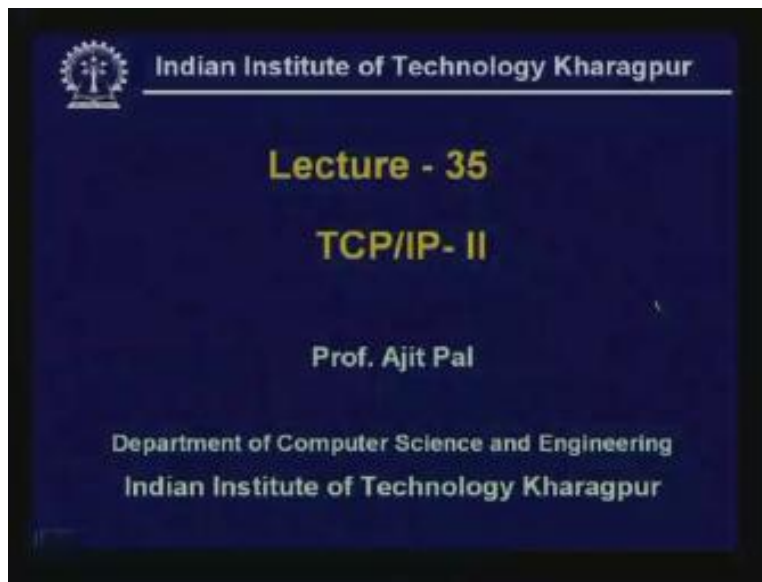


Data Communication
Prof. A. Pal
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur
Lecture - 35
TCP/IP - II

Hello viewers, we shall continue our discussion on TCP/IP.

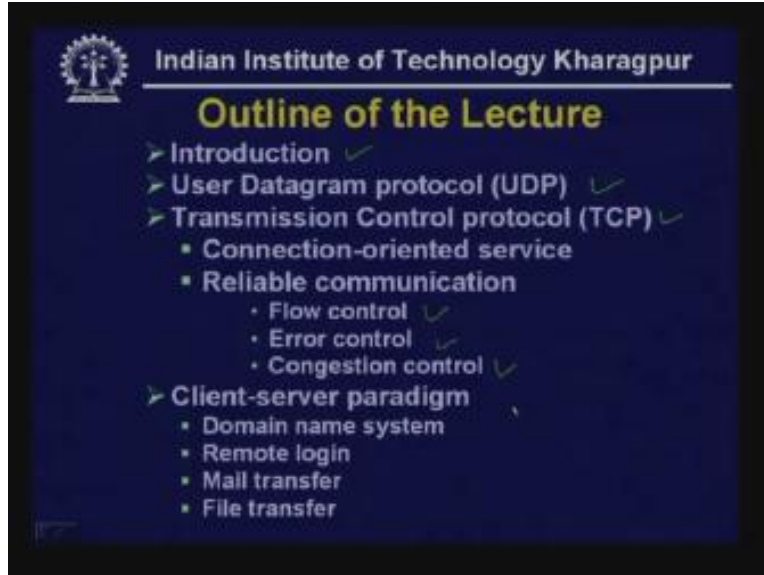
(Refer Slide Time: 00:59)



In the last lecture we have covered some of the issues of TCP/IP particularly the network layer protocols and some lower layer protocols. In this lecture we shall consider the transport layer protocols and also some of the application layer protocols. Here is the outline of today's talk.

First I shall give a brief introduction which will put you in perspective of the need for UDP and TCP that means the transport layer protocols. And we shall introduce two transport layer protocols; one is user datagram protocol known as UDP in short and another is transmission control protocol TCP. As we shall see UDP is connectionless unreliable datagram service. On the other hand TCP is a connection oriented service and it provides you reliability with the help of flow control, error control and congestion control which I shall discuss in detail in this lecture.

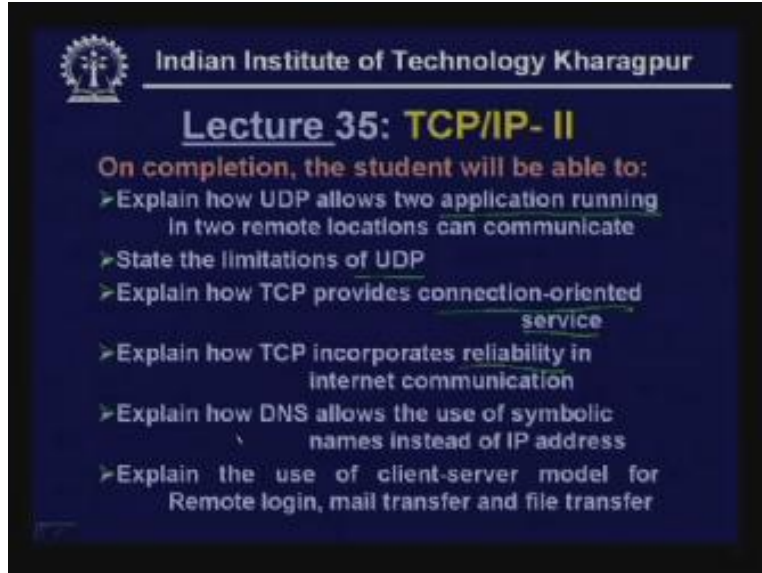
(Refer Slide Time: 02:05)



Then we shall see how the communication take place using the TCP/IP based on client-server paradigm and some applications which work on client-server paradigm such as domain name system, remote login, mail transfer we shall briefly discuss about them.

And on completion the students will be able to explain how UDP allows two applications running into remote locations can communicate. They will be able to also state the limitations of UDP because it is not connection oriented, it is unreliable so because of that it has got some limitations so the students will be able to state that or identify it. Then they will be able to explain how TCP provides connection oriented service the mechanism behind it and they will be able to explain how TCP incorporates reliability in internet communication.

(Refer Slide Time: 03:20)



Indian Institute of Technology Kharagpur

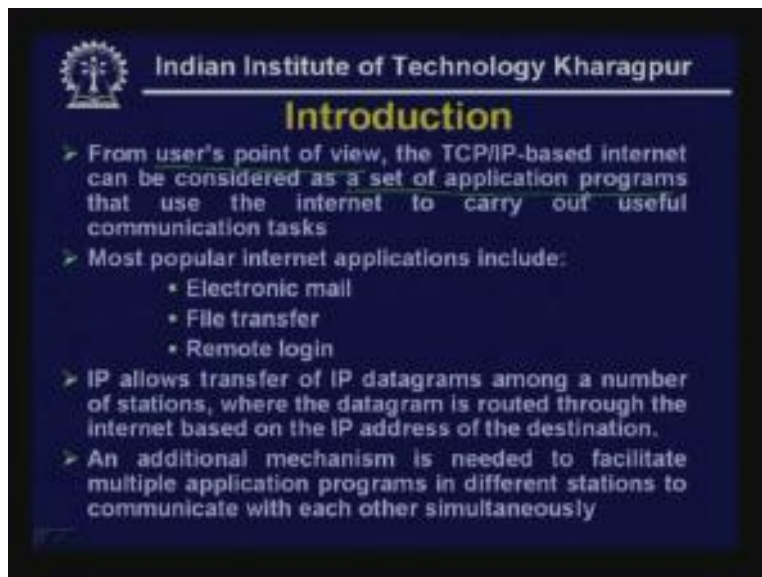
Lecture 35: TCP/IP- II

On completion, the student will be able to:

- Explain how UDP allows two application running in two remote locations can communicate
- State the limitations of UDP
- Explain how TCP provides connection-oriented service
- Explain how TCP incorporates reliability in internet communication
- Explain how DNS allows the use of symbolic names instead of IP address
- Explain the use of client-server model for Remote login, mail transfer and file transfer

They will also be able to explain how DNS domain name system allows the use of symbolic names instead of IP address which we have introduced in detail in the last lecture. The students will be also able to explain the use of LAN server model for remote login, mail transfer and file transfer and this client-server model is a basis of any distributed algorithm. Here is the introduction to put you in perspective.

(Refer Slide Time: 4:01)



Indian Institute of Technology Kharagpur

Introduction

- From user's point of view, the TCP/IP-based internet can be considered as a set of application programs that use the internet to carry out useful communication tasks
- Most popular internet applications include:
 - Electronic mail
 - File transfer
 - Remote login
- IP allows transfer of IP datagrams among a number of stations, where the datagram is routed through the internet based on the IP address of the destination.
- An additional mechanism is needed to facilitate multiple application programs in different stations to communicate with each other simultaneously

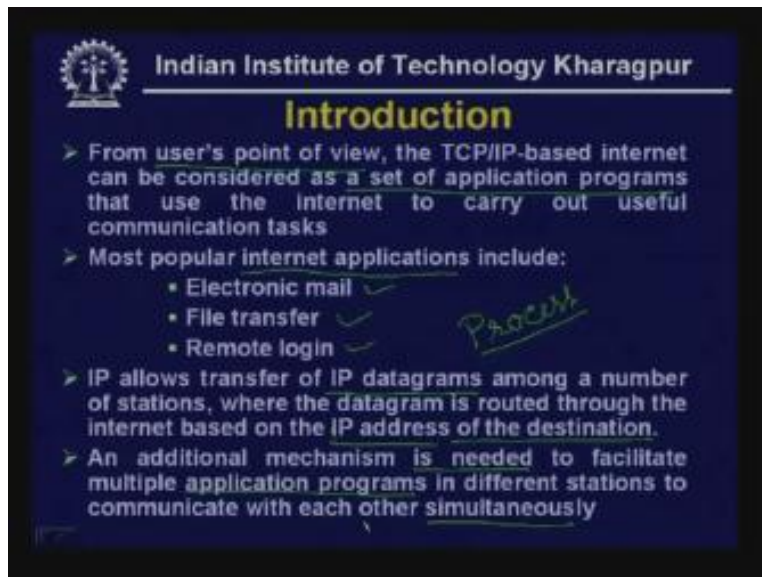
If you look from user's point of view you will find that TCP/IP based internet is nothing but a set of application programs that use the internet to carry out useful communication tasks. So a number of applications are running and how TCP/IP uses the underlying

network for that is one view of the user. Particularly the most popular internet applications that we use in our day to day life are electronic mail, file transfer and remote login. How these applications run using TCP/IP is what will be introduced in this lecture.

We have seen that internet protocol allows transfer of IP datagrams among a number of stations where the datagram is routed through the internet based on IP address of the destination. We have seen in the last lecture that internet protocol allows communication between two computers or two hosts that is done with the help of IP address. However, an additional mechanism is needed to facilitate multiple application programs which are known as processes.

What is a process? A process is nothing but a program is execution. So an application program in execution is called a process and how two processes can communicate in different stations with each other simultaneously is the need for this TCP.

(Refer Slide Time: 05:43)



Indian Institute of Technology Kharagpur

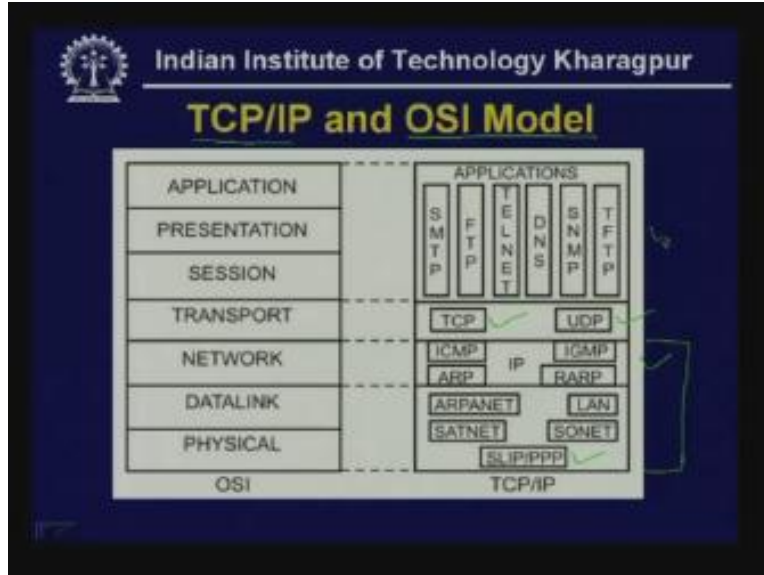
Introduction

- From user's point of view, the TCP/IP-based internet can be considered as a set of application programs that use the internet to carry out useful communication tasks
- Most popular internet applications include:
 - Electronic mail ✓
 - File transfer ✓
 - Remote login ✓
- IP allows transfer of IP datagrams among a number of stations, where the datagram is routed through the internet based on the IP address of the destination.
- An additional mechanism is needed to facilitate multiple application programs in different stations to communicate with each other simultaneously

Process

That means a particular host may be running a number of processes. As we know in a multi user environment multiple processes will be running simultaneously and how multiple processes can communicate simultaneously through the internet will be discussed how it is done using the transport layer protocol.

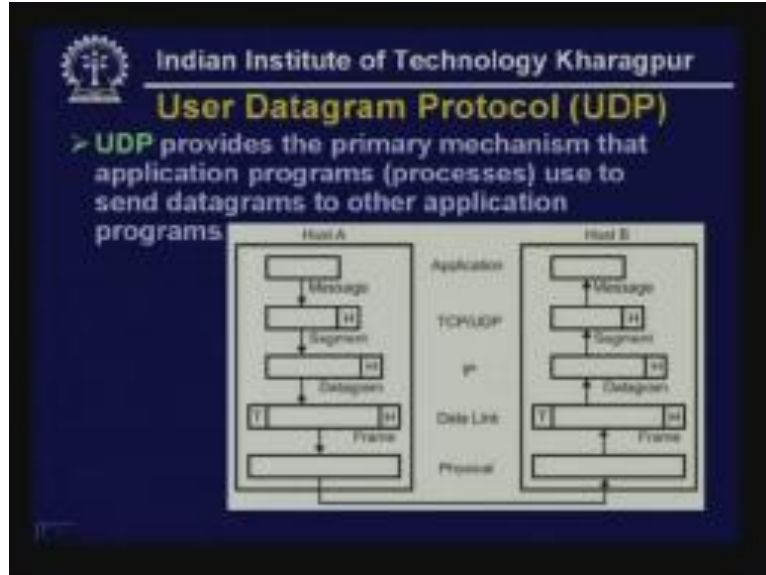
(Refer Slide Time: 06:50)



We have already discussed the relationship of the TCP/IP model with the OSI model and in the last lecture we have already introduced discussed some of the aspects of this part that means the network layer protocols, the internet protocols along with the companion protocols like ARP, RARP, ICMP and IGMP and we have briefly mentioned about this SLIP, and PPP also which were not covered earlier.

In this lecture we shall discuss first the UDP, then TCP and we shall also discuss some of the applications because ultimately the users are interested in applications and of course the TCP/IP is the underlying protocol which shall be used for that purpose. So first let us focus on the simplest of the protocols the user datagram protocol or UDP in short.

(Refer Slide Time: 07:18)



UDP provides the primary mechanism that application programs use to send datagrams to other application programs. Suppose an application program is running in host A it will send the message to one of the two transport layer protocol either TCP or UDP but in this particular case we are considering UDP and that UDP puts a header and that segment is sent and that goes to the IP layer and IP layer converts it into a datagram by adding a header and that datagram goes to the data link layer making each of these a frame by putting a separate header and a trailer and that frame is transmitted bit by bit through the physical layer and that goes to the other side to the internet and at the other end as it reaches the destination host then it goes off to the data link layer, IP layer and header and trailers are removed and ultimately the message is delivered to the application of the host.

(Refer Slide Time: 08:27)

Indian Institute of Technology Kharagpur

User Datagram Protocol (UDP)

➤ **UDP** provides the primary mechanism that application programs (processes) use to send datagrams to other application programs

This is how communication takes place with the help of the UDP. As I was mentioning multiple processes can be running in a single host and that requires what is known as multiplexing and demultiplexing. So multiple hosts will be running and they communicate using a concept known as port mechanism.

(Refer Slide Time: 08:52)

Indian Institute of Technology Kharagpur

Multiplexing, Demultiplexing and Ports

- UDP is responsible for differentiating among multiple sources and destinations within one host
- The multiplexing and demultiplexing operation are performed using the port mechanism

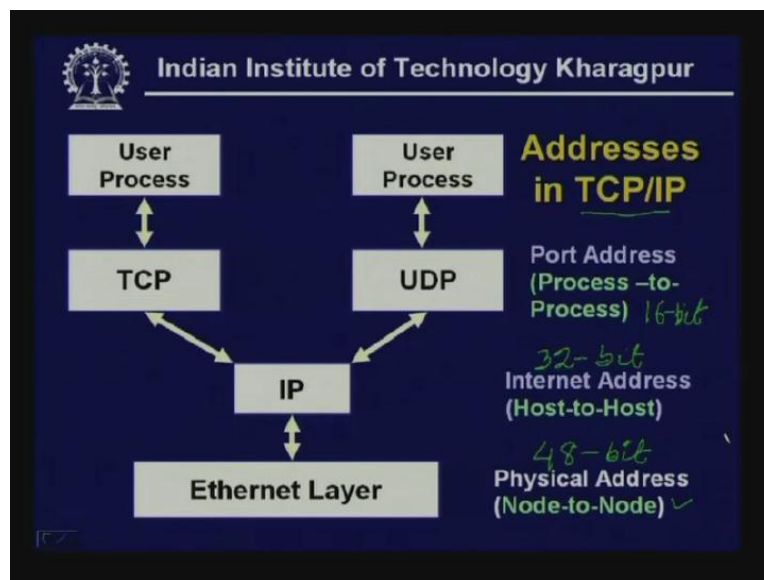
So UDP is responsible for differentiating among multiple sources and destinations within one host. So what is happening, you have got a single host and multiple applications are running and each application is associated with a port number so each application transfer messages to different ports and gives it to the UDP and UDP segment is then passed on to the IP. So here what we are doing is known as multiplexing.

Similarly the datagrams will be coming to the IP layer which will forward to UDP and UDP will identify different applications based on port numbers and it will do the demultiplexing and send it to the respective applications based on suitable port numbers. So here we are using another address which is port number.

We have already discussed about the two types of addresses, the physical address. The physical addresses are used by the data link layer for example for local area networks we use the physical layer address and as we know that physical layer address is 48-bit Ethernet address.

On the other hand the internet address that is IP address is from host to host that is 32-bit address. Now, as you can see there are port addresses which are identified as processes so it is for process to process communication which is 16-bit. So now we have three different types of addresses to be used in TCP/IP such as the physical address, internet address and port address.

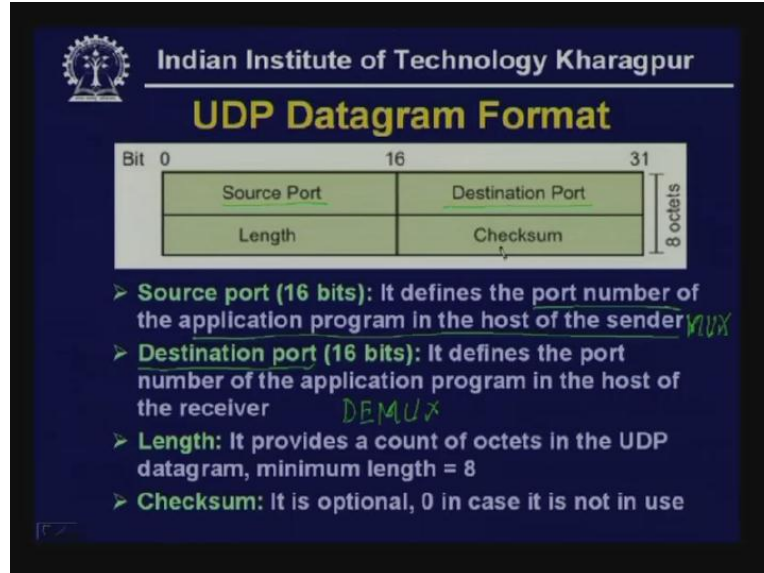
(Refer Slide Time: 10:45)



Here is the UDP datagram format. As you can see it has got a source port and destination port and the source port defines the port number of the application program in the host of the sender. That means sender may be running a number of applications so that particular port number gives you the source as from which application that particular datagram has come.

Similarly, the destination port number defines the port number of the application program in the host of the receiver. So we see that with the help of this we are doing demultiplexing and with the help of the source address we are doing multiplexing. This length field provides a count of octets in the UDP datagram and as you can see the minimum length is 8 octets which is essentially the header of the UDP.

(Refer Slide Time: 11:58)



As I mentioned UDP is an unreliable protocol. Essentially there is no need for error detection that's why the error detection is optional in case of UDP. So whenever it is not used this is made 0 so 0 is written here in this checksum field and of course there are situations where it is being used so in such a case it is not 0. As I mentioned the transport layer addresses are specified by 16-bit port numbers so these port numbers are assigned with the help of an agency known as Internet Assigned Number Authority IANA and the addresses are divided in three categories. For example, this is the range you can say (Refer Slide Time: 12:59) it is divided into three parts, 0 to 1023, this part is known as well known ports and these well known ports are controlled by Internet Assigned Number Authority IANA.

On the other hand, 1024 to 49151 up to this these are your well known ports and these are registered ports. To use these ports one has to register with the IANA. These are not well known ports but to use any of these addresses one needs to register with IANA. On the other hand there are other codes starting from 49152 to 65535 which can be used by using a 16-bit address so with the help of the sixteen bit address this is the maximum number one can use and these numbers can be dynamically assigned by user as and when needed.

(Refer Slide Time: 14:15)

Indian Institute of Technology Kharagpur

Port Numbers

- > Transport layer addresses are specified by 16-bit Port numbers
- > Internet Assigned Number Authority (IANA) has divided addresses in three ranges:
 - > Well-known ports (0 to 1023), controlled by IANA
 - > Registered ports (1024 to 49151), can only be registered with IANA
 - > Dynamic ports (49152 to 65535) are neither controlled by IANA nor can be registered.

Diagram: A horizontal bar representing a 16-bit range from 0 to 65535. The range 0-1023 is labeled 'Well-known ports'. The range 1024-49151 is labeled 'Registered ports'. The range 49152-65535 is labeled 'Dynamic ports'. Handwritten notes include '16-bit' and '49152-65535'.

So, for this it is not necessary to register with IANA and also it is not necessary to use the well known ports. The user can use them dynamically in different situations. These are the three different categories of port addresses that are commonly used. Here are the well known ports used by UDP.

(Refer Slide Time: 15:20)

Indian Institute of Technology Kharagpur

Well-known Ports used by UDP

PING

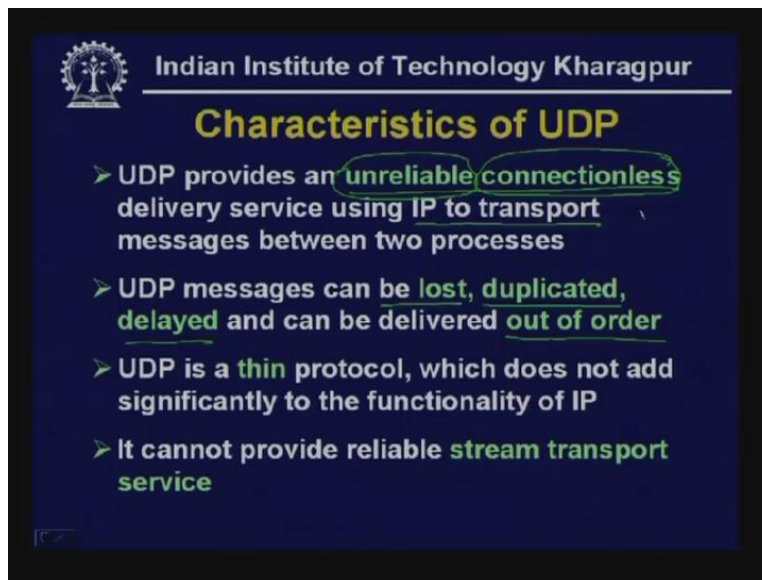
Port	Protocol	Description
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Nameserver	Domain Name Service
67	Bootpc	Server port to download bootstrap information
68	Bootpc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

There are some port numbers which are provided as well known ports. For example, port number seven is used for the purpose of echo, sometimes we do the ping operation which is being performed with the help of this port number, then there are other applications

like SNMP Simple Network Management Protocol and some special port number is assigned.

These are the common well known ports used by UDP given here for different applications. Let us now focus on the characteristics of UDP. We have seen that UDP provides an unreliable connectionless delivery service using IP to transport messages between two processes. Because it is unreliable and it is connectionless the outcome is the messages can be lost, duplicated, delayed and can be delivered out of order.

(Refer Slide Time: 16:03)



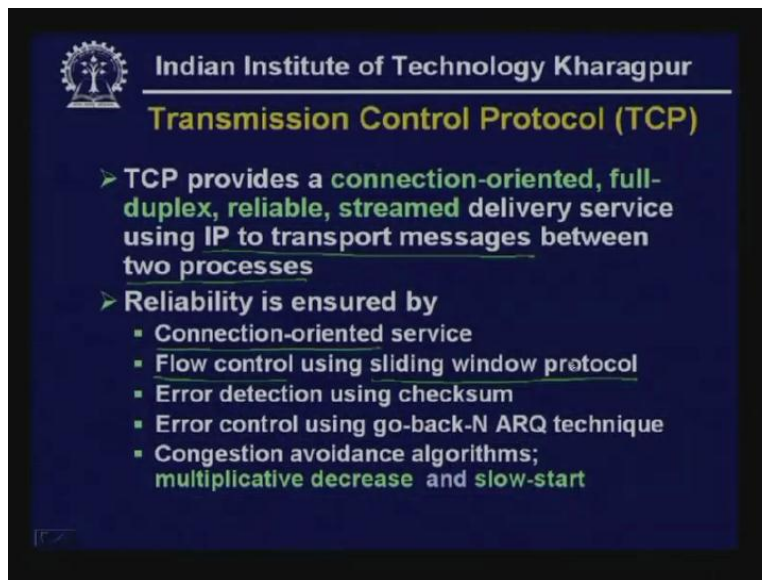
So if a packet is lost the sender will not know about it. If a packet is damaged the receiver will not know that a datagram has been received in error condition, it has become corrupted. Similarly, datagrams can pass through different routes and they can reach out of order so at the receiving end they have to be put in order. These are the limitations of the user datagram protocol.

However, UDP is a very thin protocol. As we have seen it has got only 8 octet header. As a consequence it is not having any high overhead, the overhead is much less in this particular case. However, it does not add significantly to the functionality of IP. We have seen that IP also provides you the connectionless service.

However, it provides you service from host to host and the only additional feature that UDP adds is from process to process rather than from host to host that is the only thing that is being provided. And also it cannot provide reliable stream transport service which is needed in many applications. And for many applications reliability or reliable communication is the key or is very important so in such a case we have to look for another protocol, we have to use another protocol which is known as transmission control protocol or TCP.

So TCP provides a connection oriented full-duplex reliable stream delivery service using IP to transport messages between two processes. In both cases IP is being used to transport messages between two hosts and then with the help of the port address it is communicated between two processes. And the reliability is ensured because of connection oriented service, because of flow control used in TCP and it uses the sliding window protocol. We know that by using sliding window protocol we can do flow control, so that is being performed here. It does error detection using checksum and also it uses error control using go back-N-ARQ technique so it performs error control as well.

(Refer Slide Time: 18:45)



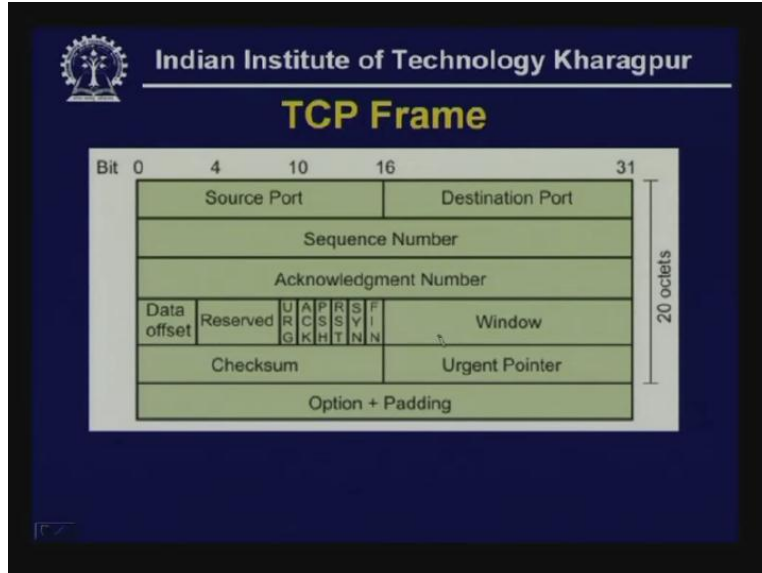
Indian Institute of Technology Kharagpur

Transmission Control Protocol (TCP)

- TCP provides a **connection-oriented, full-duplex, reliable, streamed** delivery service using IP to transport messages between two processes
- Reliability is ensured by
 - **Connection-oriented service**
 - **Flow control using sliding window protocol**
 - **Error detection using checksum**
 - **Error control using go-back-N ARQ technique**
 - **Congestion avoidance algorithms; multiplicative decrease and slow-start**

Moreover, it also performs congestion control by using congestion avoidance algorithms such as multiplicative decrease and slow start which we shall discuss later on. Now let us have a look at the TCP frame. Obviously the TCP frame is not very thin it has got 20 octets the source code, destination port and so on. apart from the source port and destination port addresses each of them are of 16-bit there are additional fields such as sequence number, acknowledgment number, data offset, reserved bits and several flag bits 6 flag bits and there is a window which is used for flow control then there is checksum which is used for error detection then there is an urgent pointer which is used in some situations and then we have the optional field plus padding.

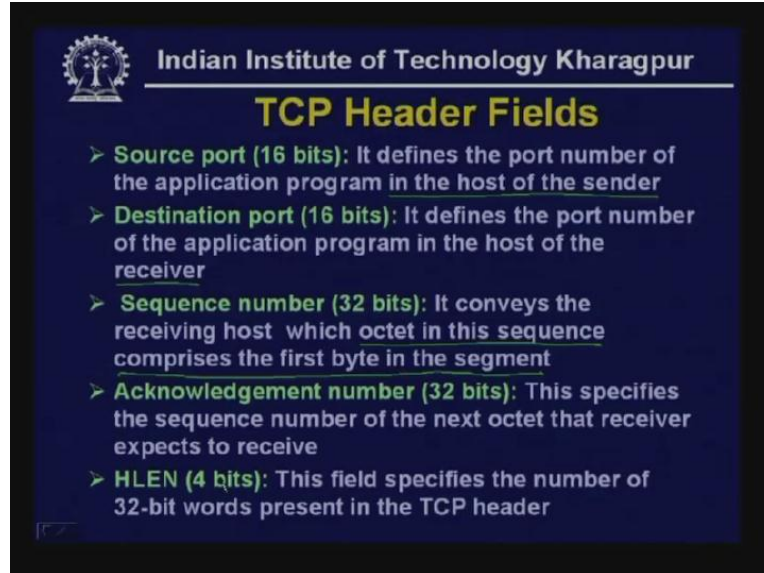
(Refer Slide Time: 19:45)



So, apart from the header you have got the optional field and padding and then comes the data. Let us look at the function of these different fields.

- Source port as usual defines the port number of the application program in the host of the sender.
- Destination port defines the port number of the application program in the host of the receiver.
- Sequence number conveys the receiving host which octet in this sequence comprises the first byte in the segment. The counting is done with the help of octets or in terms of bytes. So the sequence number is specified in terms of the byte number and the octet with which the segment starts.

(Refer Slide Time: 20:52)



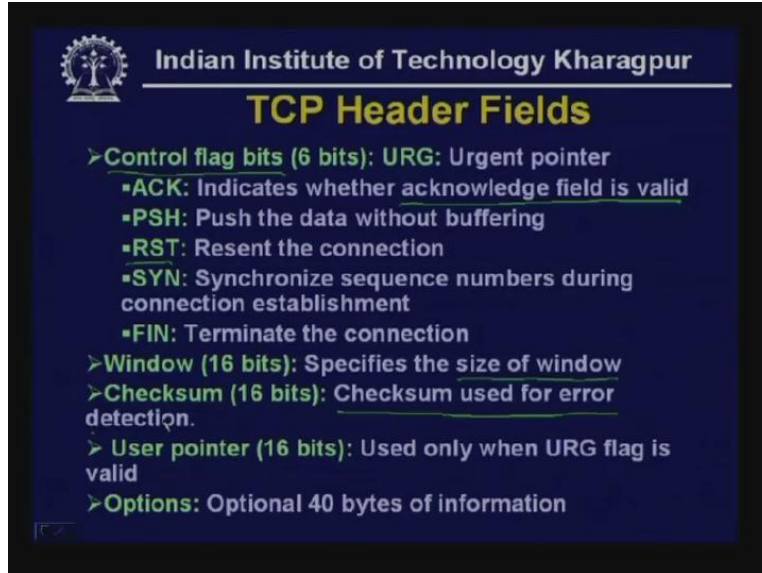
Acknowledgment number specifies the sequence number of the next octet that expects to receive and in both cases 32-bit fields are used.

- Header length is 4-bit. This bit specifies the number of 30-bit words present in that TCP header. Thus, with the help of this 4-bit field the number of 32 words as you have seen can be 1 2 3 4 5 or it can be more or it can go up to 20 octets so it specifies in terms of the 32-bit words.

Then as I mentioned there are several control flag bits. First one is;

- URG stands for urgent pointer and urgent pointer is used in some specific situation.
- Acknowledgement flag bit indicates whether acknowledgment field is valid. So if this bit is not 1 then acknowledgment field is invalid or is not used.
- PSH stands for push the data without buffering so this is a special situation where pushing of data has to be used without buffering.
- RST reset the connection. as we shall see this flag bit is used when connection establishment is done.
- SYN synchronize sequence numbers during the connection establishment so synchronize or SYN flag bit is used at the time of connection establishment.
- FIN stands for finished which is used for terminating a connection.
- 16-bit window specifies the size of the window that is the buffer size available and a 16-bit checksum is used.

(Refer Slide Time: 22:53)



Indian Institute of Technology Kharagpur

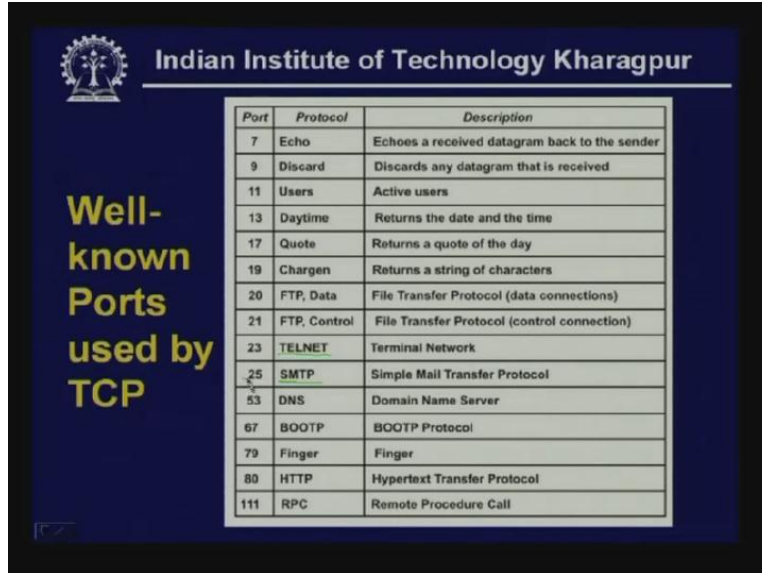
TCP Header Fields

- > **Control flag bits (6 bits):** URG: Urgent pointer
 - **ACK:** Indicates whether acknowledge field is valid
 - **PSH:** Push the data without buffering
 - **RST:** Resent the connection
 - **SYN:** Synchronize sequence numbers during connection establishment
 - **FIN:** Terminate the connection
- > **Window (16 bits):** Specifies the size of window
- > **Checksum (16 bits):** Checksum used for error detection.
- > **User pointer (16 bits):** Used only when URG flag is valid
- > **Options:** Optional 40 bytes of information

As I have already discussed, the checksum can be calculated by using ones complement addition in terms of sixteen bits then complementing it and then there is a user pointer which is used only when URG flag is valid. So when the URG flag is valid then this particular pointer is being used and they are optional, 40 byte information can be provided if necessary. So the TCP also has got a number of well known ports.

As you can see port number 7 is used for echo, 9 for discard and there are several other applications like TELNET that has got a well known port which is used, this TELNET is used for remote login then SMTP Simple Mail Transfer Protocol which is used for mail transfer which has got the well known port number 25 domain name server with a well known port number 53 and HTTP which is used for Hypertext Transfer Protocol which has got a well known port number 80 and so on. The important applications which are being used with the help of TCP/IP is provided with well known port numbers.

(Refer Slide Time: 23:50)

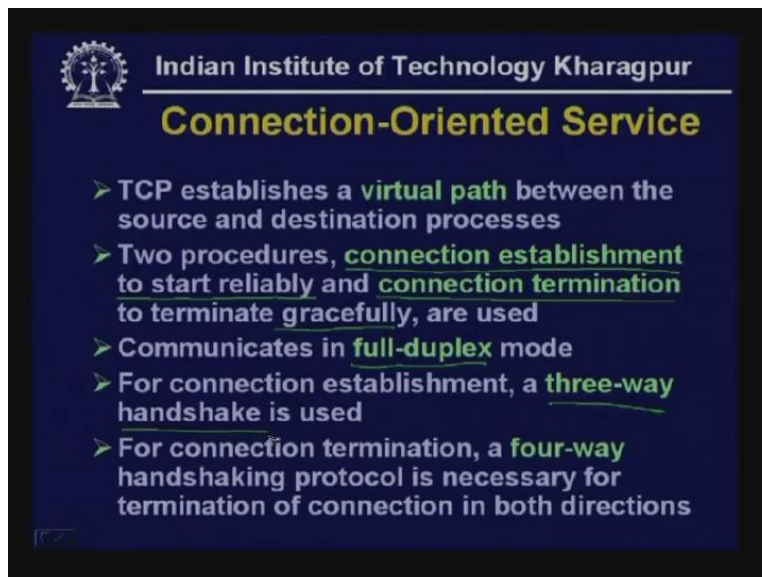


The slide features the IIT Kharagpur logo and name at the top. On the left, the text 'Well-known Ports used by TCP' is displayed in yellow. A table with three columns (Port, Protocol, Description) lists various services and their corresponding ports.

Port	Protocol	Description
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
20	FTP, Data	File Transfer Protocol (data connections)
21	FTP, Control	File Transfer Protocol (control connection)
23	TELNET	Terminal Network
25	SMT P	Simple Mail Transfer Protocol
53	DNS	Domain Name Server
67	BOOTP	BOOTP Protocol
79	Finger	Finger
80	HTTP	Hypertext Transfer Protocol
111	RPC	Remote Procedure Call

Now let us focus on the connection oriented service. As I mentioned the TCP establishes a virtual path between the source and destination processes before any application terminates. so there are two processes involved; one is connection establishment which is used to start the connection reliably and the connection termination with the help of which connection is terminated gracefully, so these two are being used and as I mentioned communication is full-duplex and for connection establishment a three-way handshake protocol is being used. Let us first discuss that.

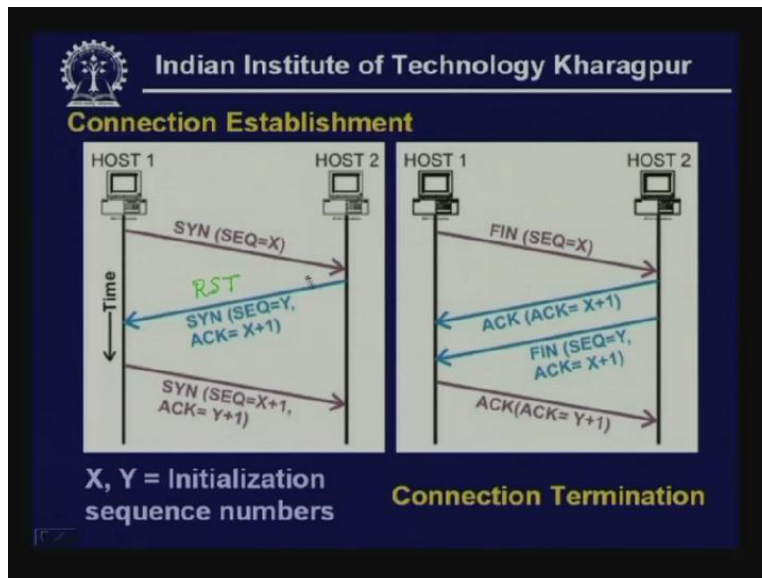
(Refer Slide Time: 25:00)

- 
- The slide features the IIT Kharagpur logo and name at the top. The title 'Connection-Oriented Service' is in yellow. Below it, a list of six bullet points describes TCP's characteristics.
- TCP establishes a **virtual path** between the source and destination processes
 - Two procedures, **connection establishment** to start reliably and **connection termination** to terminate gracefully, are used
 - Communicates in **full-duplex** mode
 - For connection establishment, a **three-way handshake** is used
 - For connection termination, a **four-way** handshaking protocol is necessary for termination of connection in both directions

So, to establish a connection as I mentioned that SYN flag bit is set and then sequence number is set to some value X and here (Refer Slide Time: 25:24) you are sending one datagram to the host on the other side. In response to that the host on the other side will send an acknowledgment, that acknowledgment will have a SYN bit set and it will provide you the sequence number Y and acknowledgment number $X + 1$.

Here you can see it is using one sequence number X that means the next datagram that is expected will start with $X + 1$ byte or octet so this is being used when the host is ready for sending data. whenever it is not ready there is a possibility that the port number is being busy for some other application, in such a case connection cannot be established so in such a case instead of SYN datagram that RST flag is set and RST packet is being sent from the HOST 2 to HOST 1 that means the connection cannot be established.

(Refer Slide Time: 26:45)



However, when HOST 2 is ready for establishing connection then this particular packet is sent to the other side and in response to that the HOST 1 will send another SYN packet with sequence number $X + 1$ but earlier it was X and acknowledgment number $Y + 1$. the sequence number was Y here so one byte is being used for the purpose of establishment of synchronization and acknowledgment number Y means this HOST 1 is also ready to receive packets starting with $Y + 1$ as the beginning of the packet. This is how the three-way protocol is being used to establish a connection.

For connection termination a four-way handshaking protocol is necessary for termination of connection in both directions. For the termination purpose as I mentioned it is a full-duplex protocol. However, we may consider it as two half-duplex or two simplex connections. That means two connections can be terminated independently.

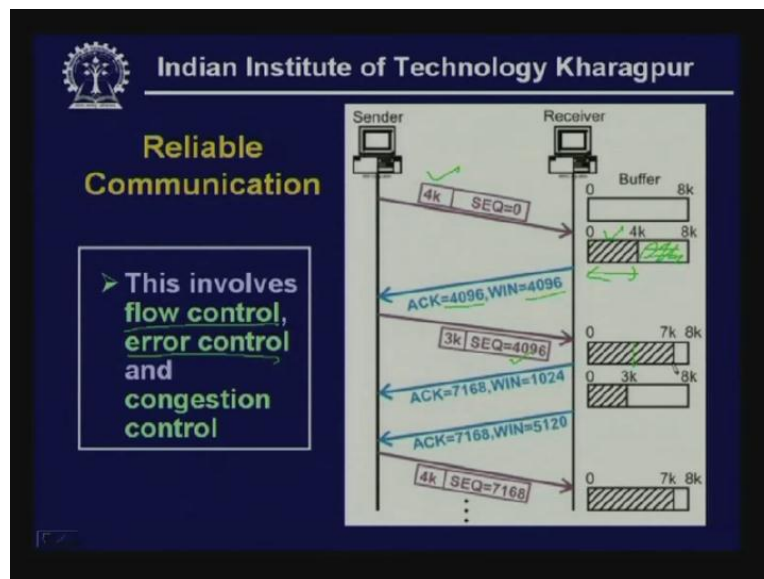
So, for example if HOST 1 wants to terminate the connection for sending data from this end to the other end then it will send one packet the datagram with FIN or finished flag

bits with sequence number X and on receiving this the HOST 2 will send an acknowledgment packet with ACK is equal to X plus 1. When this is being received the host connection which is used for sending data from host A to host B is terminated. however the data transfer in the other direction from HOST 2 to HOST 1 can continue and if HOST 2 wants to terminate that HOST 2 again sends another datagram of packet with FIN flag bit set with sequence number Y and acknowledgement X plus 1 and on receiving that the HOST 1 sends an acknowledgment packet with ACK is equal to Y plus 1.

Now, whenever both the ends want to terminate connection these two can be combined to a single packet. That means both acknowledgement flag bit and FIN flag bit can be set and by doing that ACK part that is acknowledgment filled is sent X plus 1. Only these two flag bits are to be sent and instead of two signals only one message will go from HOST 2 to HOST 1 and that way it can become a three-way protocol to terminate connection when both the connections are to be terminated simultaneously.

Now let us focus on the reliability.

(Refer Slide Time: 31:17)



As I mentioned reliability is ensured by flow control, error control and congestion control. Now, flow control is being done with the help of a buffer. So whenever sender is sending data it sends the size of the data that is 4k which is the size of the data and that starts with the sequence number 0. And as it reaches the receiver then four k buffer gets filled up as you can see this part gets filled up (Refer Slide Time: 30:41) and it sends an acknowledgment with 4096 so that is the beginning of the sequence number which the receiver is expecting and window size 4096 because 4k window is now free.

Now if the sender has a 3 kilo byte data to send then it can send it starting with sequence number which is the beginning of the octet 4096 and as it reaches the destination as you

can see so 4 plus 3 is equal to 7 kilo bytes of buffer gets filled. Now it sends an acknowledgment with acknowledgment filled 7168 that is the beginning of the sequence and window size 1k because this is window size available at the receiving end.

If the sender has more data to send it will not be able to send now because buffer is not available here. Now the application program can read the data and transfer it may be to the secondary storage so in such a case the buffer will be empty and the buffer will be free. And suppose the application reads 4k bytes these 4k bytes are read by the application.

So now the 5k buffer becomes empty so window size becomes 5120 and it sends an acknowledgment frame to the sender and on receiving that the sender can send another 4 kilo byte of segment starting with sequence number 7168. This is how the flow control is being performed and the buffer size is communicated by the receiver whenever acknowledgement segment is sent from the receiver to the sender. This is how the flow control is being performed.

Let's see how error control is performed.

(Refer Slide Time: 32:55)

The slide is titled "Reliable Communication" and is from the Indian Institute of Technology Kharagpur. It contains two main sections:

- TCP Timers to keep track of lost or discarded segments:**
 - Retransmission
 - Persistence
 - Keep-alive
 - Time-waited
- Congestion avoidance algorithms; multiplicative decrease and slow-start**

Below the text is a graph showing "Congestion window" on the y-axis and "Transmission Number" on the x-axis. The graph illustrates the slow-start phase where the congestion window increases exponentially until it reaches a "Threshold". After this point, it enters a multiplicative decrease phase where the window size drops sharply. A "Time out" is indicated at the end of the first phase. The graph then shows the slow-start phase where the window size increases again until it reaches a new, lower "Threshold".

To perform error control a number of timers are used. There are four timers;

- Retransmission timer
- Persistence timer
- Keep-alive timer
- Time-waited timer

Retransmission timer is used in situations where suppose a frame has been sent and it is lost then the retransmission timer is started and whenever the time-out occurs and if

acknowledgement does not come then the frame is retransmitted. So the retransmission timer is used for the purpose of retransmission and whenever a frame is lost.

Then the persistence timer is used to avoid some kind of dead lock. Let us consider this situation that the buffer is now not enough it is only 1 k and a frame has been sent and acknowledgment has been sent by the receiver. Now on receiving this, the sender is waiting for acknowledgment having higher window size. Now another acknowledgment sent only when this buffer becomes empty with window size 5190.

Suppose this particular acknowledgment is lost what will happen is the sender will be waiting for the acknowledgment and the receiver will be waiting for the data and this leads to some kind of dead lock and this dead lock can be avoided with the help of this persistence timer. So whenever time-out occurs, in the persistence timer, the sender will send a persistence segment to the receiver and on receiving that this particular acknowledgment will be again sent by the receiver and data communication will start.

Keep-alive timer is used in situations where the communication is not taking place for a long time. Suppose a sender has no data to send for a long time in such a case when the data communication is not taking place for a long time the transmitter may assume that the receiver has now been turned off. So to know that a keep-alive frame is sent by the sender and if the receiver responds then connection is continued otherwise connection is terminated.

Time-waited timer is used at the time of termination. Whenever termination is performed it is performed immediately but the connection is kept open for two packet transmission time, two RTT that is your Receive Transmission Time to the Round-Trip Time this is the round-trip time (Refer Slide Time: 36:36). So for two round-trip time this timer waits before the connections are shut off.

Now let us focus on the congestion avoidance algorithm. Congestion is avoided by using two algorithms; one is multiplicative decrease and another is slow start.

As you can see in the beginning there are two reasons for any congestion control. One is because of the smaller receiver capacity and another is because of lower network capacity.

And as you have seen the receiver capacity is provided with the help of the window. So window size provides the receiver capacity. However, the network capacity is not provided by this window. For that purpose a separate window known as congestion window is used and the minimum of the two is used for sending data. So initially suppose 1k word is the segment size that is being allowed by the network so initially segment one is being sent and if no time-out occurs then the two segments are sent, then four segments are sent, then eight segments are sent then it increases exponentially until the threshold is being reached.

There is a threshold which is decided by the congestion window. And after that threshold it increases linearly that means it is incremented by 1k, 1k and so on so initially it is 1k,

then 2k, then 4k etc so exponentially it increases and after the threshold is increased it increases linearly. And whenever time-out occurs this threshold is halved, so whatever was the value here half of that is considered as the threshold and then again slow start is being performed and then multiplicative decrease is performed. This is how congestion avoidance is performed with the help of multiplicative decrease and slow start.

(Refer Slide Time: 39:03)

The slide is titled "Reliable Communication" and is from the Indian Institute of Technology Kharagpur. It contains two main sections:

- TCP Timers to keep track of lost or discarded segments:**
 - Retransmission
 - Persistence
 - Keep-alive
 - Time-waited
- Congestion avoidance algorithms; multiplicative decrease and slow-start**

A graph below these sections plots "Congestion window" on the y-axis against "Transmission Number" on the x-axis. The graph shows a sawtooth pattern where the congestion window increases linearly until it reaches a "Threshold". At this point, it drops sharply, a "Time out" event is indicated, and the window then begins to increase again. A second "Threshold" is shown at a lower level than the first. Handwritten green notes at the bottom of the slide include "1K, 1, 2k, 4k, 8k Window" and "Receiver capacity, Network capacity".

There is another problem here that retransmission time. Question arises how you find out the retransmission time. How do you decide the retransmission time is another issue here in connection with the retransmission time.

In case of internet a probability of time-out is varied, it is over a large space large time period. In case of LAN the delay can be very small but the delay can vary over a large range in case of internet. So if you choose a retransmission time T_1 then there will be many retransmission time then on the other hand if you choose T_2 as the retransmission time then unnecessarily you are wasting time.

Now what is being done is initially a suitable value or an exact value is chosen based on the measurement. based on what is rate of the round-trip delay time a retransmission timer is set and after that if there is no time out then this timer is increased on the other hand if there is time out then it is decreased. So in this way the retransmission timer is controlled and variable RTT is being used the retransmission timer is used and based on the based current estimate of the timer.

(Refer Slide Time: 41:11)

Indian Institute of Technology Kharagpur

Reliable Communication

- TCP Timers to keep track of lost or discarded segments:
 - Retransmission
 - Persistence
 - Keep-alive
 - Time-waited
- Congestion avoidance algorithms; **multiplicative decrease** and **slow-start**

Handwritten notes: Variable RTT

We have discussed how the flow control and congestion control is being performed. Now let us focus on the client-server paradigm that is being used for various applications.

(Refer Slide Time: 41:35)

Indian Institute of Technology Kharagpur

Client-Server Paradigm

- The way the application programs communicate with each other is based on **client-server** model
- This provides the foundation on which **distributed algorithms** are developed
- A **client** process formulates a request, sends it to the server and then awaits for response
- A **server** process awaits a request at a **well-known port** that has been reserved for the service and sends responses

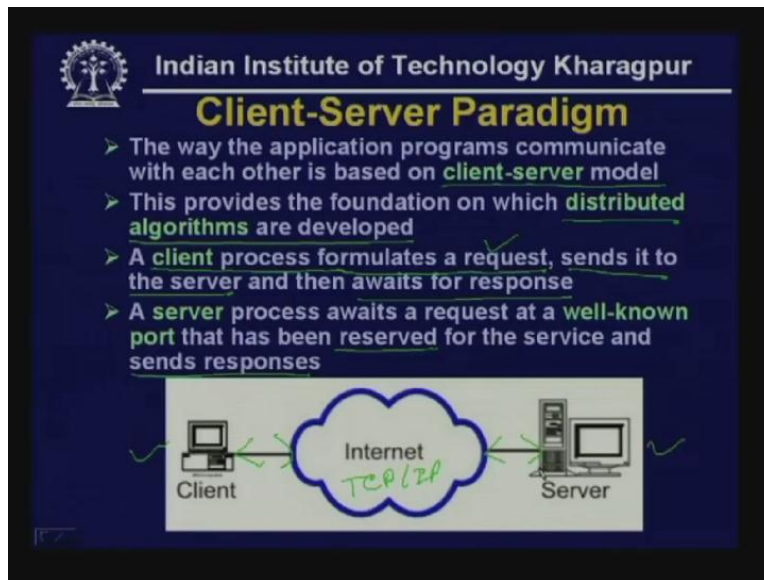
```
graph LR; Client[Client] --- Internet((Internet)); Internet --- Server[Server]
```

As we shall see the way the application programs communicate with each other is based on client-server model. In fact this provides the foundation on which distributed algorithms are developed. As we shall see, in the present day context, many algorithms are distributed algorithms and all the distributed algorithms work based on the client-server model or client-server paradigm. So let us focus on the discussion of client-server paradigm and discuss how exactly it works.

You have got a client process which formulates a request. So you have got two systems; one is client, another is server so a client runs a process which formulates a request, sends it to the server and then awaits the response.

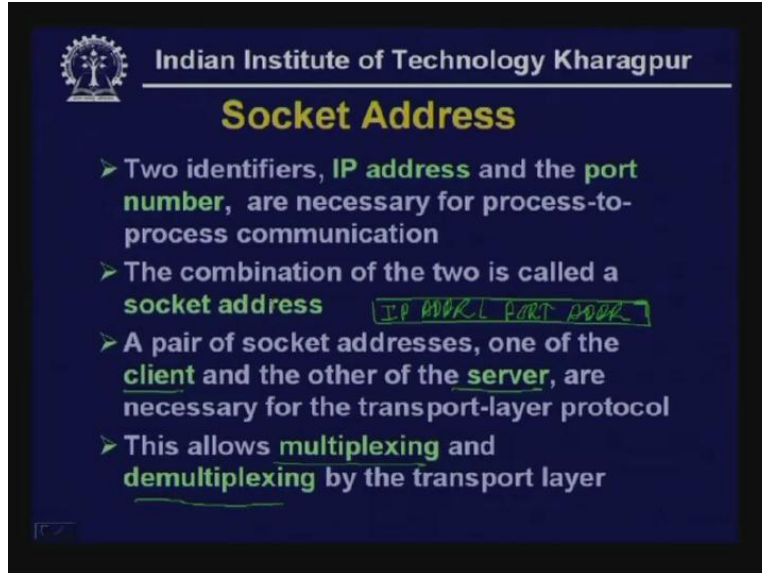
On the other side the server process awaits a request so it is not just slipping; it is waiting for a request at a well known port then after receiving that it sends responses. So the server process awaits a request at a well known port that has been reserved for different applications then sends response. This is how it works and through the internet communication takes place based on TCP/IP. Of course the communication is full-duplex as I mentioned, both client and server communicate with each other by using full-duplex communication.

(Refer Slide Time: 43:25)



To do that there are two identifiers; one is your IP address, another is your port address. So **a pair of information** is necessary for process to process communication. Just the IP address cannot serve the purpose and apart from IP address it is necessary to have the port address. These two together are known as socket address. So a socket address comprises two components; one is your IP address and another is your port address and these two together gives you the socket address. And a pair of socket addresses one at the client and the other at the server are necessary for transport layer protocol. That means at the client side you will have a socket address and at the server side another socket address will be necessary for communication between the two so these allow multiplexing and demultiplexing by the transport layer.

(Refer Slide Time: 44:55)



Indian Institute of Technology Kharagpur

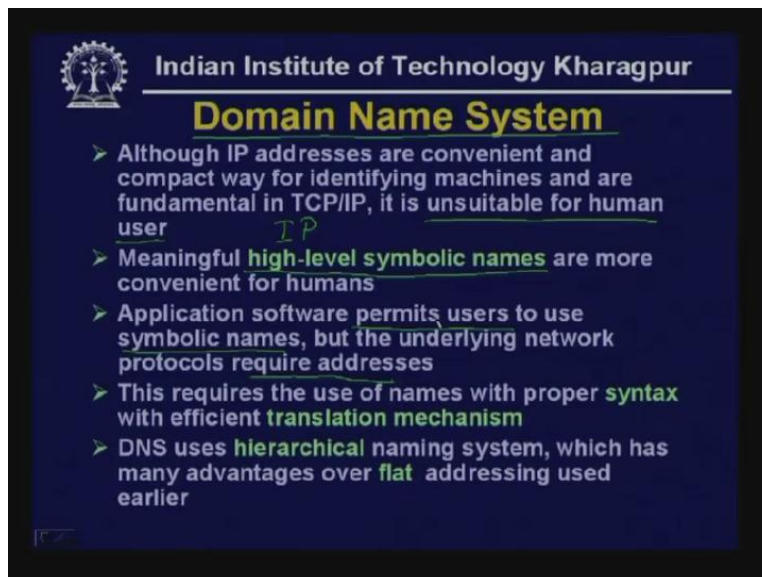
Socket Address

- Two identifiers, **IP address** and the **port number**, are necessary for process-to-process communication
- The combination of the two is called a **socket address** IP ADDRESS PORT ADDRESS
- A pair of socket addresses, one of the **client** and the other of the **server**, are necessary for the transport-layer protocol
- This allows **multiplexing** and **demultiplexing** by the transport layer

That means the socket address is used for multiplexing and demultiplexing at the transport layer with the help of IP address and port address. So the IP address is provided by the IP header and port address is provided by the TCP header or UDP header whatever it is.

Now let us focus on some applications which work on which work using the client-server paradigm. The one very important application protocol is known as domain name system.

(Refer Slide Time: 46:25)



Indian Institute of Technology Kharagpur

Domain Name System

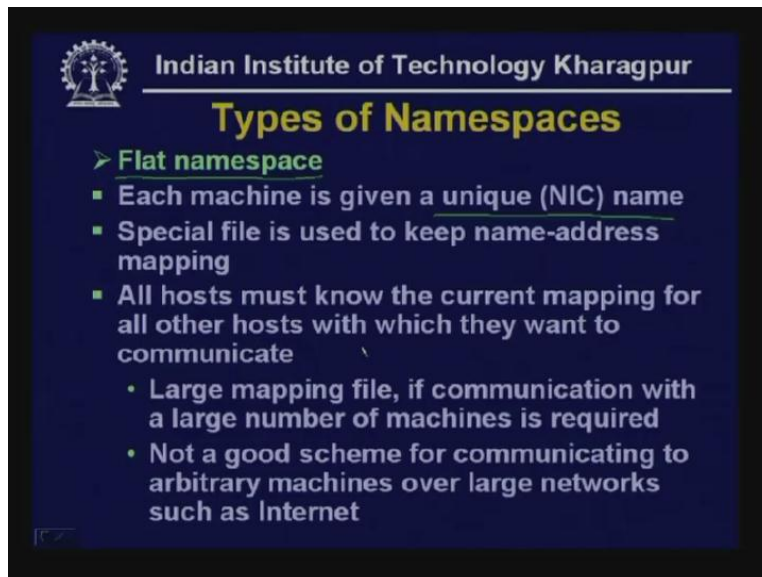
- Although IP addresses are convenient and compact way for identifying machines and are fundamental in TCP/IP, it is unsuitable for human user IP
- Meaningful **high-level symbolic names** are more convenient for humans
- Application software permits users to use symbolic names, but the underlying network protocols require addresses
- This requires the use of names with proper **syntax** with efficient **translation mechanism**
- DNS uses **hierarchical** naming system, which has many advantages over **flat** addressing used earlier

We have seen that IP addresses are a convenient and a compact way of identifying machines are fundamental in TCP/IP. So IP addresses are used for communication between two hosts. However, it is unsuitable for human user. Human user cannot really remember a long sequence of zeros and ones although by using dotted decimal notation it has been made slightly user friendly but it is not enough. So it is necessary to use high level symbolic names for convenience of the humans.

This domain name system application software permits users to use symbolic names but the underlying network protocols requires addresses. That means the TCP/IP protocol requires IP address that means the machines require IP address and the users require symbolic address so the gap has to be bridged and that is being done by using the domain name system and that requires a suitable syntax and also an efficient translation mechanism.

There are two ways of providing address; one is known as hierarchical naming system, another is flat addressing system. So the DNS uses hierarchical naming system. We shall see how it is being done. Before that let us briefly discuss about the flat namespace.

(Refer Slide Time: 47:35)



Indian Institute of Technology Kharagpur

Types of Namespaces

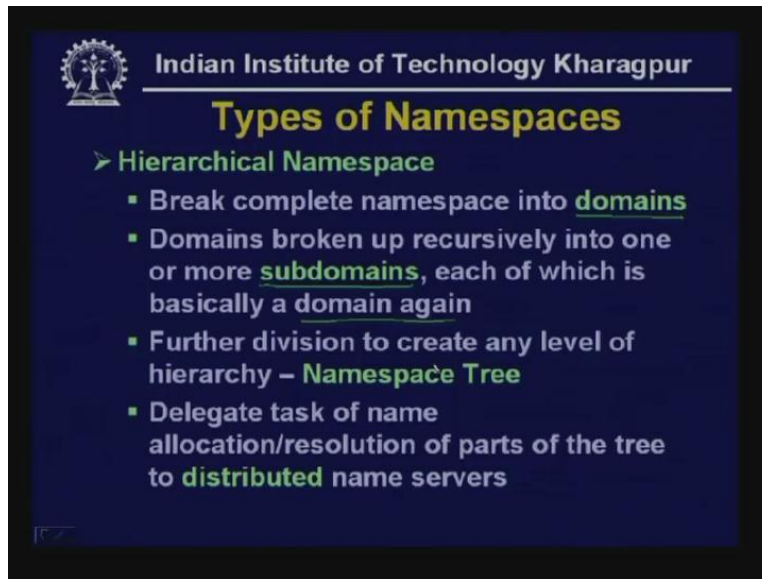
- **Flat namespace**
 - Each machine is given a unique (NIC) name
 - Special file is used to keep name-address mapping
 - All hosts must know the current mapping for all other hosts with which they want to communicate
 - Large mapping file, if communication with a large number of machines is required
 - Not a good scheme for communicating to arbitrary machines over large networks such as Internet

In this case each machine is provided with a unique name as assigned by the NIC Network Information Centre and a special file is used to keep name address mapping and all hosts must know the current mapping from all other hosts with which they want to communicate. This requires large mapping file if communication with a large number of machines is required.

Particularly in the present day internet context the size of the number of people that is being communicating with each other is very, very large so this will require a huge mapping file. It is not really a good scheme for communication communicating to arbitrary machine over a large network such as internet so this flat namespace which was

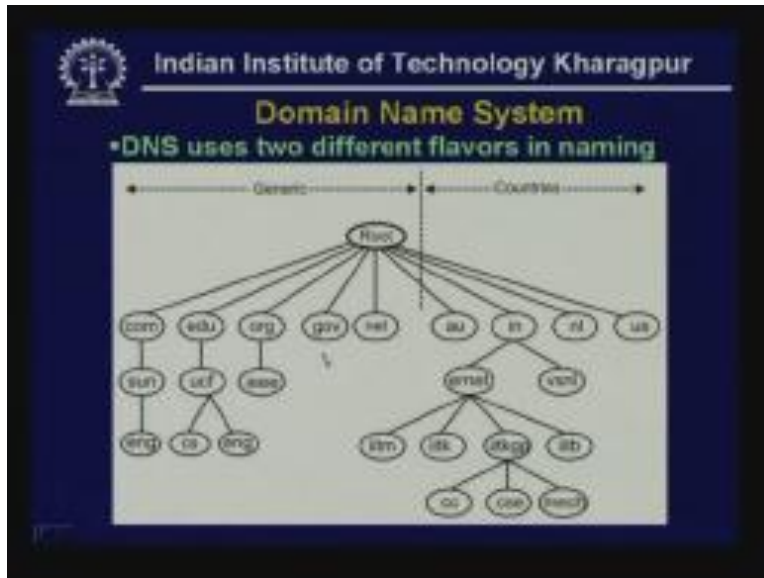
used earlier is not really suitable and for that reason hierarchical namespace has been used which break complete namespace into domains. So instead of providing a unique name it is divided into a number of domains. So domains are broken up recursively into one or more subdomains and each domain is divided into a number of subdomains for each of which is basically a domain again.

(Refer Slide Time: 48:40)



Further division to create any level of hierarchy is being provided with the help of the namespace tree. And it delegates the task of name allocation/resolution parts of the tree to distributed name server. The translation from name to address and address to name is performed in a distributed manner. This is how the domain names are given in a hierarchical manner. As you can see there are two flavors of naming systems; one is generic and another is based on country.

(Refer Slide Time: 49:22)

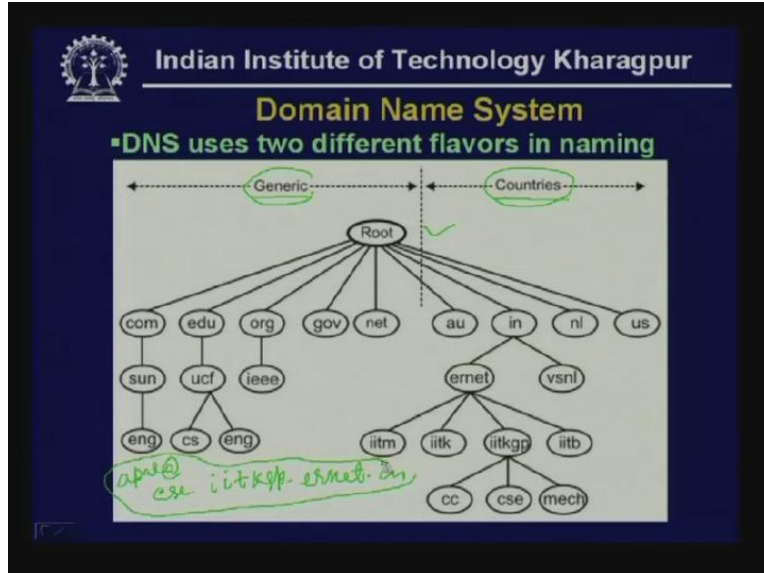


Generic is a com which is assigned to communication companies, edu for educational institutions, org for different organizations like IEEE, gov for government organizations and net for networking companies.

On the other hand AU for Australia, IN for India, US for USA which are based on the countries. So this is your root domain and under the root domain you have got several sub domains, it can be based on generic or country based. So you have got a number of subdomains and under each sub domain again you have got several subdomains. For example, under India in sub domain you have got ernet, vsnl and so on so under ernet again you can have different IITs like IITM, IITK, IITKGP, IITB and so on and then under IITKGP you have got CC, CSE, mechanical and so on.

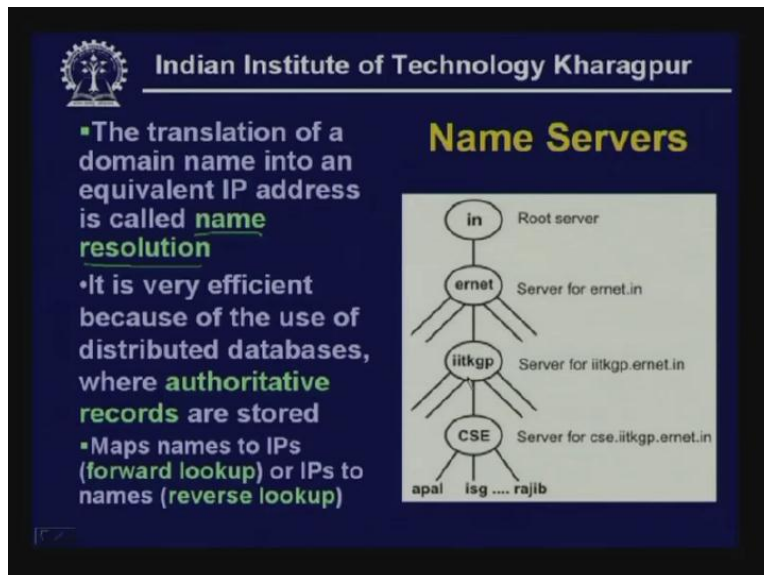
So you can see it starts with a root node and then you have got next layer of domains and then another layer of domains so in this way it can continue. And ultimately for example a name can be like cse.iitkgp.ernet.in that forms a name and any user connected to this server can be for example, apal@cse so this forms the complete name for sending any email or other things.

(Refer Slide Time: 51:03)



And the translation of a domain name into an equivalent IP address is called name resolution. It is very efficient because of the use of distributed databases and the databases are stored in different servers, as you can see some of the databases are stored in inserver, some of the databases are stored in ernet server, **some of the databases are stored in IIT KGP servers**, some of the databases are stored in cse server and so on.

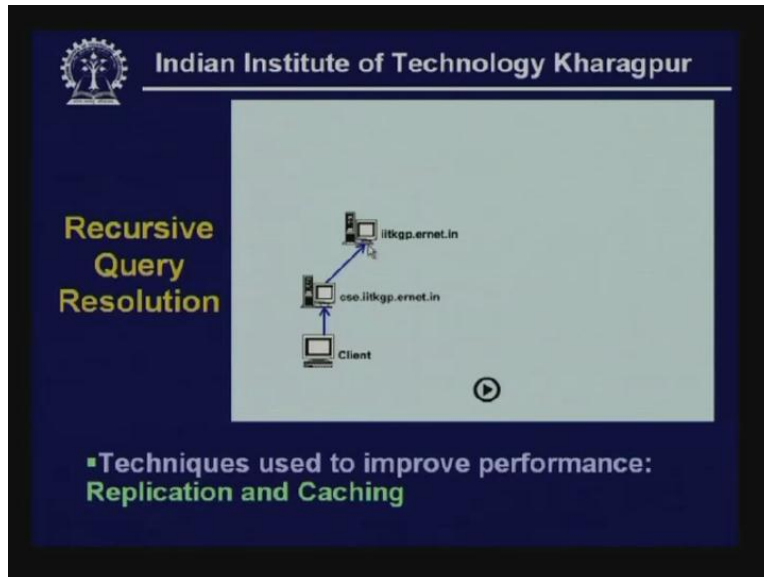
(Refer Slide Time: 51:40)



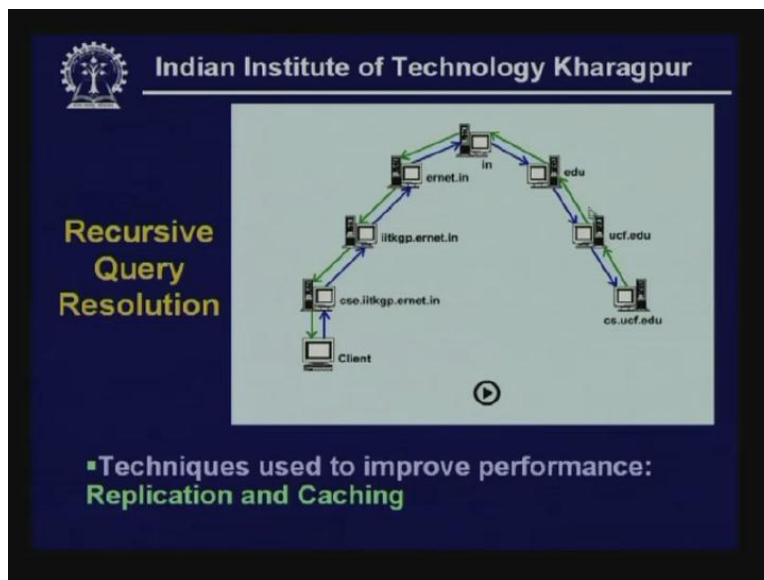
Then mapping is being performed in this way by using recursive query resolution. the client sends the query to the to the next server so cse iitkgp ernet in then it sends to the next server iitkgp ernet in that query goes to the next higher domain then it goes to the

ernet in then it goes to the in and it goes again to the edu and if the information is not absolute here then it goes to ucf.edu and then it goes to the cd.edu and it definitely has the record and it sends the IP address and goes back to the client.

(Refer Slide Time: 52:16)



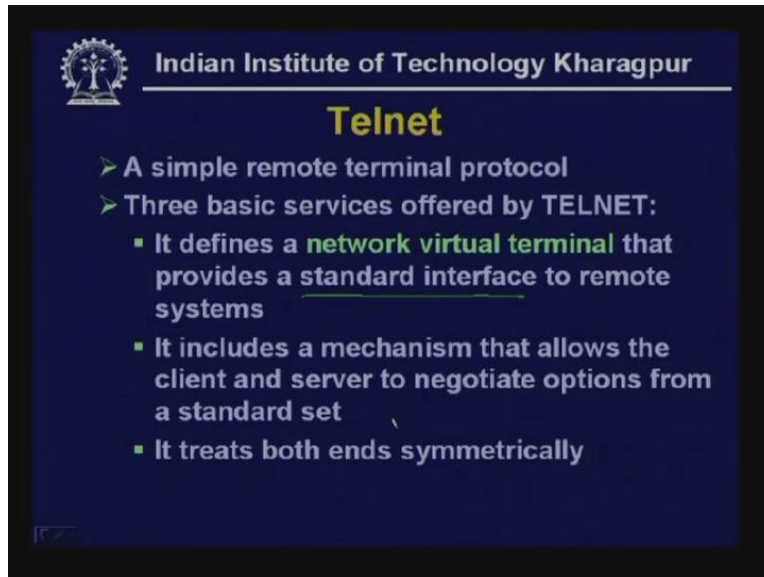
(Refer Slide Time: 52:40)



This is how the query is resolved that means the name is mapped to address so name to address mapping is performed in this manner. Here are some of the applications based on the client-server protocol. The Telnet is used for remote terminal protocol. There are three basic services offered by Telnet; it defines a network virtual terminal that provides a standard interface to remote systems. So any key press goes to the remote system and

also the response is displayed on the client system so in this way there is communication as if a computer is directly connected to a remote server so a client can be connected to a remote server this is how it works.

(Refer Slide Time: 53:55)



Indian Institute of Technology Kharagpur

Telnet

- A simple remote terminal protocol
- Three basic services offered by TELNET:
 - It defines a **network virtual terminal** that provides a standard interface to remote systems
 - It includes a mechanism that allows the client and server to negotiate options from a standard set
 - It treats both ends symmetrically

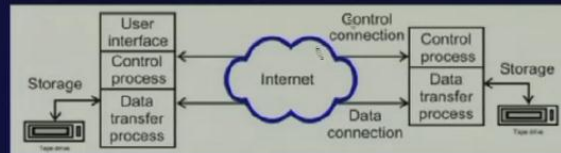
It includes mechanism that allows the client and server to negotiate options from a standard set. It treats both ends symmetrically. That means client to server communication is performed in symmetrical manner. Similarly the file transfer protocol is another protocol that uses the client-server paradigm and this is being used to transfer files between two remote machines through internet. A TCP connection is set up before the transfer and it persists throughout the session.

(Refer Slide Time: 54:25)



File Transfer Protocol (FTP)

- Used to transfer files between two remote machines through internet
- A TCP connection is set up before file transfer and it persists through the session
- More than one file can be sent before disconnecting the link
- Users view FTP as an interactive system



It includes mechanism that allows the client and server to negotiate options a standard set. Through the internet the files can be transferred from the server to the client and for that purpose two connections are necessary. One is control connection and another is data connection and more than one file can be sent before disconnecting the link. The user views FTP as an interactive system because file transfer can be done in an interactive manner with the help of this control connection then data transfer is performed with the help of this data connection.

Finally, electronic mail is among the most widely used available application services. The mail system buffers outgoing and incoming messages allowing transfer to take place in the background so some kind of mail gateway can be used, the protocol that is being used is known as SMTP.

(Refer Slide Time: 55:22)



Indian Institute of Technology Kharagpur

Electronic Mail

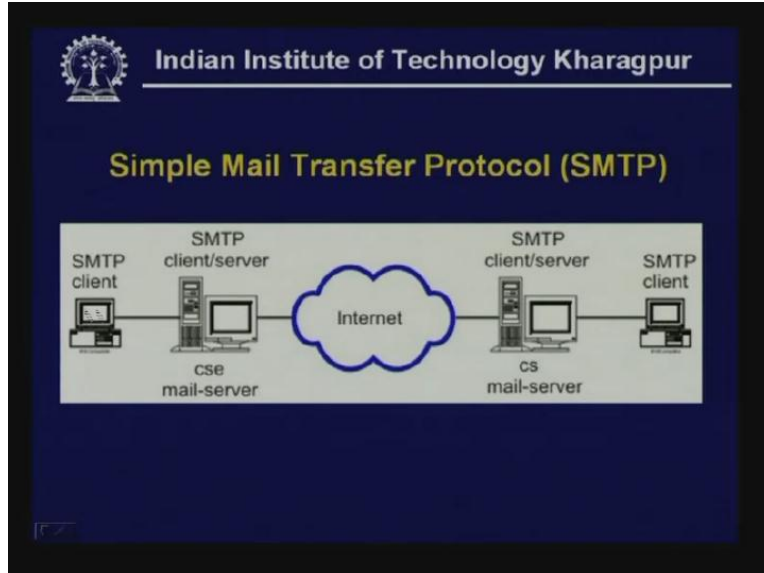
- Electronic mail is among the most widely available application services
- The mail system buffers (mail gateways) outgoing and incoming messages, allowing the transfer to take place in the background
- The TCP/IP protocol that supports electronic mail on the internet is called Simple Mail Transfer Protocol (**SMTP**), which supports the following:
 - Sending a message to one or more recipients
 - Sending messages that include text, voice, video, or graphics

The TCP/IP protocol that supports the electronic mail on the internet is called Simple Mail Transfer Protocol which supports the following:

- sending a message to one or more recipients
- sending messages that include text, voice, video or graphics

This is the basic system so you have got a client and here you have got a server and that server holds the messages that can be sent through the internet using TCP/IP to another server the mail server and that mail server is connected to another client and these two clients communicate with each other through these servers which act as the gateways. With this we come to the end of the discussion on TCP/IP. Here are the review questions.

(Refer Slide Time: 55:55)



(Refer Slide Time: 56:30)

Indian Institute of Technology Kharagpur

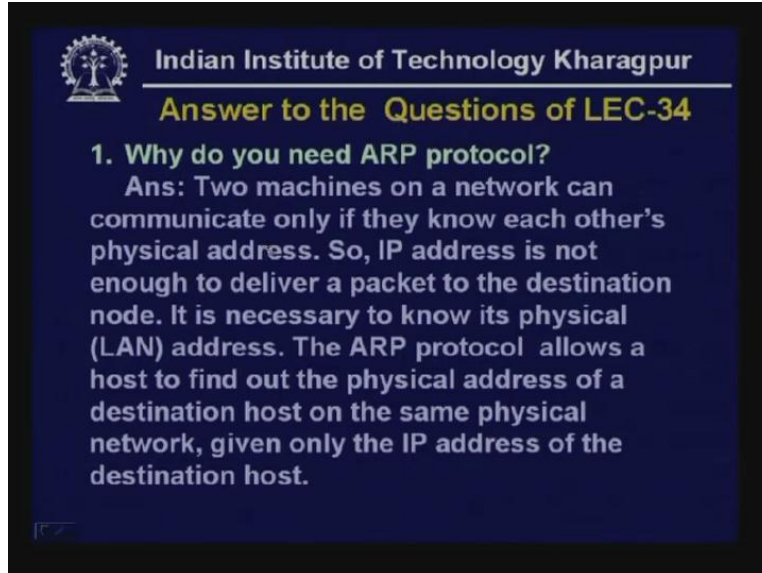
Review Questions

1. What is the main function of UDP protocol?
2. Why pseudo-header is added in a UDP datagram?
3. How TCP establishes and terminates connection?
4. What are the advantages of DNS?
5. Explain how FTP works?

To be answered in the next lecture

- 1) What is the main function of UDP PROTOCOL?
- 2) Why pseudo header is added in a UDP datagram?
- 3) How TCP establishes and terminates connection?
- 4) What are the advantages of DNS?
- 5) Explain how FTP works.

(Refer Slide Time: 56:45)



Indian Institute of Technology Kharagpur

Answer to the Questions of LEC-34

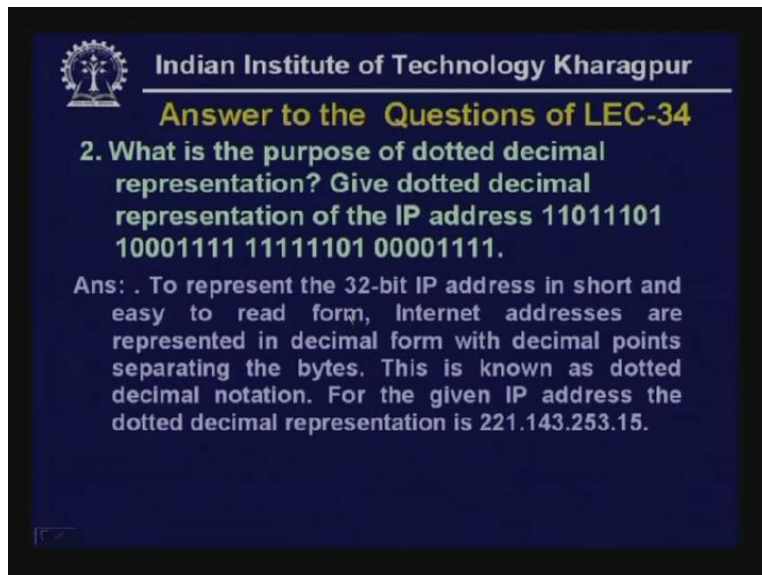
1. Why do you need ARP protocol?

Ans: Two machines on a network can communicate only if they know each other's physical address. So, IP address is not enough to deliver a packet to the destination node. It is necessary to know its physical (LAN) address. The ARP protocol allows a host to find out the physical address of a destination host on the same physical network, given only the IP address of the destination host.

Why do you need ARP PROTOCOL?

Two machines on a network can communicate only if each other they know each other's physical address. So, IP address is not enough to deliver the packets so you have to ((perform)) mapping from IP address to physical address that is being performed in ARP protocol as you have discussed in detail in the last lecture.

(Refer Slide Time: 57:09)



Indian Institute of Technology Kharagpur

Answer to the Questions of LEC-34

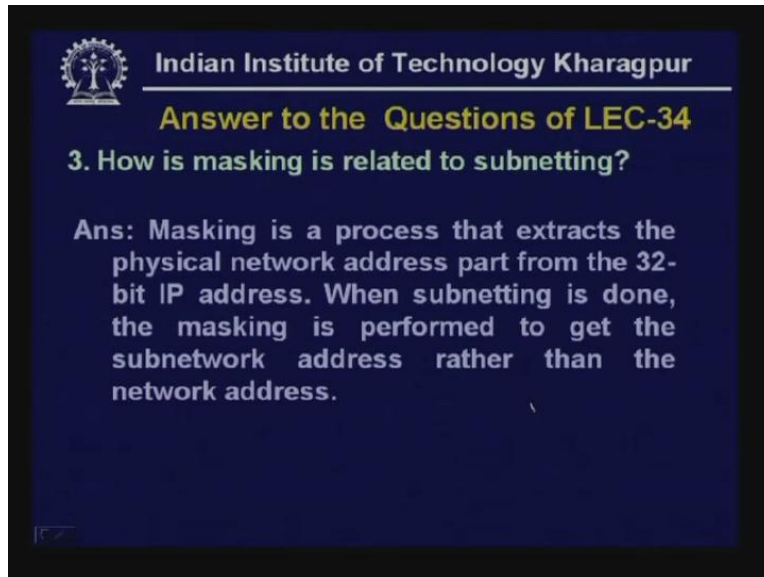
2. What is the purpose of dotted decimal representation? Give dotted decimal representation of the IP address 11011101 10001111 11111101 00001111.


Ans: . To represent the 32-bit IP address in short and easy to read form, Internet addresses are represented in decimal form with decimal points separating the bytes. This is known as dotted decimal notation. For the given IP address the dotted decimal representation is 221.143.253.15.

2) What is the purpose of dotted decimal representation? Give the dotted decimal representation of the IP address.

As you know the 32-bit field is divided into four octets and each is represented in decimal form and that gives the IP address in dotted decimal notation and for this bit sequence this is the dotted decimal notation.

(Refer Slide Time: 57:30)



 Indian Institute of Technology Kharagpur
Answer to the Questions of LEC-34

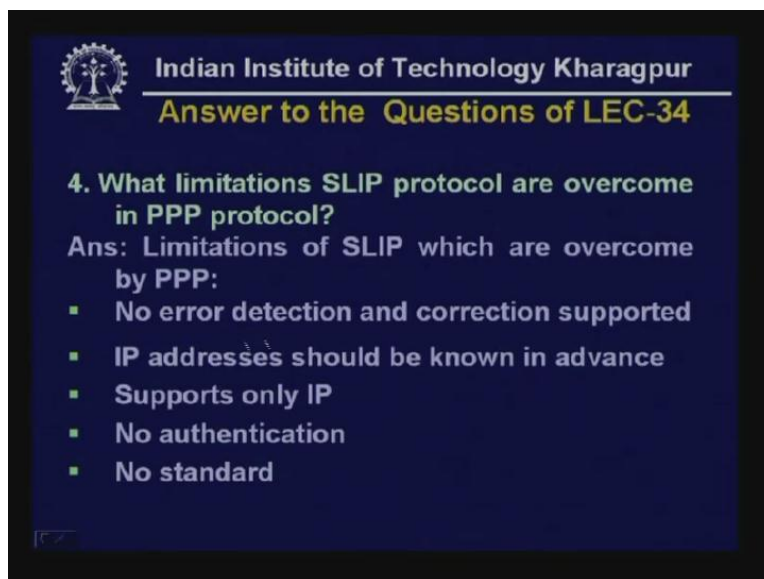
3. How is masking is related to subnetting?


Ans: Masking is a process that extracts the physical network address part from the 32-bit IP address. When subnetting is done, the masking is performed to get the subnetwork address rather than the network address.

3) How masking is related to subnetting?

Masking is the process that extracts the physical network address part from the 32-bit IP address. When subnetting is done the masking is performed to get the subnetwork address rather than the network address. So three levels of hierarchy has been used as we have discussed

(Refer Slide Time: 57:55)



 Indian Institute of Technology Kharagpur
Answer to the Questions of LEC-34

4. What limitations SLIP protocol are overcome in PPP protocol?

Ans: Limitations of SLIP which are overcome by PPP:

- No error detection and correction supported
- IP addresses should be known in advance
- Supports only IP
- No authentication
- No standard

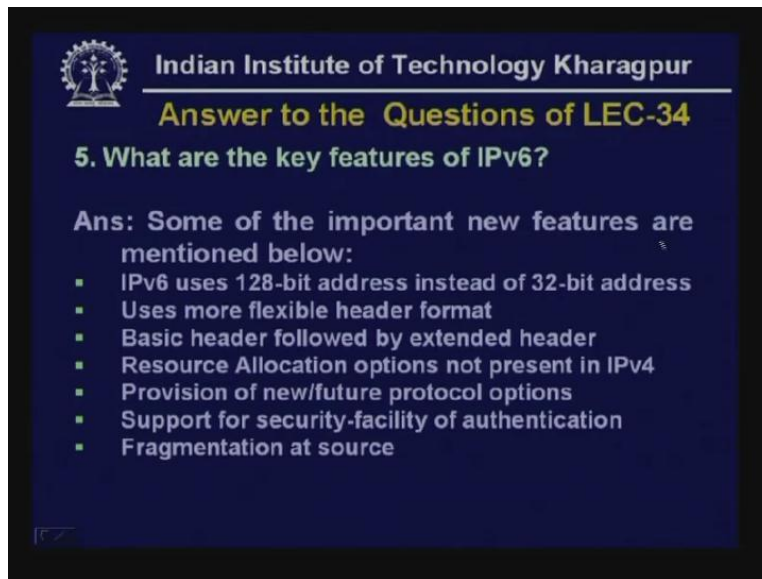
4) What limitations SLIP protocol are overcome in PPP protocol?

Limitations of SLIP which are overcome by PPP are;

- No error detection correction is supported
- IP addresses should be known in advance
- Supports only IP
- No authentication
- No standard

This is being overcome by using PPP protocol.

(Refer Slide Time: 58:15)



5) What are the key features of IPv6?

Some of the most important new features are mentioned below:

- IPv6 uses 128-bit address
- It uses more flexible header format and
- there are two parts basic header followed by extended header and
- resource allocation options are not present in IPv4
- provision for new and future protocol options are provided
- supports security-facility for authentication and
- fragmentation is done at source

These are the features of IPv6.

With this we come to the end of our discussion on TCP/IP, thank you.