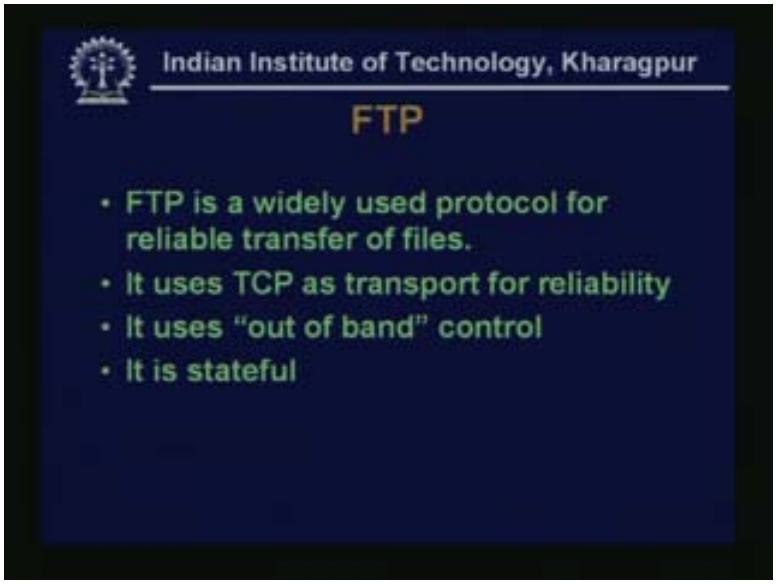
Computer Networks
Prof. S. Ghosh
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Lecture # 39
FTP-SMTP

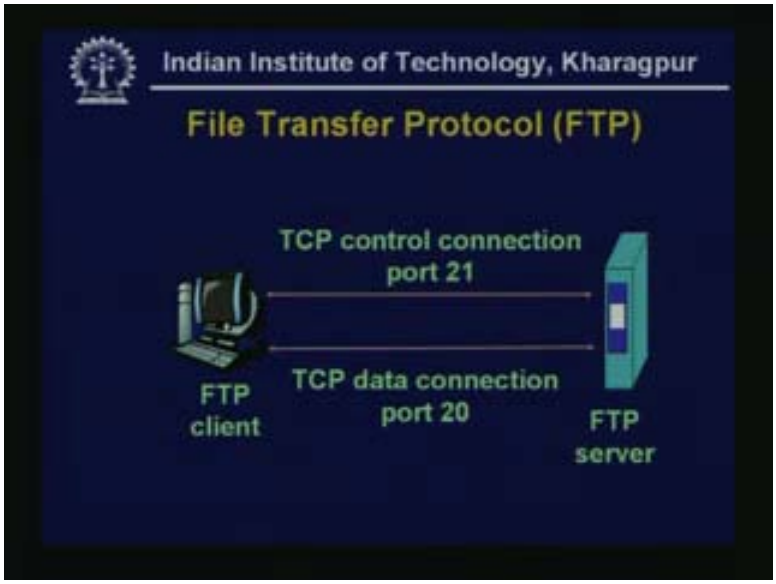
Good day, so today we are going to take up two important application layer protocols namely FTP and SMTP. So, first let us look at FTP.

(Refer Slide Time: 01:07 - 01:33)



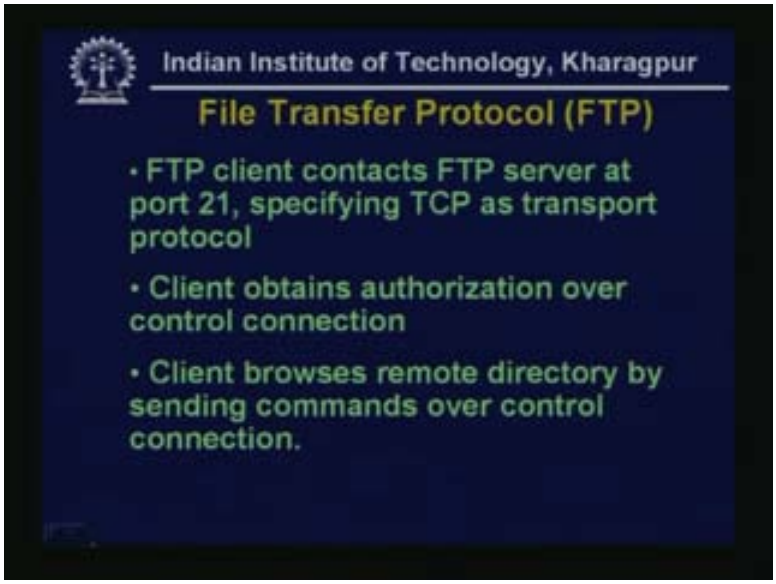
This is called File Transfer Protocol and it is a very widely used protocol for reliable transfer of files. It uses TCP as transport for reliability, it uses out of band control and this is stateful. So we will go into these one by one. First thing is that, we want this file transfer to be very reliable, in the sense that even missing one bit in between may actually corrupt the entire file. So we want a good degree of reliability and as we have seen in our discussion on TCP that one way to achieve this reliability is to use TCP. You use FTP for downloading files, now, as soon as you start an FTP to download a file it should start immediately unlike mail which may be delivered after sometime i.e. 10 or 15 minutes or so. That is one requirement of FTP. The other requirement is that the speed is not constant. The speed should remain constant but that is not a crucial requirement for FTP. A file may be transferred fast in the beginning, then if the network gets congested it may slowdown but then that is alright, in all these respects TCP is a very ideal transport protocol for running FTP.

(Refer Slide Time: 02:56 - 04:35)



Let us look into the general scheme. First of all, suppose this is an FTP client trying to contact an FTP server there is a TCP control connection at port 21. On the above we have mentioned the control connection and below we have mentioned a data connection. These two connections are different, that is why it says that its control is out of band. Actually, out of band control means, whenever two nodes are communicating this is a general term that if there is some control information going between the two, for example, in a telephone the connection setup there is control information, ringing etc there is a control information and when this control information and the actual data occupy two distinct channels then it is called out of band communication. That means the control is out of band. For control connection TCP uses different port and for data connection it uses a different port. One is a port number 21, the other is a port number 20 and these are basically well known ports. The request comes from the client through this port number 21. So the server opens a port number 20 with the client and starts communicating.

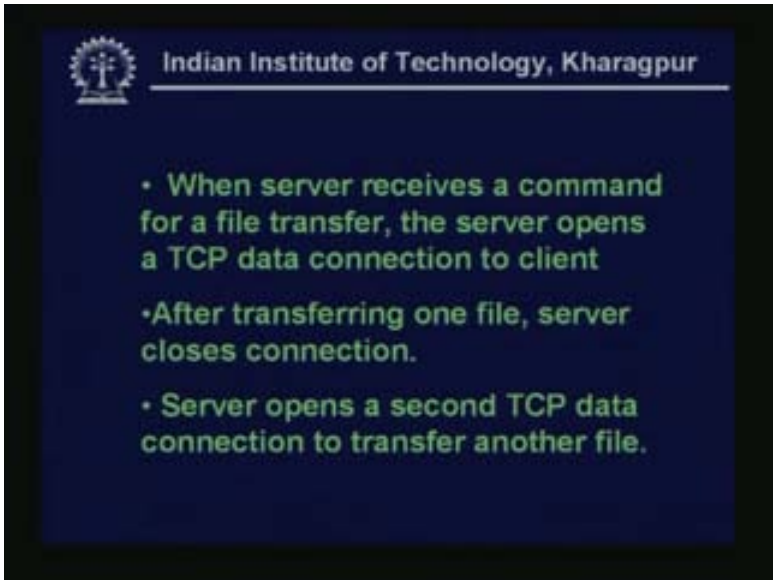
(Refer Slide Time: 04:35 - 06:37)



Now, FTP client contacts FTP server at port 21 specifying TCP as transport protocol. Client obtains authorization over control connection. Now this is important, when you want to download a file it is important to check whether whoever wants to download has a proper authorization or not because not all files and all information should be available to everybody in the world. There are questions of privacy and security etc so that is why some authorization is necessary.

Of course in this case what FTP does is, the authorization is rather at a rudimentary level in the sense that it is through users and passwords only but anyway that much of authorization is there and this authorization process goes through the control channel. Therefore client obtains authorization over control connection. A client browses remote directory by sending commands over control connection which means that, if the authorization phase goes through without any trouble then the current directory at the server end can be seen by the client from the other end by giving some commands. Now, in raw FTP, the commands are a little bit cryptic whereas there are FTP clients with good graphical user interface where you can actually graphically see the directory etc, this is the remote directory we are talking about it. Therefore he can look into the remote directory and select whatever file he wants to download.

(Refer Slide Time: 06:38 - 07:27)

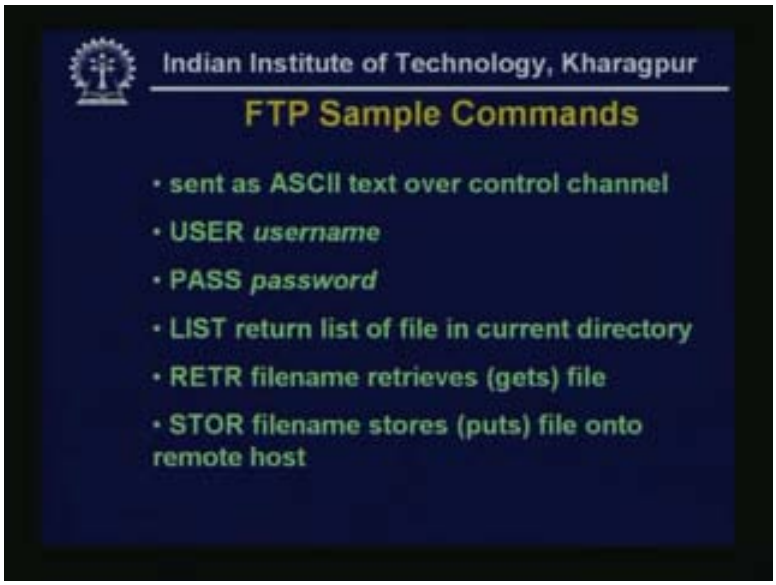


When server receives a command for a file transfer the server opens a TCP data connection to client and this is again done on a well known port 20 on the client so a new data connection is setup. After transferring one file server closes the connection. Now, what might happen is, a person may want to transfer a number of files from the server, so if again there is a request for another connection the server will open another connection again on the same port and the second TCP data connection is setup to transfer another file, so this is how it goes.

(Refer Slide Time: 07:27 - 08:41)

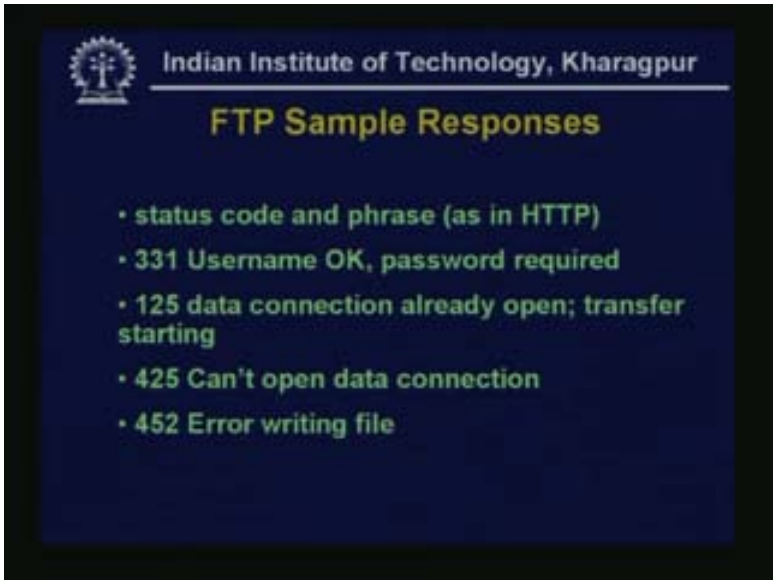
So control connection is out of band and FTP server maintains state. And what is the meaning of state here? It is the current directory and earlier authentication. That means that once a client connects to a server and gives his authorization then for that entire session till such time he does not log out that authorization is kept valid. That means he can go on accessing whatever is available on the other side, then the current directory means the directory which he is browsing at the moment, the directory from which he is downloading the present file etc. The current directory for this particular client is also maintained at the FTP server. Therefore it is called a stateful protocol in the sense that it maintains the state. Even if one file transfer is over and it closes but the authorization in the current directory remains, hence this is a stateful protocol.

(Refer Slide Time: 08:42 - 09:57)



FTP Sample Commands: These commands are sent as ASCII text over a control channel. This is a simple ASCII text going over the control channel. It expects the user's username and the password. These are commands you can give and these commands go to from the client to the server through the control channel. So, you can do a return list of file in current directory. This is the basic FTP command. You may also have the user interface which shows it very attractively in a graphical fashion. RETR is a file name, it retrieves or gets a file, you can use it to get and retrieve a filename. STOR filename stores or puts file onto the remote host. You can both get a file from the remote host as well as you can upload a file on the host and for doing that you have to give the particular filename. These are some sample commands for FTP which you can use.

(Refer Slide Time: 09:58 - 11:07)



And then once again we are talking about the control channel only. The FTP sample responses give a status code and phase as in HTTP. These are basically codes 331 user name OK, password required. When you have just entered the user name and if that user name is in his list he will sent an ok for the user name but then you still have to give the password. And 125 data connection already open transfer starting, so, in the control channel these are the kinds of responses from the server side that may come. And 425 cannot open the data connection, 452 error writing file. So, all these errors and other kinds of responses come through the control channel. These are some of the FTP sample responses. That is all about FTP as far as it is concerned.

It is a very simple protocol but at the same time it is an extremely important protocol. People always want to download files and these files could be of any kind, there is no restriction on the kind of file. If it is a binary file you have to set the mode that way. But the file could be a text file, it could be a binary file, it could be images, multimedia or anything. If it is a binary file you have to set a mode.. And the next important application we are going to discuss today is SMTP which is at the route of all the emails or the so called electronic mails.

Electronic mail has been a very important application of the network. As a matter of fact these two things taken together that one is the web which is the HTTP protocol which we are going to discuss in the next lecture. HTTP and email i.e. web and email have been the killer applications so as to say for computer networks which made computer network immensely popular, contributed a lot to the growth of a network in general. As you grow you get all the economies of scale so naturally money is invested and a lot of money comes in which sparse further growth and further innovation etc. HTTP and SMTP have been the so called killer apps of computer network. Let us now look at the details of SMTP.

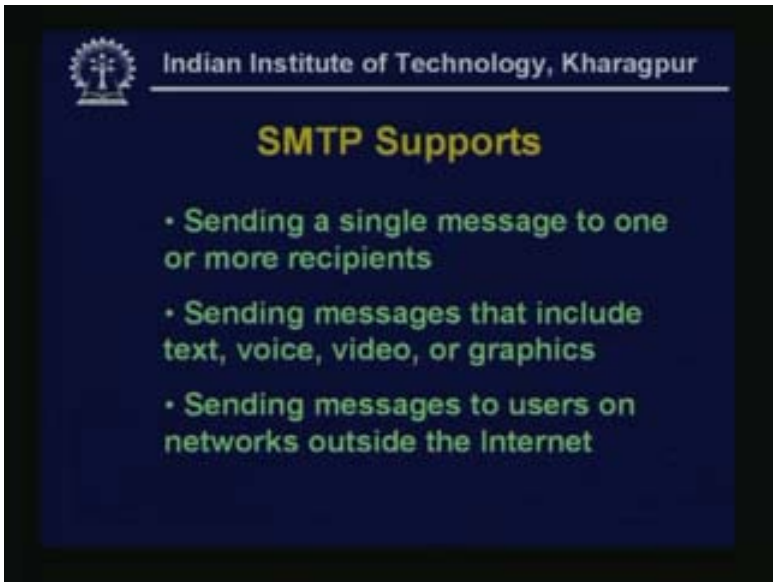
(Refer Slide Time: 13:24 - 14:17)

SMTP stands for Simple Mail Transfer Protocol. It is a simple protocol. Like FTP it is quite simple. But for sending mails etc there are some more intricacies which we will see. This is one of the most popular network services, email, is supported by TCP IP protocol SMTP. System for sending message to other computer users and provides a mail exchange between users. Actually, many of you must have used emails and you know that you can send a mail to a number of people and there are lots of advantages of this email. The chief advantage is being the speed at which your mail reaches the other end.

Actually once email came into vogue the classical mail which goes into the post office have been renamed as snail mail that it goes at a snail space whereas email goes extremely fast. It may not be instantaneous depending on how your mail servers etc are configured but it goes very fast and it may be almost instantaneous. So, in a few minutes the mail reaches the other side. And the other thing is that it is so cheap and you can attach any kind file to it. Therefore all these make the email very attractive and the volume of email has grown tremendously dwarfing the number of ordinary mails that had been going on before that by orders of magnitude. That is what SMTP does.

(Refer Slide Time: 15:15 - 15:19)

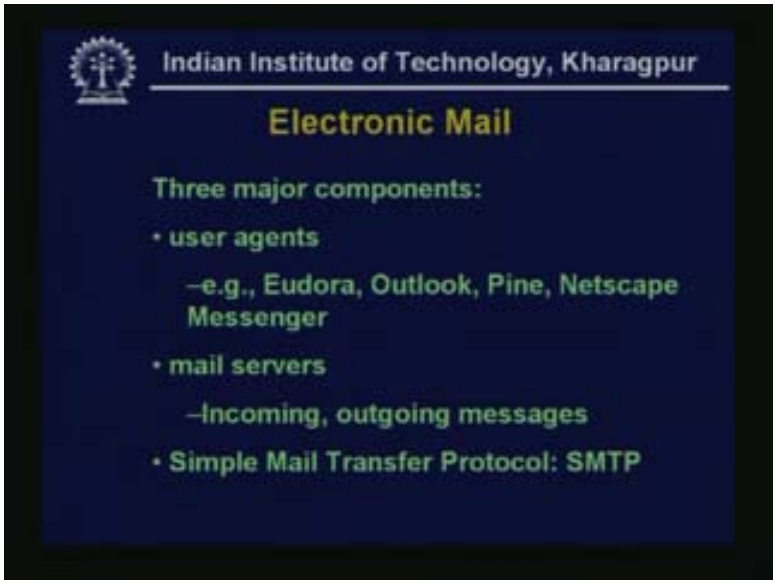
(Refer Slide Time: 15:20 - 17:08)



It supports sending a single message to one or more recipients. Sending messages include text, voice, video or graphics. Actually as the SMTP was originally envisaged it can carry only text messages but then people found a way around for encoding other kinds of a data because as soon as text messages become per say people wanted to send pictures, voice, video and all kinds of graphics etc through the email so there was a way of encoding it. So we can send messages to users on networks outside the internet which is also possible. Actually this is becoming less important by the day because what is happening is that internet is swamping all other types of network but when SMTP came into existence not only the TCP IP networks but other types of networks were also in vogue and so a way

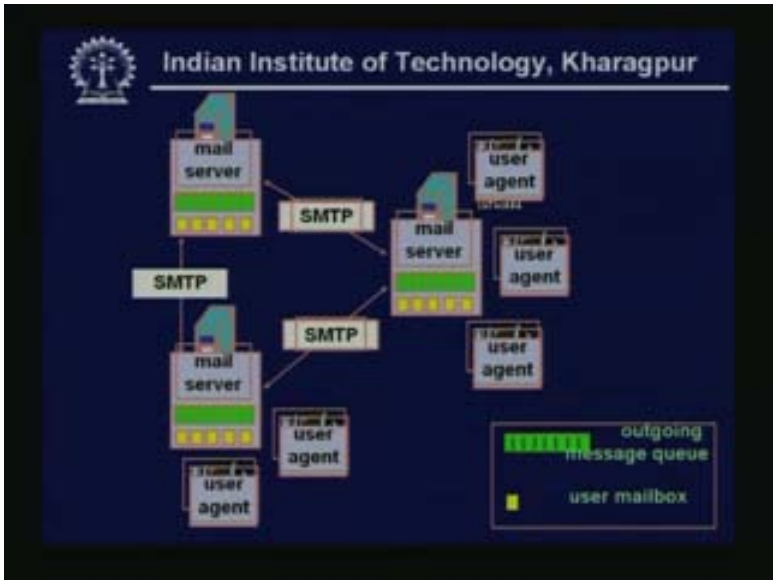
had to be found to exchange mails to users who were on different networks. It was in the heterogeneous kind of network and that has actually introduced a lot of complexity into the SMTP gateways. This is also possible although this is becoming less important now because everybody seems to almost switch over to this internet and TCP IP protocol.

(Refer Slide Time: 17:09 - 17:59)



An electronic mail system has three major components: One is the user agents. Here are some examples of user agents like Eudora, Outlook, Pine, Netscape messenger. So all kinds of user agents are there which provides the user with a graphical interface and it makes it sort of simpler even for late people to use this mail system. And then there are mail servers which handles the incoming and outgoing messages and they use the simple mail transfer protocol between the mail transfer to reach the mail from one end of the network to another.

(Refer Slide Time: 18:00 - 19:21)



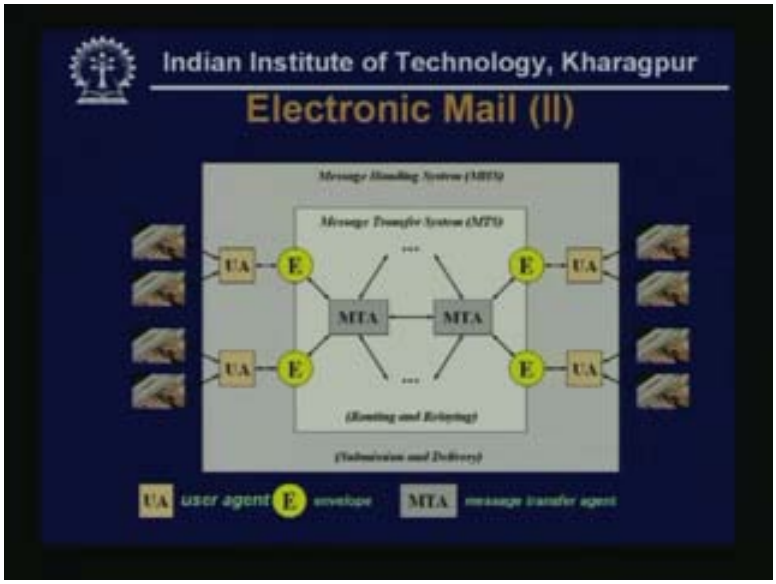
This is just a diagram. We have these user agents who are connected to mail servers. The mail server maintains an outgoing message queue so any of the user messages may keep on sending mails so they are in the outgoing message queue. Similarly, whatever incoming messages are coming they also come into this mail server and by looking at the users address they go into the individual users mail boxes. So, if these yellow boxes represent the users mail boxes then the user agent will connect to these mail boxes. Mail boxes are basically directories where all your mails are kept and it can access its own mail i.e. the mails which are incoming so they go into these mail boxes and mails which are outgoing they go into a queue and from this the mail server sends it to different other mail servers and these different mail servers talk to each other via this SMTP protocol so it may reach another mail server to the mail box to a particular user agent. That is how the system works.

(Refer Slide Time: 19:22 - 21:11)

One important component of the system is the user agent. It prepares the message and creates an envelope. This envelope is nothing but, the term obviously has been brought in from classical mail just as you write a letter and then put it in an envelope similarly you put your message into an envelope which has got a certain format. It is the job of the user agent to create the message, the user has to type in the message or upload it from somewhere and then it is put in the envelope in the proper format. This would be the job of the user agent. Then he puts the message in the envelope and sends it over to the mail transfer agent, which is one job of user agent.

The other job is that, for the incoming mails it has to show the incoming mails and then depending on what kind of user agent you have you may be able to create directories for classifying the mails. You may have mails from friends in some directory and mails for business purposes in other directory and some user agents also allow you to define rules so that whenever an incoming mail has come to the user agent it will automatically examine the mail and put it in its proper directory. All these are done by the user agent and then there is this mail transfer agent. So from the user agent the envelope goes to the mail transfer agent who transfers the mail across the internet.

(Refer Slide Time: 21:12 - 22:04)



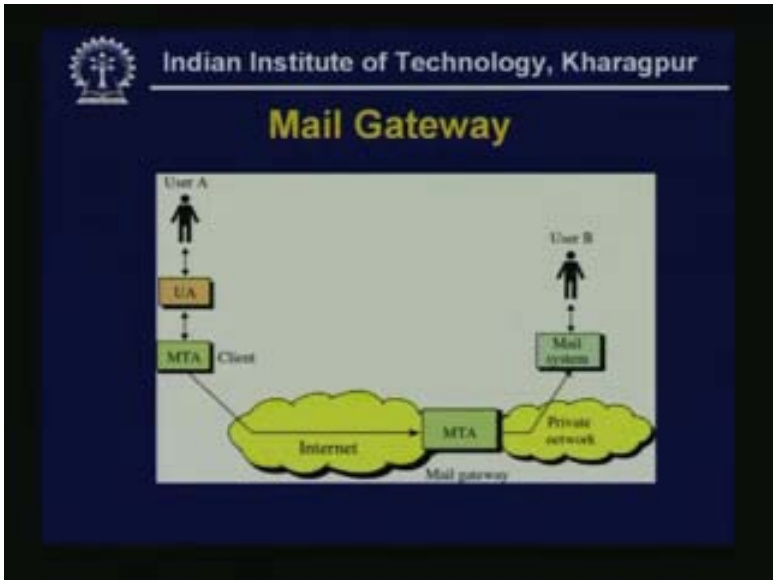
You have these different machines who uses the user agent to create the envelope which are then sent to the message transfer agent or mail transfer agent. Now it travels through the internet, for the time being let us consider only internet as a homogenous kind of a networking system and then the envelope is delivered to the proper user agent and then the user accesses his mail through the user agent into his own machine. So this is the overall email architecture. We will look at another overall architecture a little later.

(Refer Slide Time: 22:05 - 22:32)

There is some relay MTAs. So mail may be relayed through a number of MTAs, allows systems not using TCP IP to send email to users on other sites, is accomplished through a

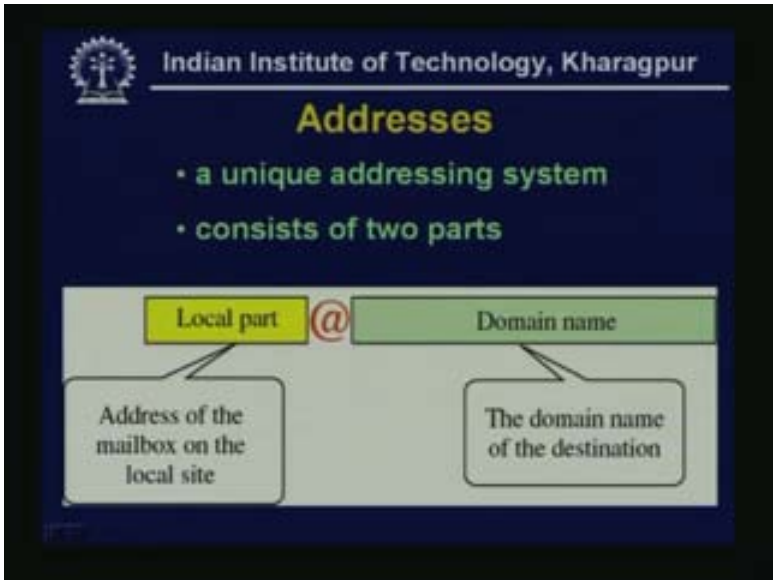
mail gateway. A relay MTA that can receive and send a mail prepared by a protocol other than SMTP. So this relay MTAs are very important. First of all, by looking at the address remotely it may not be possible to disambiguate the address of the final mail server who is going to give directly to the user agent, it may not be possible to disambiguate it. So maybe there are mail relays before that are bigger entities who are known and the mail can be sent to him and he will send it to the proper mail server who will directly give it to the user agent, that is one job of mail relay. The other job of mail relay is that, suppose it is possible to transfer mails between heterogeneous networks, so you have one kind of network here and another kind of network here, so in between there has to be a mail gateway. This mail gateway is also some kind of relay MTA so the message will first come to this relay MTA who will then actually store it and then the format etc of the envelope and other things may not be the same in the two networks. So he will make those proper changes and then send it over to the second type of network. This is the job of a mail gateway.

(Refer Slide Time: 24:08 - 24:47)



Here is an example, suppose a user agent has sent a mail through its own MTA via the internet, now this MTA is actually working as a mail gateway and this mail gateway then can push it to another kind of network which is a private network to the corresponding mail system. So this way it makes it possible for user A and user B to communicate, exchange mails although they are on different kinds of networks. This is becoming less important by the day because other kind of network is giving way to TCP IP networks.

(Refer Slide Time: 24:48 - 26:47)



Now, for delivering mails of course you need the addresses so there is a unique addressing system. It consists of two parts; one is the local part followed by ACK followed by a domain name. You must have come across mail addresses and seen ACK something dot something dot something. So this is what it looks like. Now the part before the ACK is the local part and this local part is supposed to specify the mail box within that local network where this mail is finally to be deposited. Suppose that is the address of the person to whom you are sending the mail, the destination of that particular mail box to which this mail is to be deposited is given by the local part, whereas the domain name is what allows him to send a mail to the proper destination. By looking at the domain name he knows the destination so he will send it to the corresponding mail server in that domain. Sometimes this may not be done in a single hop so it may go through a few mail relays.. So this is the address part. Once the domain name reaches its destination then by looking at the local part finding out the address of the mail box is easy.

(Refer Slide Time: 26:48 - 26:59)

The local part defines the user mailbox where mail is stored for the user and domain name is the name of the host used as the mail exchanger.

(Refer Slide Time: 27:00 - 28:28)

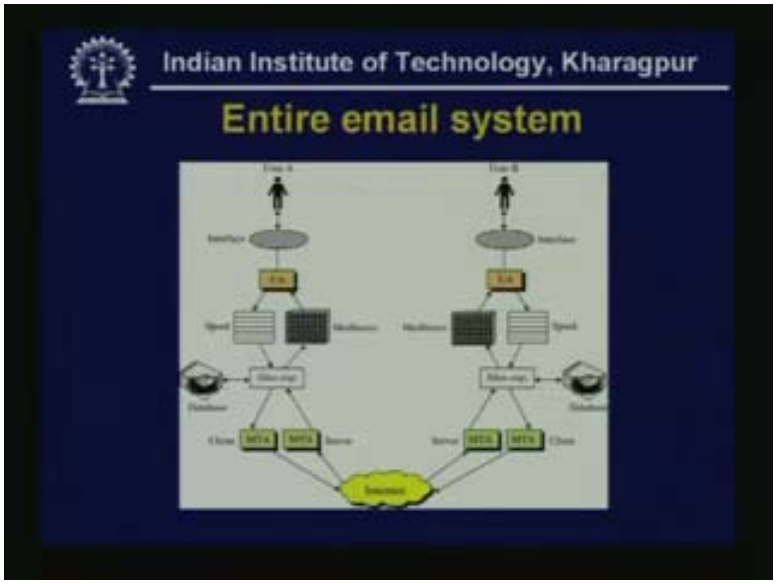
Unlike FTP or HTTP for mails messages do not necessarily have to be delivered immediately. It can be delayed at the sender site or receiver site or intermediate servers. During this time what happens to the mail is that it is waiting in some queue and the mail servers may be configured to pole the different machines connected to it for outgoing mail from time to time. That is the time when it delivers the mail so it sends and receives messages during that time and that time may be set, that time may be a few minutes or

may be more than that. And then it goes and waits in the outgoing message queue before it is processed etc. Therefore the delivery is not instantaneous it may be delayed. But usually nowadays mail is so important so people put powerful machines for doing these jobs and the mail goes really fast compared to snail mail but then compared to other protocols like FTTP or HTTP this SMTP, the messages may have to wait in some queues in some servers either on the sending site or receiving site or intermediate servers.

(Refer Slide Time: 28:29 - 29:34)

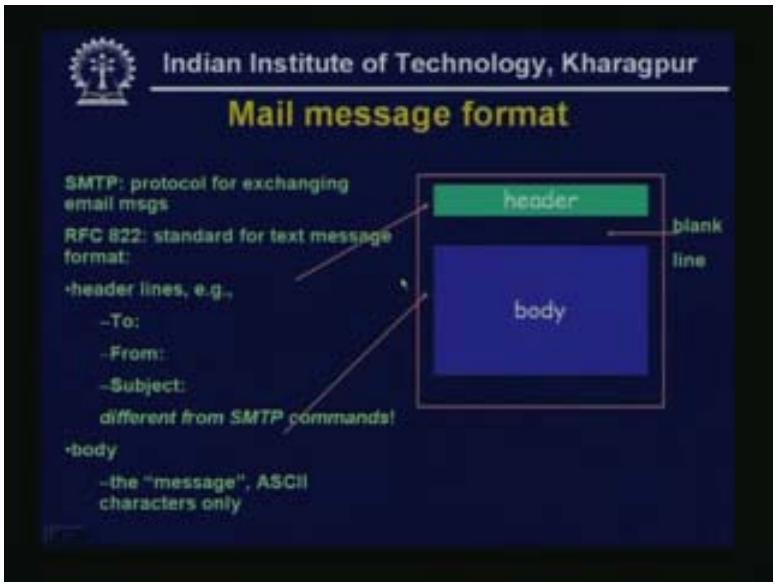
Aliases mean different names for the same person. SMTP allows one name an alias to represent several email addresses. This is one-to-many alias expansion. A single user can be defined by several email addresses so this is many-to-one email expansion, so both are possible. For example, you know that it is possible to send mails to a group of people. Suppose you regularly communicate to a group and when you send a message you want that message to reach to every member of the group, something like multicasting, so what you can do is that, you can define the group giving the email addresses of all the members in the group and then when you actually compose the mail you can simply send it to the group so the machine will automatically send one copy to each member of the group. That is one-to-many alias expansion. Similarly many-to-one is also possible.

(Refer Slide Time: 29:35 - 30:11)



This is a general kind of picture, you have the user who goes through the user interface to the user agent which puts it in this spoon for the outgoing mails and it collects the mails from the mail boxes from the corresponding mail boxes. User A will have one mail box here and then there will be alias expansion. Then it will go through the MT's through the internet to other MT's to reach its destination.

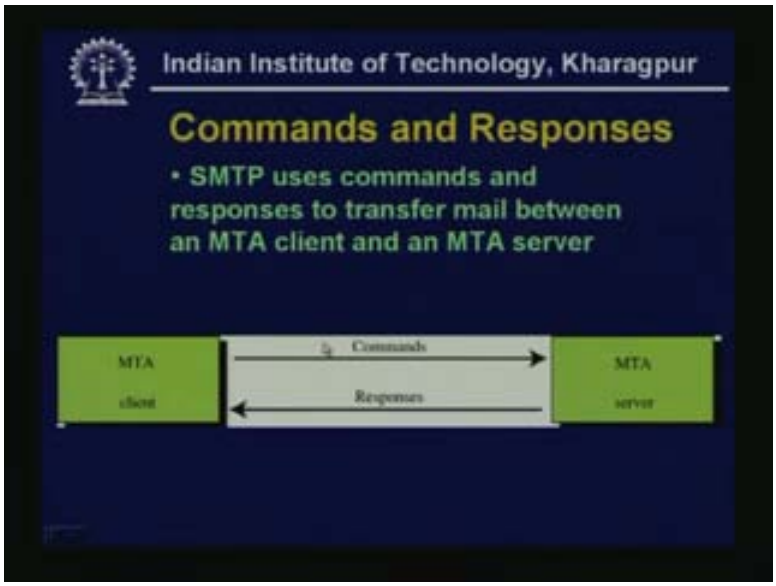
(Refer Slide Time: 30:13 – 30:57)



Now, looking at the mail message format, this is the SMTP protocol for exchanging email messages and RFC 822 gives the standard for text message format. The header lines contain two things namely to whom it is addressed from, i.e. where the mail is

coming from and then there is a subject line. So these are members of the header. Then there is one single blank line and then there is the body that is the message and the message as you know is the ASCII characters only.

(Refer Slide Time: 30:59 - 31:12)

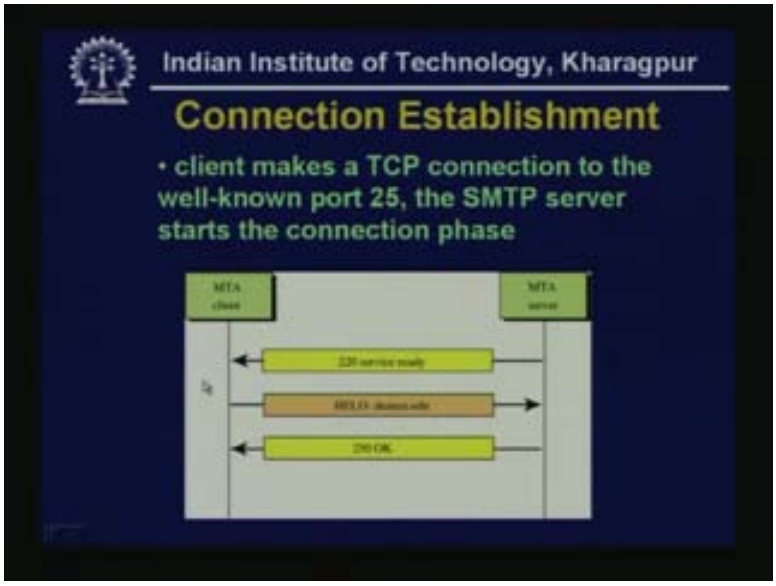


Between the MTA client and MTA servers SMTP uses commands and responses to transfer mail between an MTP client and MTP server. Therefore this is the command and response mode like other application protocols.

(Refer Slide Time: 31:14 - 31:29)

Mail transfer has the following phases: The process of transferring a mail message occurs in three phases: connection establishment, mail transfer, connection termination.

(Refer Slide Time: 31:30 - 32:12)

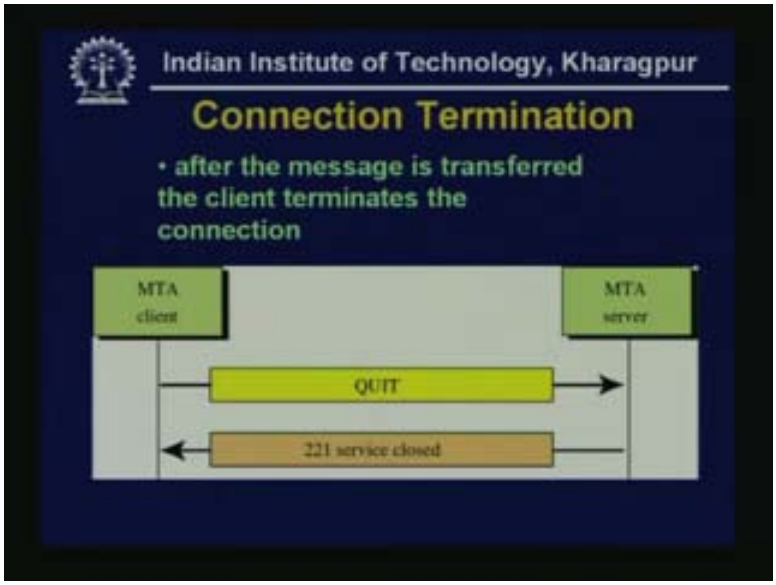


So, for connection establishment client makes a TCP connection to the well known port 25. The well known port 25 is used for SMTP. The SMTP server starts the connection phase. So the server will reply with the 220 service ready. As soon as MTA client sends a request to this port 25 if the sever is ready or 220 service ready will come back automatically then there will be a hello message from client to the server side and server side will respond with a 250 OK. These are the commands and responses which are going on.

(Refer Slide Time: 32:13 - 32:33)

We will look into more details of an actual SMTP session later. When the connection is ready a single message between the sender and one or more recipient can be exchanged, this is the message transfer phase.

(Refer Slide Time: 32:34 - 32:42)

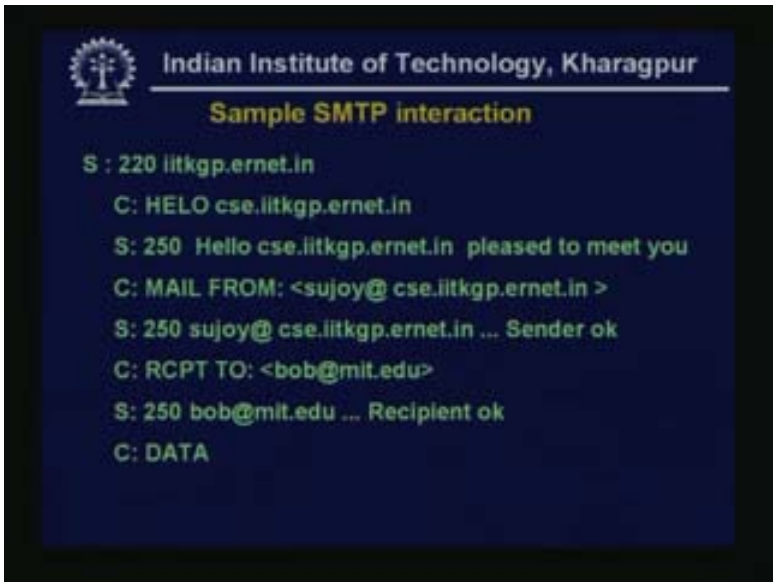


And finally there is a connection termination when the client sends the quit and the server responds saying that the 221 service is closed.

(Refer Slide Time: 3:45 - 33:17)

Actually, this protocol is so simple. You can just try it out yourself to send mails by directly doing a telnet on a particular port suppose you know the server name. So you just telnet the sever name 25 and you will get a 220 reply from server, then you enter your commands hello, mail, from, receipt, etc and then finally you quit. The above lets you send mail without using email client or a reader.

(Refer Slide Time: 33:18 - 34:50)



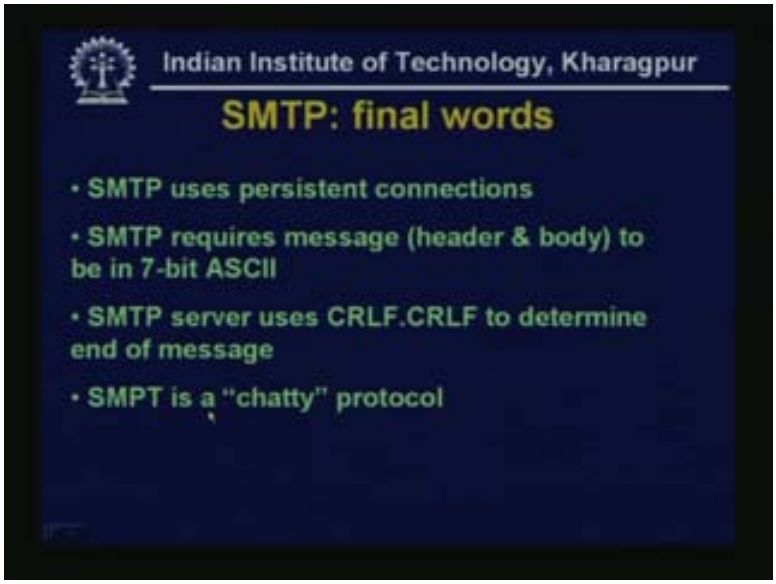
This is of course also used by people who do not want to be identified that who is sending the mail etc, so all other misapplications and misuse of these facilities are also there. Here is one sample SMTP interaction, after you have telneted it the server comes with a

response that 220 iitkgp ernet.in if suppose that is the name of the server. Then the client sends a hello cse iitkgp ernet.in, so you receive a helo from this client. Then the server will say helo cse iitkgp ernet.in pleased to meet you, then it will say mail from then maybe you will give an address, then 250 so the sender,OK. So, this is the client and server, the client is sending one command and the server is responding saying they are giving the feedback all the time that it is ok. Then you have receipt TO, this mail is to be sent to such and such, so he will say that recipient ok. Then he will say data, meaning that, what follows would be the data which is the body of the message that is to be sent.

(Refer Slide Time: 34:52 - 35:45)

In response to this data command what the server will say is, give 354 enter mail, end with dot, there is a full stop on a line by itself. So, suppose this is the body of the mail, this is a test mail checking protocol etc then you end the mail with a dot by itself. So carries return nine feeds, CRLF followed by dot followed by CRLF that is the SMTPs way of checking that the mail body is ended. So the message 250 means message accepted for delivery. Now, when the message has been accepted you can quit and then it quits.

(Refer Slide Time: 35:47 - 38:06)

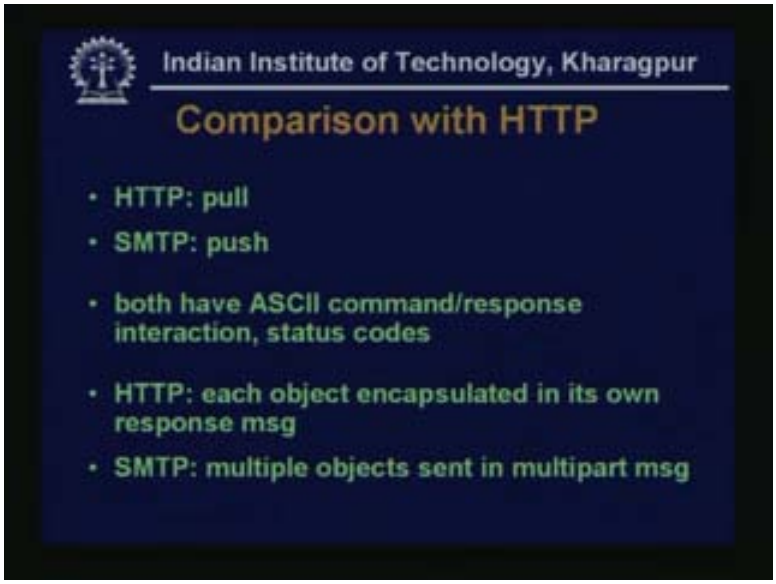


Here are some final words about SMTP. First of all it uses persistent connections. That means, so as long as you do not quit you can go on sending mails one after the other. Actually that is what the machines will do. The session I showed you is just an example, usually you would not send mail by directly using SMTP. As a matter of fact there is no point doing that unless you have some other intention. So what you will do is that you will use a standard mail agent for doing that.

When the machine opens an SMTP connection this is going to be persistent till such time you quit. So, a number of messages can be sent in the same session. I should also mention that, apart from these standard mail agents like using SMTP etc, one service which is becoming quite popular now-a-days are the web mails. Actually for the web mail you connect to the web mail server i.e. whoever gives you that service through HTTP. From that point onwards it works the same way. So SMTP uses persistent connections.

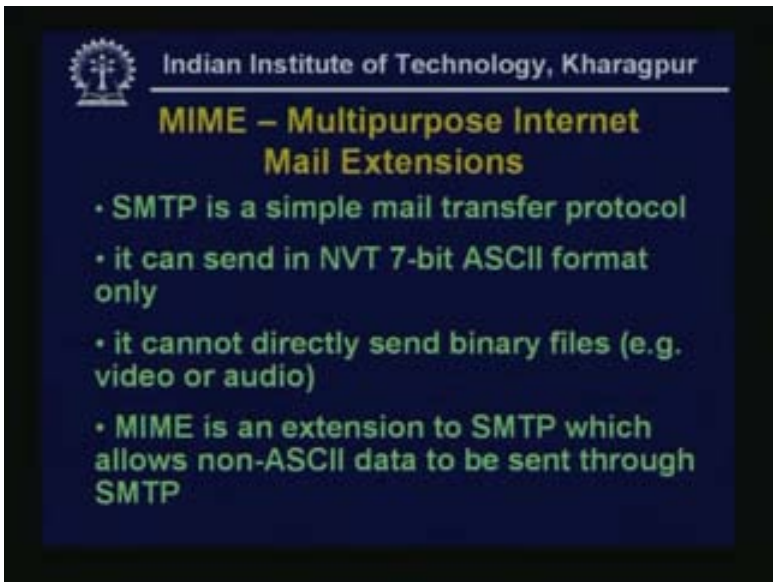
SMTP requires message that is header and body to be in 7-bit ASCII. SMTP server uses CRLF. CRLF that means carriage return line feed. carriage return line to determine end of message. And as we have seen that SMTP is a chatty protocol in the sense that you say something immediately it tells you something back so this back and forth command and then the response goes on for this CRLF. CRLF sometimes some flexibility is also given when some part of it is missing.

(Refer Slide Time: 38:08 - 38:53)



And comparing it with HTTP one thing is that HTTP is a pull kind of a protocol that means the user rather makes a request and pulls the content whereas here the content is pushed from the sender to the receiver. Both have ASCII command response interaction and status codes. In HTTP each object is encapsulated in its own response message. SMTP multiple objects sent in multipart message. Actually you can send a multipart message, this is where all the attachments etc to a mail come in.

(Refer Slide Time: 38:55 - 39:59)



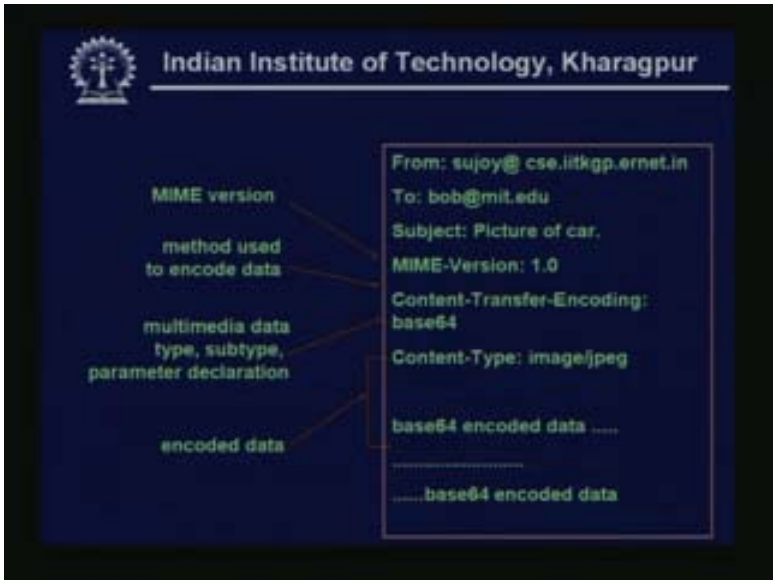
Now, of course as we know that now-a-days we can send not just text messages but other kinds of things namely pictures, graphics, voice recording etc we can send through the

mail. This means it is possible to send binary files through the mail. And the way it is done is by encoding this binary file into a textual form. There are various ways to do this encoding. SMTP is the Simple Mail Transfer Protocol, it can send in NVT 7-bit ASCII format only. It cannot directly send binary files, for example, video or audio or images.
MIME: Is a Multipurpose Internet Mail Extension. So MIME is an extension to SMTP which allows non ASCII data to be sent through SMTP. This is so important and useful and very common now-a-days.

(Refer Slide Time: 40:01 - 40:15)

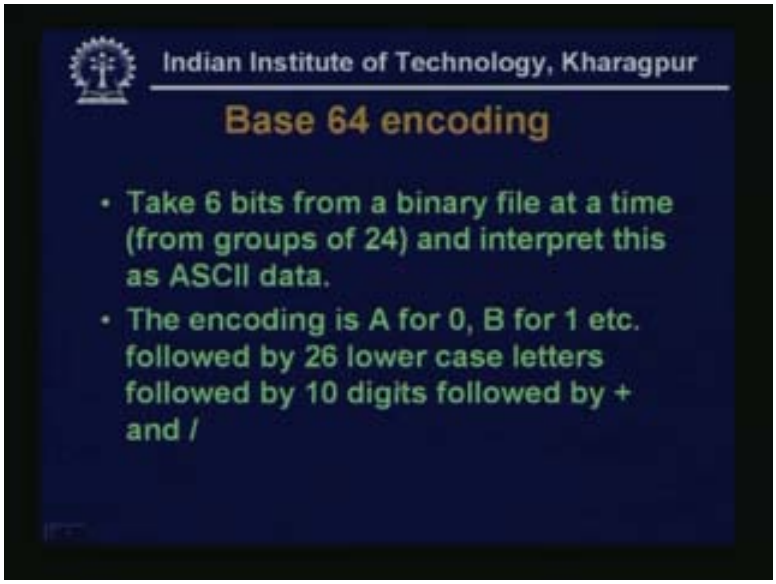
It transforms non ASCII data into NVT ASCII at the sending end which can be transformed back at the other end. What it does is that it adds additional lines in message header to declare MIME content type.

(Refer Slide Time: 40:17 - 42:05)



This is an example of some message which actually contains a binary file. The initial part of the header from sujoy etc to etc etc, subject: picture of car where maybe we are sending a picture through the message. So the from to subject is like a standard message. Then since this is using MIME it says MIME version 1.0 so MIME version has to be mentioned. This is important because whatever you are sending has to be decoded on that. Then content: The method used to encode data, content transfer encoding is base 64. So base 64 is one way of encoding a binary file into a text file but you have to mention that. There are different ways of encoding binary file into text and base 64 is just one of them. So the sender has to mention or sender has to clearly specify that this is the standard encoding scheme that he has used for encoding the file so that the receiver on the other end can actually decode it. So you have to mention the method used to encode data. Then you say the content type which is image or JPEG and then you give the base 64 encoded data.

(Refer Slide Time: 42:06 - 43:42)



Base 64 encoding is actually quite simple. Suppose you have a binary file, you take 6-bits from a binary file at a time from groups of 24. So you take 3 bytes and make four 6-bit sequences from this so 3 into 8 is equal to 24 and 4 into 6 is equal to 24 so from three bytes you can get four groups of 6. So you interpret these 6-bit binary strings as a binary number and then depending on that number you encode it into a textual form. So 6-bit will give you 2 power 6 which is 64 different numbers which is 0 to 63 and the encoding is the following: Encoding is A B for 0, 1 etc followed by 26 lower case letters so that makes 26 upper case plus 26 lower case that is 52 letters followed by ten digits (0 - 9) so that makes it 62 followed by ‘plus’ and ‘/’ for number 62 and 63. This way from (0 - 63) you get A to B upper case, a to b lower case, digits, and ‘plus’and ‘/’ that gives you an encoding so that is a text and you encode it as a text and then send it as a text.

Now, we will come to the user end so this is POP3 which is the Post Office Protocol. POP3 is a very simple protocol for interacting with your mail box which is in the server. This is very widely used. What this is used for is that, actually your incoming mails will be deposited in your mail box, now POP3 is a protocol for getting that content of your mail box or downloading it on to your desktop machine. that has advantages for example, the server gets less loaded because with people getting so many mails now-a-days, very soon if you have a departmental server giving the mail services to all the members of the department and if all the mails get accumulated there then it will load the mail server variable, it is good practice to download it to your machine and POP3 is a protocol for doing this download.

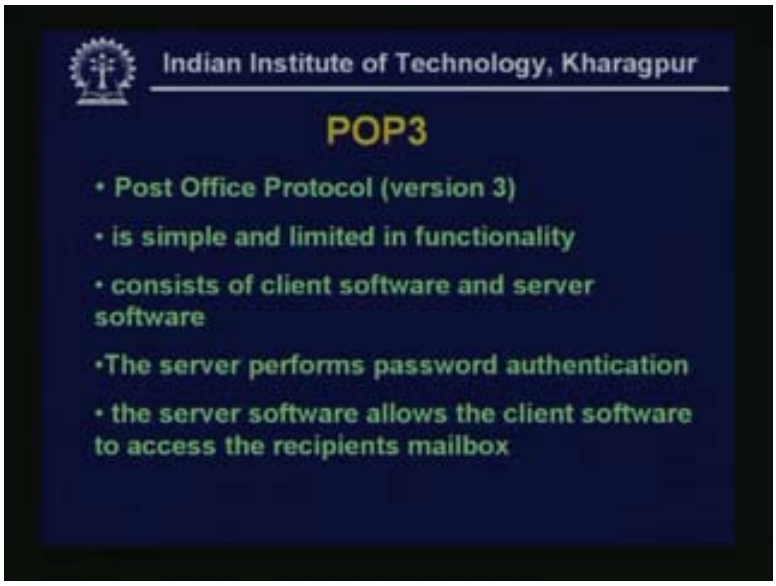
(Refer Slide Time: 45:01 – 45:42)

Mail Access Protocols: there are other protocols apart from POP3. SMTP is a push protocol. Now, if somebody has pushed a message for you how will you access that email. So, this Mail Access Protocol is retrieval from the server, it allows mail stored in mail boxes to be accessed by the recipient.

(Refer Slide Time: 45:45 – 46:36)

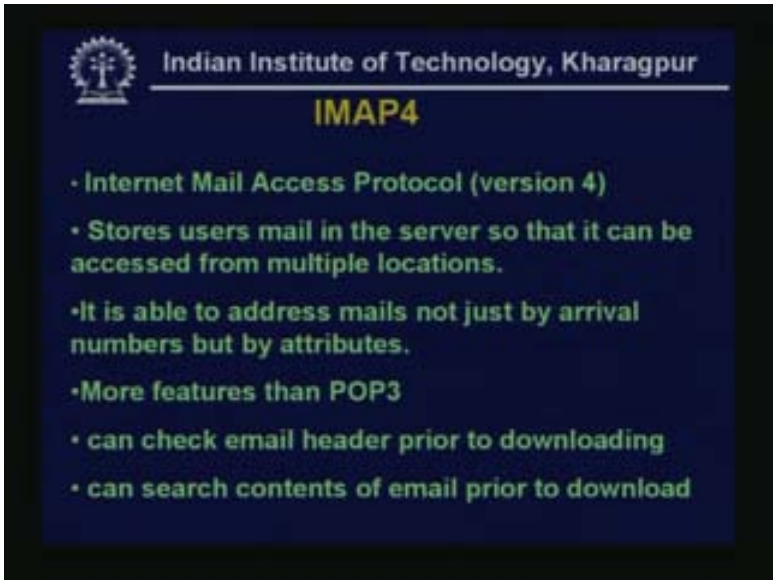
POP is the Post Office Protocol where 3 is the version 3 RFC 1939. Users can not create folders on mail server usually. POP is a protocol for doing this mail access. Another one is IMAP which is an Internet Mail Access Protocol. This has more features and you can manipulate messages on the server. There are web mail services also which uses HTTP that means it may be hot mail, yahoo mail etc., these are web mail services which are accessed through HTTP. These are three major protocols which people use for accessing mails.

(Refer Slide Time: 46:37 – 48:29)



Post Office Protocol version three is simple and limited in functionality. It consists of client software and server software. The server performs password authentication, the server software allows the client software to access the recipients mail box. So what the POP3 client will do is, it will connect to the POP3 server on the main mail server of a work group or department or organization. So, the POP3 server would do a user authentication once again through user and password as this is simply a password check. And when the user is authenticated he is given access to his particular mail box. And what POP3 would usually do is, download all the mails from the mail server to the desktop machine. For example, it is possible to configure it so that only a copy comes to your desktop and another copy remains in the mail server. But such practice leads to overloading of the mail server and very soon its buffer starts getting full, its memory space starts getting used up. To POP3 you are not supposed to do much here and simply get the machine on your desktop. You can organize your local mail folders in anyway you like by creating directories, subdirectories, etc.

(Refer Slide Time: 48:31 – 53:06)



POP3 has a problem for some users in the sense that, if all the mails are actually downloaded onto your desktop machine then when you are away from your desktop, if it is a desktop and not a laptop, so if all the mails have been downloaded on your desktop then when you are away from office you have no access to the email folders. Now-a-days, the mail has become a common mode of not only exchanging information as well as storing old information etc., so people need access to pass mails almost all the time.

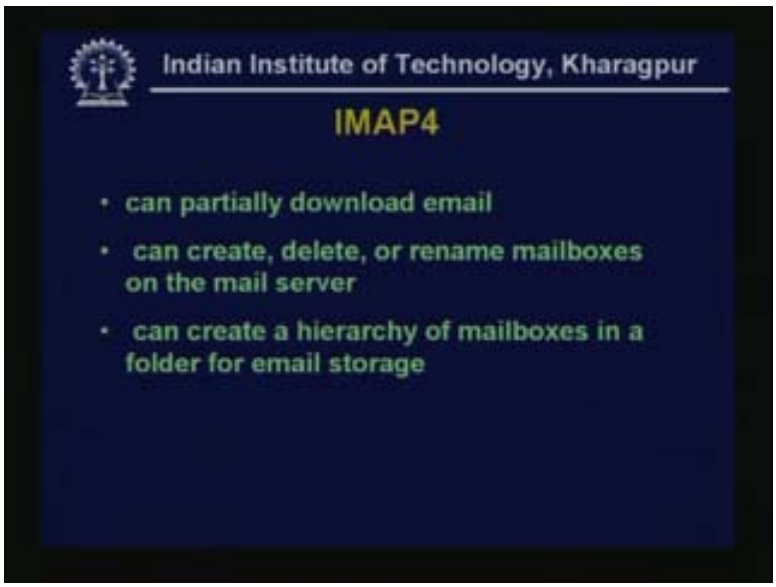
One way to do it, instead of downloading it using POP3 onto your desktop you can download it onto your laptop and carry that laptop around wherever you go. The other alternative way is to keep it in the server. If you can keep it in the server what would happen is that, naturally you can get access to the server from other places and you can see your mails. But the same advantage is there for web mail also. If you are using a web

mail service your mail is stored in the server whoever is giving you the service. Both web mail and IMAP4 has this facility of storing the mail in your server.

What facility you require? They are in the server, mail box, and anywhere. You can have access to it and if you directly access the mail box the interface is not so good so these access protocols gives you somewhat better interface. Specifically, IMAP4 is more complicated than POP3 and has got more features. IMAP4 means Internet Mail Access Protocol version 4. It stores user's mail in the server so that it can be access from multiple locations. It is able to address mails not just by arrival numbers but by attributes. So you can define attributes on mails and you can say something like give me the last mail from Steve so it is going to do a search on your directory and access that particular mail. But if you have it on your desktop you can organize it anyway you like. here of course you are not taking it to your desktop it is in the server so this protocol has got more features so that you can access your mails in a more flexible manner and that is important because now-a-days it is all the old mails you have, there are many people who have thousands of mails and many of them are useful of course it keeps accumulating junk also somehow we fail to delete. So, anyway some kind of organization and some kind of searching this is necessary in order to operate with mails in an easy manner. So it is able to address mails not just by arrival numbers but by attributes also. It has more features than POP3 as I mentioned.

Can check email header prior to downloading which is another feature. What POP3 would do is, download all mails that are in your mail box in a blind fashion. For accessing through IMAP4, before accessing you can do some kind of filtering by checking the email header. It can search contents of email prior to download so contents can also be searched before you download.

(Refer Slide Time: 53:07 – 53:26)



It can partially download email, it can create delete or rename mail boxes on the mail server and can create a hierarchy of mailboxes in a folder for email storage. This is like creating a hierarchy of directories for sorting the mail into its proper category. These are the references.

(Refer Slide Time: 53:43 – 53:56)

For example, RFC:821 for mail transfer protocol, RFC:2821 once again for SMTP, RFC:2045 for MIME and there are other RFC's associated with mail. RFC's contain all the information that is necessary for these protocols. With this we come to the end of our discussion about these two important protocols namely FTP and mails that means file download upload and mail exchange. Another protocol we have to talk about is naturally the HTTP or the web which we will take up in the next class, thank you.