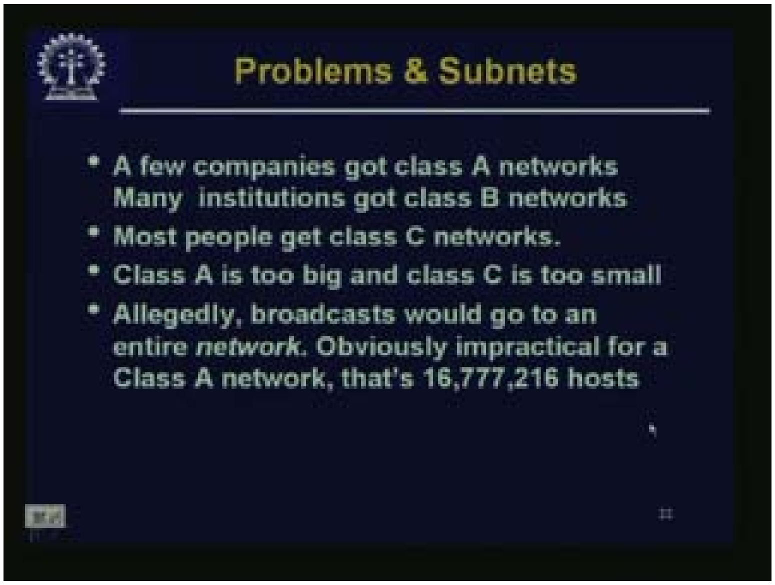


Special IP addresses.

00000000000000000000000000000000		This host
00 ... 00	Host	A host on this network
11111111111111111111111111111111		Broadcast on the local network
Network	1111 ... 1111	Broadcast on a distant network
127	(Anything)	Loopback

So all zeros means this host and all zeros in the network part and the host part is something specific which is like a particular host on this particular network. All 1s mean broadcast on the local networks. If the networks part is specified and if host part is all 1s it means that broadcast is on a distant network. If it is 127 then it is all 1s on the first byte and anything in the rest of it is a loop back.

(Refer Slide Time: 29:15 - 34:10)



Problems & Subnets

- A few companies got class A networks Many institutions got class B networks
- Most people get class C networks.
- Class A is too big and class C is too small
- Allegedly, broadcasts would go to an entire *network*. Obviously impractical for a Class A network, that's 16,777,216 hosts

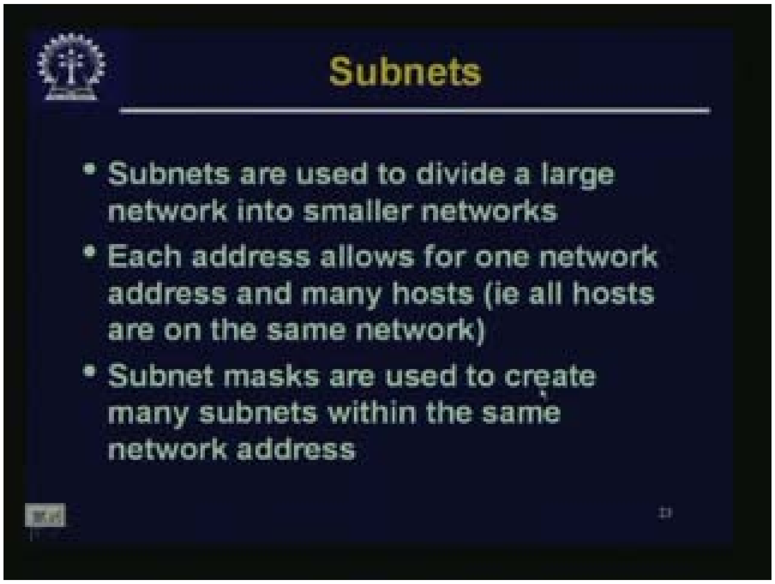
Now we come to a problem. A few companies got class A like Xerox and some other companies got a class A address when they were actually very closely involved with designing internet in the Arpanet days so people really did not know who designed this network and had no idea that their project is finally going to blossom so wildly and become greatly successful. They had no idea that they had to be careful with all these addresses that they were doling out so the companies which came in got some class A addresses. These class A addresses can have 16 million hosts which is really much bigger than anything that a company might want. Many institutions got class B networks such as 12 institutions. For 14 bits you can have only 16,000 so it is not such a big number when you are talking about in the global scale so 16,000 institutions and may be 126 companies over there is really a small number. But now-a-days there are millions of companies and ripped hundreds of thousands of companies who want to have their own network etc. They are now reduced to accepting only class C address. Class C address is too big because you cannot have 16 million hosts but a class C address can accommodate only 256 hosts but 256 is a very small number.

Any institution or many institutions now-a-days have got thousands of computers in their network so this too small for them. For the first one let us think of another problem. Suppose you have a class B network where you have 10,000 nodes now if all these 10,000 nodes is one network, the network in our parlance at this particular moment is one particular broadcast domain, so you can always broadcast in this network. If you are sitting in one particular network and you communicate to some other node in that particular network then you need to know his MAC address which you do not know but you know his IP address so what you will do is you will broadcast the IP address asking for the MAC address. Now whichever machine has got that particular IP address you will get it and answer with his MAC address and that is how the ARP protocol works. So whenever you try to communicate if you do not know the MAC address of the other side naturally you will send a broadcast to the entire network.

Now, if all these 10,000 hosts start sending broadcast messages from time to time then the broadcast traffic would be too much. Since the network is so big we have to break it up into smaller parts so that broadcasts are limited to smaller sub networks for which you need some more bits.

Previously we were talking about the two parts of the address, one is saying this network and then we are saying this host in this network. Now we have to say three things; it is this network, this particular sub network in that network and then this particular host in that sub network. It is just like instead of a town if you have a city then in a city there will be a large number of post offices in the same city. From outside may be from another country they will send it to that particular city and in the city you will decide this is that particular post office in that city and that particular post office would know that it is for this particular house on this street in this region. So we have a network, a sub network which is the breaking of a big network into smaller sub networks and then we have a host in that particular sub network. So, as we have said that broadcast would enter a network obviously it is impractical for class A networks and even for class B networks.

(Refer Slide Time: 34:10 - 34:29)



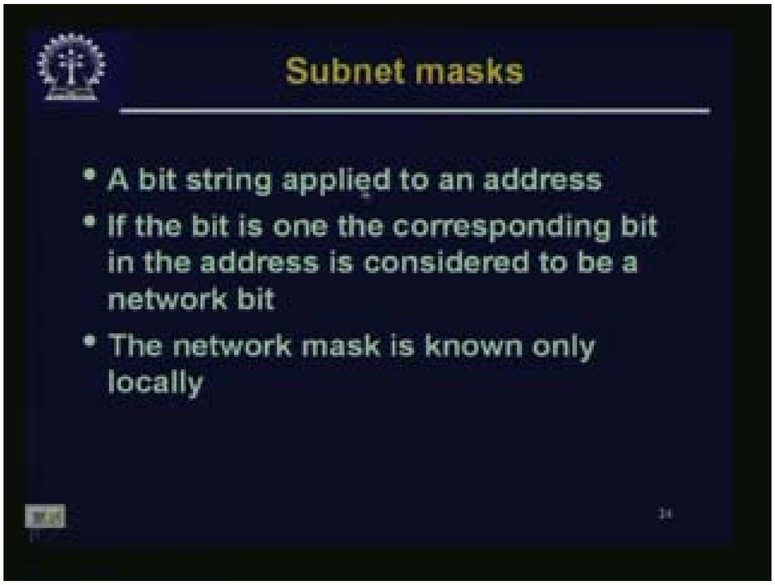
Subnets

- Subnets are used to divide a large network into smaller networks
- Each address allows for one network address and many hosts (ie all hosts are on the same network)
- Subnet masks are used to create many subnets within the same network address

23

So subnets are used to divide a large network into smaller networks. Each address allows for one network address and many hosts that is, all hosts are on the same network. Subnet masks are used to create many subnets within the same network address.

(Refer Slide Time: 34:30 - 35:32)



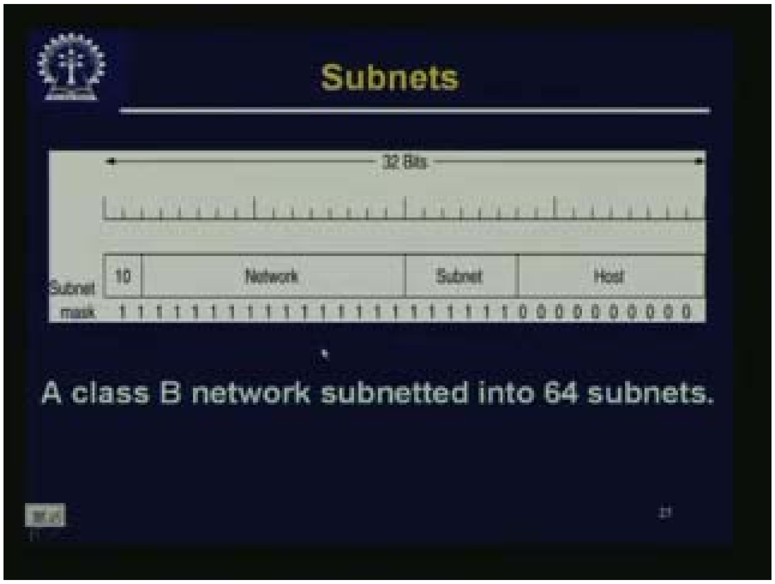
Subnet masks

- A bit string applied to an address
- If the bit is one the corresponding bit in the address is considered to be a network bit
- The network mask is known only locally

24

So we will look at subnet masks. This is a bit string applied to an address. If the bit is 1 the corresponding bit in the address is considered to be a network bit. The network mask is known only locally. If we have one part which is the network part and the next part which is the host part then we take some bits from the host part and use them for specifying the sub network. The number of bits of the network which you take for the sub network is given by the subnet mask by placing those particular bits to be 1 and this subnet mask is known only locally.

(Refer Slide Time: 35:32 - 36:55)

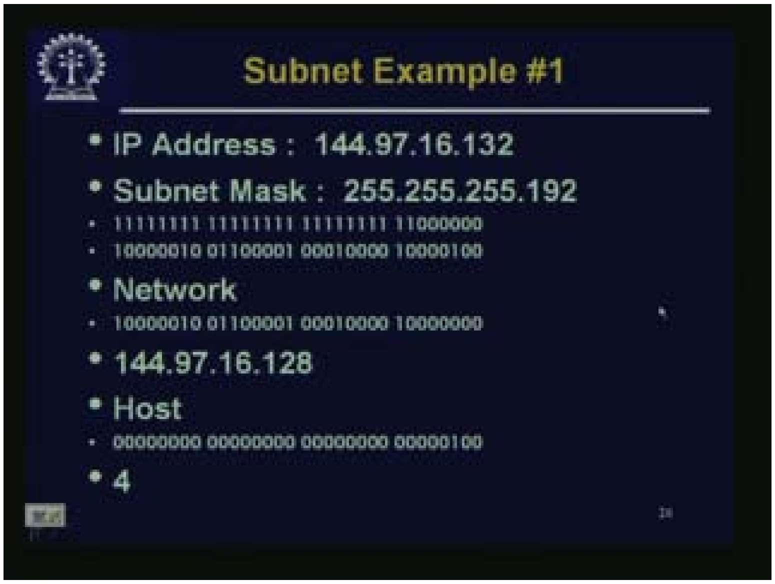


Subnets

A class B network subnetted into 64 subnets.

So this is an example. Suppose we have a class B network, in a class B network you know that the first two bytes is the network part and the other two bytes is the host part. Now in this host part these 6 bits will show my sub network address. So you make the corresponding bits in the network mask to be 1 and the rest are all zeros. The host part are all zeros, Now, looking at the address we will know whether it is a class A, B, C just by looking at the first number and we know how much is the network part. If we know the subnet mask we know how much is the network and sub network part. If we take out the network part from that we get the subnet address here. Actually this not the address, the masks only tells you that these are the bits which are used for the subnet address and the rest of it is for the host address. We will see examples of this.

(Refer Slide Time: 36:55 - 39:07)



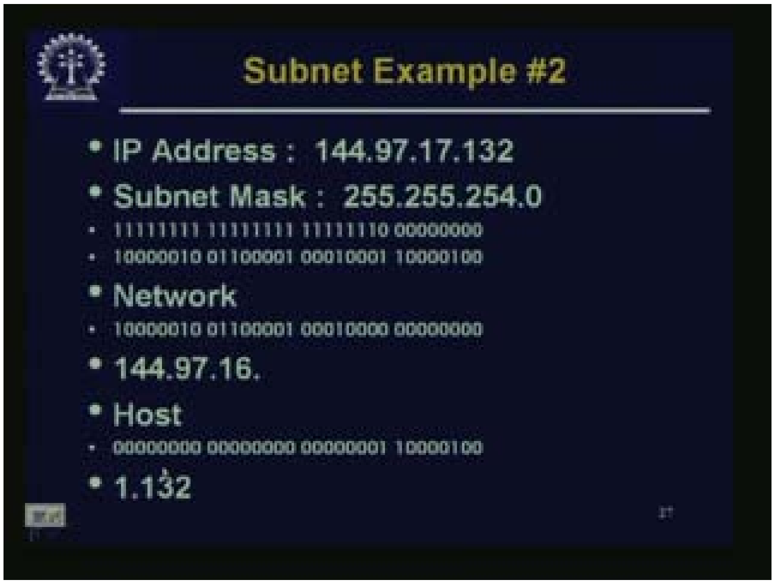
Subnet Example #1

- IP Address : 144.97.16.132
- Subnet Mask : 255.255.255.192
  - 11111111 11111111 11111111 11000000
  - 10000010 01100001 00010000 10000100
- Network
  - 10000010 01100001 00010000 10000000
- 144.97.16.128
- Host
  - 00000000 00000000 00000000 00000100
- 4

Let us say we have an IP address 144.97.16.132 that is the IP address of a particular host. And we are also told that we have a subnet mask of 255.255.255.192,1 etc. These are all for only human communication but of course these are all individual bytes which are to be converted to the corresponding bit strings. The 255 are all 1s in the byte so all 1s and 192 is 128 plus 64 which means the first two bits are 1 and the rest is 0 so this is subnet mask.

Since this is the class B address we know that if you convert 144.97.16.132 then this is the string you get 144. So the first two bytes is for the network part. Now we also know that the 8 plus 10 bits are for the sub network part. So this is the sub network address 0001000010 and the host is 100 part. So the network part is the first two and then the sub network part 000100 so this is the entire network part wherever we have a 1. And beyond that the point where we have a 1 we put all zeros for the network address. So network address is 144.97.16.128, the 144.97 is telling me which particular network it is and 16.128 is telling which particular sub network is within that network. And the host is this 100 so the host is 4. Let us just see another example.

(Refer Slide Time: 39:08 - 39:57)



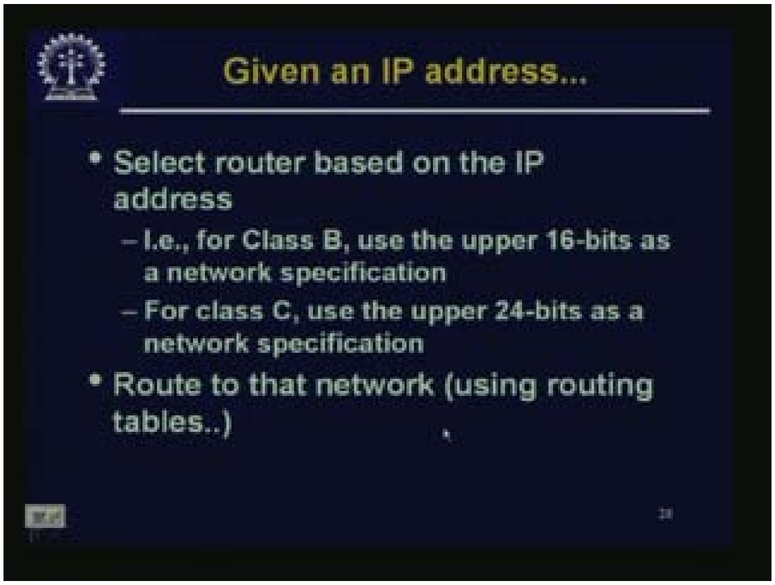
Subnet Example #2

- IP Address : 144.97.17.132
- Subnet Mask : 255.255.254.0
- 11111111 11111111 11111110 00000000
- 10000010 01100001 00010001 10000100
- Network
- 10000010 01100001 00010000 00000000
- 144.97.16.
- Host
- 00000000 00000000 00000001 10000100
- 1.132

27

So we have this IP address which is 144.97.17.132 and the subnet mask is this. So we see that the first seven bits so the network has been broken down into 127 different networks and then we have 9 bits for hosts so we can have may be 512 hosts in each network. This is the network part up to this one so we take all the 1s there and the rest are put as 0. So network is 144.97.16 and host is 1.132.

(Refer Slide Time: 39:58 - 40:53)



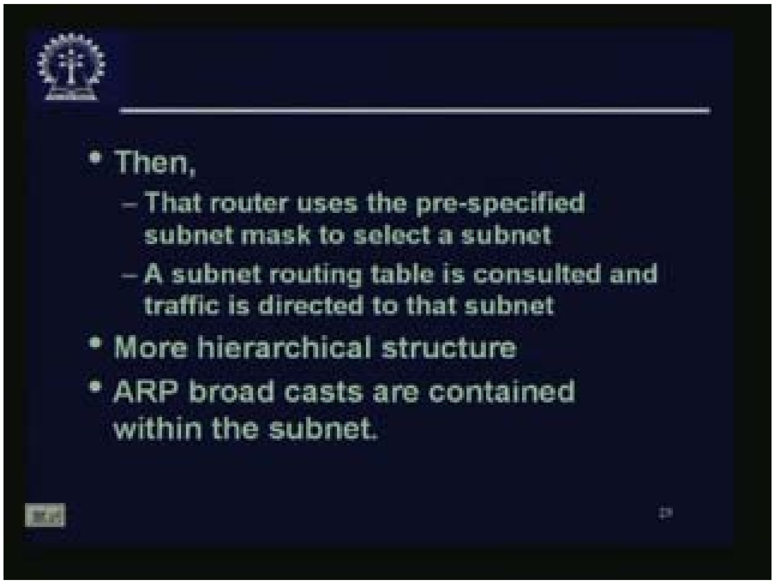
Given an IP address...

- Select router based on the IP address
  - I.e., for Class B, use the upper 16-bits as a network specification
  - For class C, use the upper 24-bits as a network specification
- Route to that network (using routing tables..)

28

So how do packets get to the other end? Select router based on the IP address. That is, for class B use the upper 16 bits as a network specification, for class C use the upper 24 bits as network specification and so on and naturally for class A just look at the first 8 bits. Route to that network using the routing tables as we have seen. So depending on whether it is a class A, class B or class C you use 1 byte, 2 bytes or 3 bytes as the network address. Route to that network using routing tables. If your routers, RIP or OSPF etc is working properly then it will reach that particular network. The point is what happens after that?

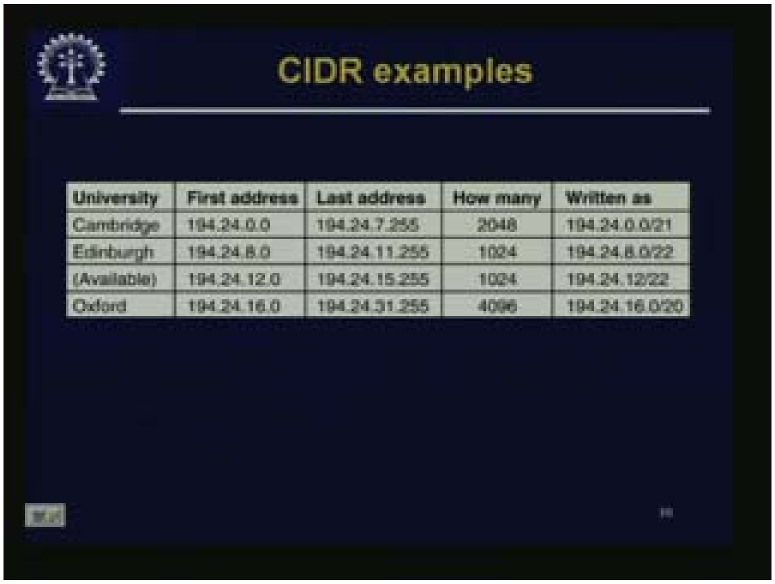
(Refer Slide Time: 40:53 - 41:36)



- Then,
  - That router uses the pre-specified subnet mask to select a subnet
  - A subnet routing table is consulted and traffic is directed to that subnet
- More hierarchical structure
- ARP broad casts are contained within the subnet.

Then the router uses the pre-specified subnet mask to select a subnet because looking at the subnet mask and at the IP address it knows how many bits are there for specifying the subnet. So it finds the subnet mask and it just takes out the subnet part of the address which is looked up in a subnet routing table. A subnet routing table is consulted and traffic is directed to that particular subnet. So this gives you a more hierarchical structure and ARP broadcasts are contained within the subnet. If you reach that particular subnet then the host number is given and you can go to that particular host.

(Refer Slide Time: 41:37 - 46:14)



CIDR examples

University	First address	Last address	How many	Written as
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

Now, we come to the other part of the problem. As I mentioned class A is of course huge, class B is also very large so we have to break them up into the subnet. The other side of the problem is that, the class C address is very small. The other thing to note is that the network has grown so much. There is a tremendous demand for IP addresses. This IP addresses have to a global standard. You cannot have your local standard because other people would be looking at your address may be at a different corner of the globe and try to route packets to you. So these addresses have to follow a global knob and that was used to be controlled by one central body.

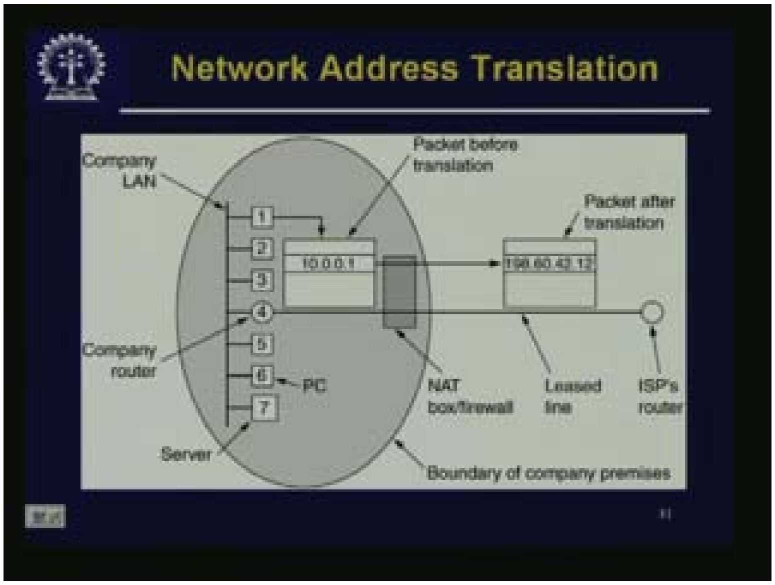
Now, if you have to give the address to somebody then you have to give internetwork address for his particular network usually. And now this creates a problem in the sense that when the demand for these addresses becomes too high you run out of addresses. As a matter of fact we have come to a stage today where for all practical purposes we have run out of addresses. Now you only have a few class C addresses left.

If we had been much more careful, if we could have envisaged how the networks would grow and if we had been much more careful earlier in assigning addresses and not waste big addresses like that may be we could have stretched this for by a few more years. But anyway this cannot be helped now. So people are working on various types of solutions to overcome the shortage of IP addresses.

And the other side was that quite a number of years back people worked on a new protocol, you must have noticed that this lecture is titled IP version 4 possibly implying that there are other versions available and actually there is IP version 6 which was finalized quite a few years back. Anyway in the IP version 4 we have this problem, what are the various kinds of workarounds? One thing could be, if you take a big address chunk let us say class B address and give it to different organizations may be some parts of it, the one problem is that, first of all you are breaking down these classes class A, B, C at the byte boundary. So the point was that do not have them at the byte boundary so they no longer belong to one particular class, they are called classless. And classless interdomain routing that is the CIDR is the protocol which is there where you specify your starting address and then specify how many hosts you have in your particular network, so this is an example.

Suppose some British Universities like Cambridge, suppose the first address is this and the last address is this that means how many nodes they can have? You can have from 0 to 7 so that is 8 over here and of course 256 on the last byte so 8 into 256 that is about 2048 hosts. So it gives the starting address 194240.0/21. This 21 is a code which really shows that there are 2048 hosts. Now various such numbers are possible, 22 means 1024 hosts, 20 means 4096 hosts. This way it goes down on one side and on the other side you can even have less than 1024 hosts so various numbers are there and there is a table here. Similarly 19 would mean 8000 hosts and so on.

(Refer Slide Time: 46:15 - 50:57)



Network Address Translation

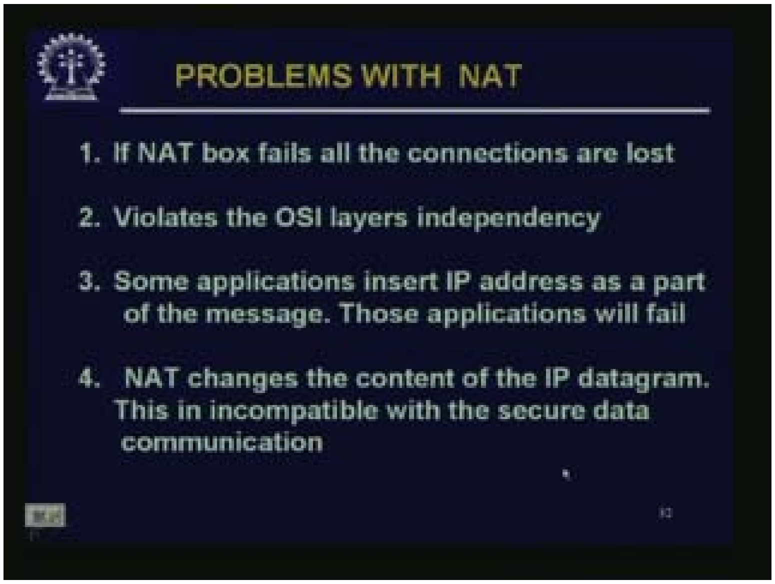
There is another workaround people have done and actually many organizations are now doing it and in common parlance it is called NATing that is Network Address Translation. As I said earlier your IP address has to be known globally, it has to be a global standard, globally assigned. Now, if we make an observation that this is true only when we are communicating with somebody or only when somebody else is trying to communicate with me that is when it needs to be global. So you have an entirely private address and if it is a private why not it be a class A address? So whatever be the size of your organization may be you use a class A address inside. This class A address that you have just to use it without permission means that this address is not recognized globally.

Assuming that you are in a big organization you do all your internal communication using this private address. The point is that only when you are going out to communicate with somebody you will mask your private address, keep a temporary table and tat table will be dynamic one, for the time being you want to communicate, dynamically there will be your local address which is actually a private address and not a globally legal private address and you will put it and then you will have a pool of legal addresses. And you will use one of them whichever is free and then start communicating. To the outside world it will be as if you are communicating with this particular legal address which you have assumed for a temporary point of time while you are communicating. That is called Network Address Translation.

Suppose we have this company LAN that is the company router etc and then a packet arrives and suppose this has an address 10.0.0.1 then the first number is 10 which immediately tell you that this is a class A address but really this company does not have A class A address but is using it. And there is a convention that when we use private address some how we use 10. But whenever somebody looks at an address that starts with 10 he knows that this is a private IP address. Therefore this is a private address he is using and this is going through may be a NAT box or firewall.

Now, this NATing could be done at a firewall. The NATing can be done in the router also and so all these boxes usually come with that capability. So, in the NAT box over here or this firewall maintains a table of this IP address it is trying to communicate, it assigns the pool of IP addresses and out of those IP addresses may be this particular IP address 198604212 is free at the moment. So he will take out this particular address, put in this particular address and send it to the outside world. So the outside world will know the source to be this particular address. When the outside world replies back it will come back here to this particular address, it will go through the same box, the box will know this is really an address which is temporarily assigned to him so he will now take this out, put 10.0.0.1 and send the packet to the particular host in that particular network. So the outside world will know that he is communicating with this fellow whereas this is not really a fellow this is just one of a pool of address which is shared by a large number of hosts and the translation is done here. So that is an example of how NATing is done.

(Refer Slide Time: 50:57 - 52:13)

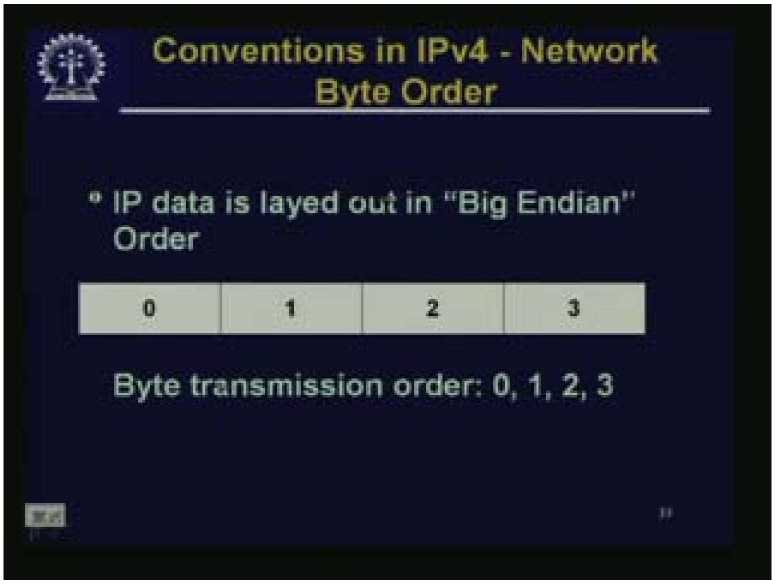


PROBLEMS WITH NAT

1. If NAT box fails all the connections are lost
2. Violates the OSI layers independency
3. Some applications insert IP address as a part of the message. Those applications will fail
4. NAT changes the content of the IP datagram. This in incompatible with the secure data communication

This is not entirely satisfactory but this is used quite often for example in my organization IIT we have got more than 10,000 machines and we do not have a class B address. You cannot help, so we just have a bunch of class C address. So we assign these class C addresses to all these boxes, we do the NATing and that is how we communicate. One problem is that if the NAT box fails all the connections are lost. It violates the OSI layer independency because this is a workaround and we do not have so many addresses that people demand. Some applications insert IP addresses as a part of the message then of course that application will fail because if the IP address is some how hard coated inside the application message somewhere that is not going to work. And NAT changes the content of the IP datagram, this is incompatible with secured data communication. If you want to do the entire thing secured including your IP addresses where you are communicating this NAT will not work or wherever you are doing NATing you can not encrypt that part.

(Refer Slide Time: 52:14 - 52:30)



Conventions in IPv4 - Network Byte Order

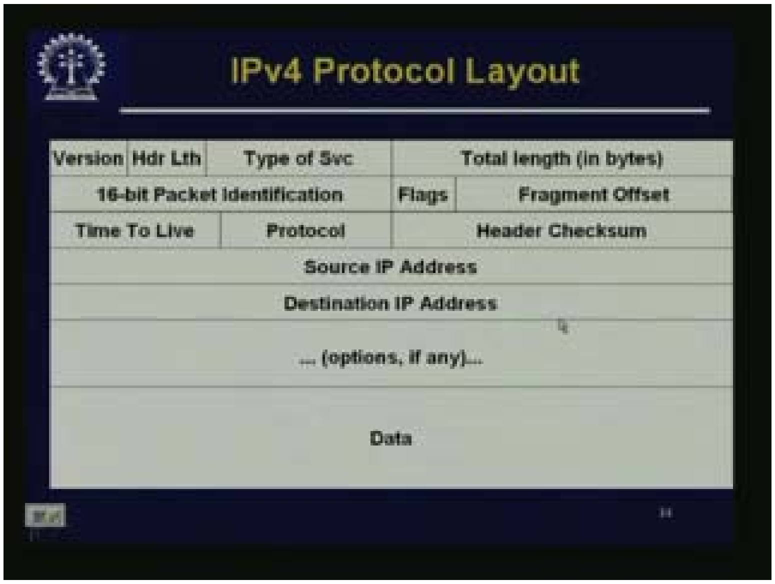
- IP data is layed out in "Big Endian" Order

0	1	2	3

Byte transmission order: 0, 1, 2, 3

IP data is laid out in big Endian order. That means byte transmission order is 0, 1, 2, 3. You know big Endian or little Endian or which way you go 0, 1, 2, 3 1 or 3, 2, 1, 0. So, in networks this is the network byte transition.

(Refer Slide Time: 52:31 - 53:23)

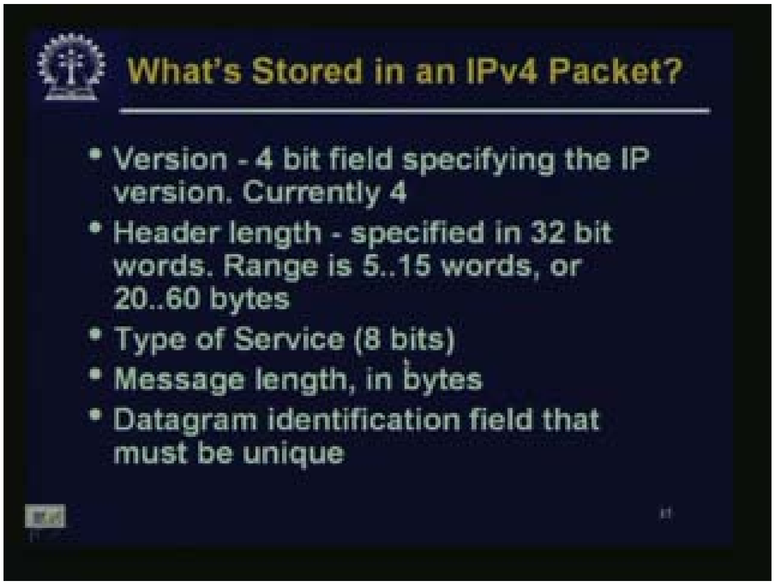


IPv4 Protocol Layout

Version	Hdr Lth	Type of Svc		Total length (in bytes)
16-bit Packet Identification			Flags	Fragment Offset
Time To Live		Protocol	Header Checksum	
Source IP Address				
Destination IP Address				
... (options, if any)...				
Data				

In this IP header we have version, header length etc and the IP header is 20 bytes or more, it is minimum 20 bytes so this is 4 bytes each so that is 32 bits each. The source IP address is given as 4 bytes, the destination IP address is given so that is another 4 bytes so that is 8 bytes gone. These three batches of 4 bytes each which amounts to 12 bytes are used for various things. Later on there may be some options.

(Refer Slide Time: 53:23 - 54:36)

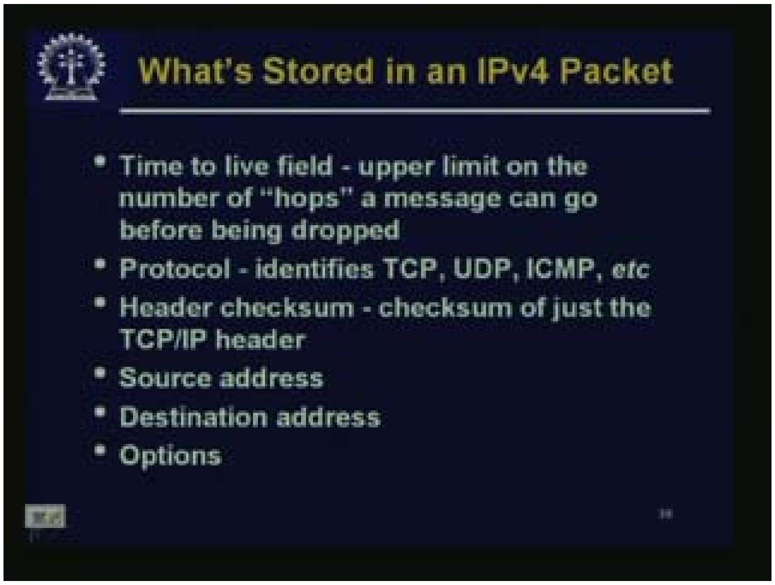


What's Stored in an IPv4 Packet?

- Version - 4 bit field specifying the IP version. Currently 4
- Header length - specified in 32 bit words. Range is 5..15 words, or 20..60 bytes
- Type of Service (8 bits)
- Message length, in bytes
- Datagram identification field that must be unique

This is version 4 then version 6 would be there, bit filled specifying the IP version currently 4. Header length specified in 32 bit words and range is from 5 to 15 words or 20 to 60 bytes. So what is the length of the header? Why do we need the length of the header? If you look at the previous one there could be options, so how do you know whether just after destination IP address the data starts or the header goes some more and more options are exercised? The header length has to be given. The type of service: Some kind of quality in service was expected but this did not work out very well and is mostly ignored now. Then the message length is in bytes. The datagram identification field must be unique. So there is a datagram identification field, 16 bit packet identification. We will talk about it when we talk about Fragmentation.

(Refer Slide Time: 54:37 - 56:49)



What's Stored in an IPv4 Packet

- Time to live field - upper limit on the number of "hops" a message can go before being dropped
- Protocol - identifies TCP, UDP, ICMP, *etc*
- Header checksum - checksum of just the TCP/IP header
- Source address
- Destination address
- Options

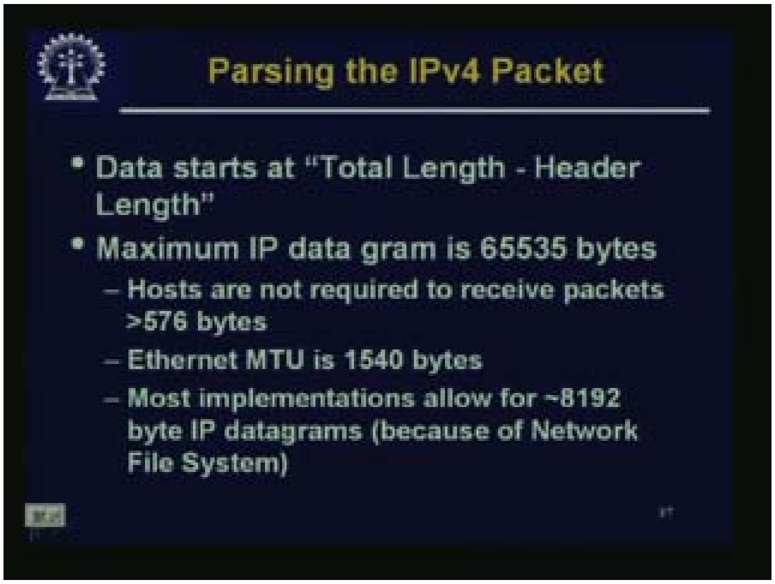
Time to live field: Upper limit on the number of hops that a message can go before being dropped. Although this is called Time to live, actually this is given by the number of hops and why do you require that? Sometimes the routers work in a distributed fashion as we have seen in this RIP and other protocols that they work in a distributed fashion and there may be some problem somewhere. Now because of that problem you may get a routing loop. That means virtually since this loop is stored in a distributed fashion nobody really detects that there is a loop but actually one particular packet finds that this has gone in a loop.

Now this packet if it does not die out naturally it will keep on circulating at infinitum and such packets will get accumulated and it will bring down the whole network. So there is a mechanism and there are other reasons for this of course. But there is a mechanism that if a packet has gone very much astray or if it is just circulating after some number of hops the router will see that it has already crossed so many hops and this is the time to live and the time to live has come down to 0 so drop it. Otherwise you just reduce the time to live by 1 and send it to the next hop.

Now there is a protocol which identifies TCP, UDP or ICMP. You remember that this is on the network layer. Now above the network layer there is a transport layer. IP is a hourglass design which sort of concentrates on the IP from the various different types of networks like Ethernet, token ring etc. It also comes to IP, from IP it goes to various different protocols like TCP, UDP and so on.

Now in the network layer how does it decide where to go to, whether to send it to TCP in transport layer or whether to send it to UDP in the transport layer, so that must also be mentioned. So this protocol identifier is there whether it is TCP, UDP, ICMP, IGMP etc. Header checksum: Checksum of just the TCP, IP header that is this IP header and source address which is 4 bytes, 32 bits destination address another IP address this is again 32 bits and options.

(Refer Slide Time: 56:48 - 57:45)



Parsing the IPv4 Packet

- Data starts at "Total Length - Header Length"
- Maximum IP data gram is 65535 bytes
  - Hosts are not required to receive packets >576 bytes
  - Ethernet MTU is 1540 bytes
  - Most implementations allow for ~8192 byte IP datagrams (because of Network File System)

Data starts at total length that is the header length etc and maximum IP datagram size is 64 kilo bytes. Hosts are not required to receive packets greater than 576 bytes. That means at least 576 bytes they have to accepted. Ethernet, MTU is only 1540 bytes. So most implementation allowed is about 8000 bytes for IP datagrams. The point is that when a particular packet has come to a network the packet may be too large for that particular network to handle, so then something has to be done. One thing is of course to drop it but if you drop it then every time he wants to send the packet the packet will get dropped. So what is done Is, this packet is broken down into smaller parts called fragments and these fragments are then sent through the network.

In the next lecture we will talk about the IP version 6 and mobile IP.