Digital Switching
Prof. Y. N. Singh
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture – 32

So far what we actually had done is looked in to a VoIP system a possible design technically, I am not still gone into actual sip is tough. So, first thing we that we figure out there was a indexing server requirement. If it is a purely VoIP system where only; where all end clients are, actually on the net and they are I p capable. And. secondly then I moved on this is, actually not a proper system if you want to implement with conventional telephony will exist for time to come.

So, I introduced the concept of media gateway. So, media gateway is where if you are on the conventional side, you will see as if the system is like a, conventional telephony system or conventional exchanges are there conventional network. And if you look from the I p networks side, all conventional telephone end points will look like as if there VoIP end points. That could be 1 approach and sip itself or any other protocol h.32 can be used to control the media gateway is in that case.

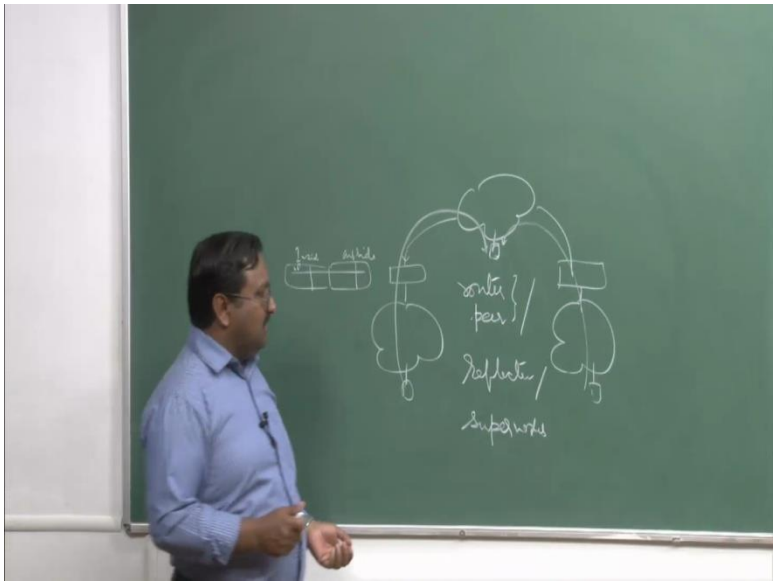
But usually this is not done, because now maintaining so many VoIP instances, in a media gateway is going to be complicated. So, a separate media gateway control protocol actually was evolved, that was the reason why Megaco actually was evolved. And of course, 1 more thing which I clarified yesterday; was that, media transport all media is encryption and whatever media will be transported or whether you are going to do the control of the media once the session is setup is not going to be handled by sip.

Usually say sip is a demarcation which was done, but some designs actually do not follow this; sip by design does not do it. But a Skype for example: call control other things are parts of the signaling itself, they want demarcate these 2 things; when we implanted Bruhaspathi Singh we also have not done this demarcation. So, we follow a different architecture because of course, ours is not a VoIP is, a not a telephony system it is lecture disable system. But sip technically can be used even for setting up of these live lecture delivery system also sip is, very generic in that sense.

So far everything is fine, but the problem is security. I have not mentioned about security and I have not also introduced I have introduced in a sense something called register or

indexing server, I have not introduce so far what we call proxies. Only reflector node I have told what a is a reflector node, reflector node is for media streaming.

(Refer Slide Time: 03:09)



So, when I said that when you are behind a fire wall, in this network you want to talk to somebody who is also behind a fire wall. So, they cannot communicate directly because, only outgoing TCP connections or UDP connections can be setup; incoming is not possible. Because, these gateways usually will never, permit incoming connections ok. Even if you send a UDP packet unless there is a table entry here mentioning that, a packet which is coming on this particular IP address.

And on this port number has to be transferred to the destination port number and IP address has to be changed to another entry. So, there is corresponding inside entry and there is an outside entry. So, inside IP address entry will be for all the IP addresses which are used inside, port number used by the actually the client; outside is the IP addresses it can be 1 or it can be multiple a bunch of them which can be allotted to the netting or outer and the port number.

So, these maps are usually unique and that is, how the translation happens here. So, incoming will be permitted if this table entry has been made. So, usually it will be done of for example, you want send an UDP outside. Then, on the reverse actually the packet end can come till the time this entry is made and policy permits it. For UDP, TCP you can only setup the connection from inside to outside, outside to inside usually it is not

permitted, because table entry itself would not be, will not be there; so you cannot setup a connection from outside to inside. So, usually something which is there on the internet will be used. So, this guy will setup a connection all the way to this node, this guy will also setup a connection to this node; remember, these will initiate and these will now maintain the entries and this will act as a router pear.

We call it a reflector, a just oppose just a terms; we call it a router pear: a router peer is, a router peer some people call it reflector, some people call it super nodes, but technically they are all same things. They are different terms actually being used because; there is no standardization of terms. Now in VoIP system this is, we actually can have something is we call it media gateway, is connecting to medias.

But most of the time this transport will be can be UDP TCP, but if you are this not an acting gateway is, a proxy router this will be STTP tunneling. So, with this STTP connection will periodically fetching the whatever is stored here and keep on through a post method, keep on pushing the information. So, that is what I mentioned, sip does not bother about it; sip assumes that 2 end points can talk directly.

If they cannot talk directly, it has to find out a intermediary. And they should be able to talk to intermediary can do this; whether they will use STTP tunneling, UDP, TCP; that is not the headache of sip headache will only give the session description. And the end points once they are connected through a signaling path, they have to figure out how they will setup this connection, whether they will use a reflector or not, whether they will use a STTP tunneling or UDP or TCP for different media streams.

Are they had to decide by negotiation between themselves; for negotiation; they can use sip. In our design we actually have an intelligent client, which figures out whether you are behind a proxy or not behind a proxy and based on that it automatically switches over to 1 of the 2 options; it is not through negotiation between end peers. So, this is slightly more conservative kind of design not very flexible; is Skype also does the very similar thing.

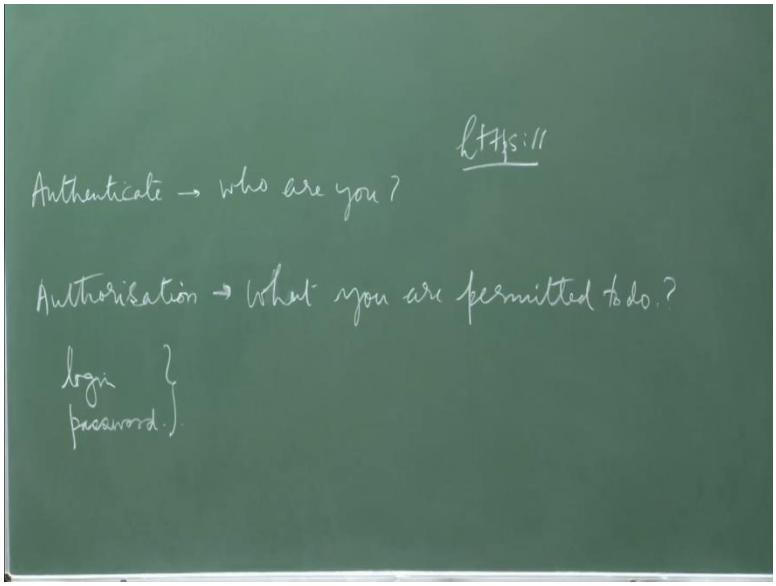
The client figures out whether it is behind a proxy or it is not behind a proxy; well it is behind a netting router or not behind a netting router and based on that actually it will figure out what will be the connection mode. So, Bsruhaspathi sink system is very similar to a Skype in that sense. The client capability actually sip does not bother about

it, sip is the end peers have to negotiate and figure out what is the best method and they will do it that way.

So, I think that will this part we had come so far. But again, a before I move over to because there is a still question. Somebody can always make fool of this system, can enter into the system how we will know that other guy with whom you want to talk, is actually that guy. Indexing server with whom you are trying to talk is, actually the indexing server. How that problem will be solved? I have not talked about this things so far.

So what I am going to do is, am going to give a generic security system. Because I think that is, an essential requirement before we move forward and, then will move over to sips. Basically sip architecture what is that. So, fundamental principle let me, actually a 1 of the simple idea of in security systems,

(Refer Slide Time: 08:24)



One thing which we need always need to do always need to do is authentication. So, you verify who is the person, who is trying to talk. So, who is the... who are you basically, that is the question which will be answered. Second thing which 1 always need to look into this thing is, authorization. What you can do? What you are permitted to do? That is the answer, which will be given by this mechanism these 2 things usually need to be answered.

In any security system, the best thing is that you always become very conservative. It do not worry about the whole system usually, I have seen this thing happening even with the lot of students when they build up their projects or they write software or they build up a design the actually assume lot of things. So, this should never ever we done. So whenever we write a distributed system, we assume you are in an NTT; which is participating that NTT will be transacting if it lot of other entities.

For every transaction it has to be ensured that, other person with whom you are transacting is the, person who actually he is telling that he is. If he is saying I am so and so a b c, he should be a b c by that is 1 thing. In secondly, you have to check with your own axis control list usually we call it ACL; your own structure, data structure or somebody if to whom you know again that also has to be authenticated from name he has to get what all permissions has been given to this person.

And based on that you will allow, but it is your discretion. So, you have to actually safeguard your own local domain, every entity has to do that in disable system design. So, if you are multiple entities, we have to now think as if you are each 1 of those entities in c. If you are satisfying this particular requirement, that every transaction which we are making with anybody you are checking who is the other person. We should never miss out even 1 single operation, we are if you are not doing that you are vulnerable.

And you should always check, whether authority has been given to him or not for doing what he is trying to do is basically is, give trying to execute some command by sending a message you are doing something and sending him information back you are responding. So, you will only respond successfully if he is authorized to execute that method. And when you are going to request, other person is going to check. So, this is if you are acting as a server, so requestors only you are verifying.

Now there is another thing not only requestors need to be verified when you make a request to somebody as a client, we have to always ensure that you are requesting the right guy. Somebody who is spoofing can actually say I am a server is sitting in and you have a faith in him you go at that particular place, give password and all the information, then he can manipulate everything.

So, even when as a client you are connecting to somebody, say indexing server you should know this is, the right indexing server. What he is what he is seeing now, how

this will be done? So, how you know that www.gmail.com is, actually the gmail's server. What you are getting is www.gmail.com is going to domain name services in turn you are getting back an IP address. How do you know this is Gmail's IP address?

You are querying only IIT Kanpur site and if some a student hacks our DNS server and gives a different IP address puts up a server in IIT Kanpur itself. So, you will get that particular servers IP address you will login and if you can create a same GOI, he will say it is a gmail; you will put login password and password will be trapped. DNS poisoning this what we call. So, DNS actually can be a 1 of the weakest links in this kind of system.

So, we have to even take care of this. So, what is the most basic system for authentication which we use, which we are aware of. Login and passwords right, shared secret we call it. In that 2 ways of handling this login and password mechanism we call it a challenge and response. So usually what happens is, most of the systems for example, Gmail; when you go there you have to first of all do a login and password you have to provide.

You try whatever it is, it will always come to this default screen and once it comes to default screen, the problem is you have to put your login password a session key will be given to you and after that session will be tracked on both sides. Hence, so far you give the right session key that session will know who you are.

So, your identification is bound to the session key. So, with the ever transaction you are not being authenticated remember; only in the first transaction you are authenticated a session key was given to you it is known as cookie actually in a web browser system. And that cookies always transacted and cookie expires a new cookie will be given. And with that you will know that who you are; for the whole session till you log out. Not necessary, I will now I am now coming to that picture.

Now, in certain web servers you have logged in for example, Face book is a very good example; Face book linked in both actually use this same thing. You have logged into the system, you do not log out; simply closed your browser. Next time you start the browser go to www.facebook.com. And interestingly you will find out that the guy is to remember who you are and he just logs in into the session; you do not have to give a login password.

How that thing happens? Because it is using what we call persistent cookies; which are there for certain time this is a random in string. So, when you do a login and password, login that time that cookie random string is generated by the server given back to you and this is stored in your browser. So, browser if you look carefully cookies is stored for cookie is a random string for a certain sight.

So, whenever in a STTP message is sent, STTP request is sent to that server. This cookie will be going as one of the fields in the STTP request. So, whenever you try www.facebook.com your browser by default has a cookie for Facebook, it will send that cookie to that Facebook thing and that Facebook guys looks at there is a cookie which has come along with the request. So, must be session must be on. So, once a session is on, it will search this cookie was sent to whom.

So, it will then figure out from the data base cookie was sent to you. And most likely, the same cookie is only assigned that cookie is only assigned to you is a randomly generated the string remember. If it is by chance if 2 persons you might end up in login is some somebody else actually that is also possible, but that chance is extremely rare, because cookie is pretty long.

So, if you look at cookie at any point of time there, there is a large number of characters and each character requires 8 bits. So, is the actually is a very, very large number, much larger than the total population on the earth or total number of logins in the Facebook system. So, it is pretty must safe system and of course, once it... and some once in a while you find systems where even if you login, suddenly in between they will ask for verification.

Because actually there was no login it was using a persistent cookie. So, they will do a periodic check, Gmail usually does this? Once in a while suddenly you will say blank screen, you have to login to further continue; yahoo also do the same thing. Now banking system do not use this kind of mechanism; banking system you should try to go to another back screen or try to do you can never do this.

They are doing easily smarter thing, now every STTP request will send a cookie, when the response will come a new cookie will come which will get stored. And that cookie has to be used for the next response. So, for every transaction cookie keeps on changing;

the moment you try to use a back button or something, which is going to an older page for that older page it will not permit.

Because it actually remembers that when you do the back, the older request is going; which is having a different cookie, older request command is send actually. So, unless the new cookie goes it would not accept. So, you can only do the forward transaction, you cannot go back actually. For every request there is separate cookie, new cookie get's generated. And in with the every response a new cookie comes and replaces.

So, he start a persistent cookies is a non persistent. So, if you log out and login, the older cookie will not work; it is only works for certain time. So, if you are not going to send another STTP request within certain time, they will time out and remove the cookie; that is still better security system.

But they are as if now what I am telling is, the server is authenticating you, but how you are figuring out that gmail is actually gmail is not been created by as if fishing site or something by the server. Now that is we are the concept of security certificates will come. Because I think this is extremely important, most of the peer to peer system will ultimately or actually, the ultimately using sip going to use security certificates for identification.

There has to be 2 way, your machine actually is doing 2 way authentication when you are login into gmail. And how it is done for gmail? You have always using something called as STTP, you are not using HTTP you are always using STTP; s if you carefully observe for gmail session. Facebook session you should be using STTPS. If you are not using, you can be vulnerable to facing actually in that case.

Facing is different in that sense, fishing is sending a URL in the email which hidden as an link behind certain text. So, time if you know where it is being redirecting once you click on there, you can figure out this URL is not matching. And never ensure that you never try HTTP. So, whenever you doing banking or anything which is require security, you need to authenticate a server make sure it is always STTPS; STTPS does 2 things.

You can authenticate a server and you can also create a encrypted channel from your browser to the server. Because that is also 1 of our requirement I have to authenticate user, I have to authenticate servers, servers have to authenticate another servers and I

have to also always keep the channel secure. Remember how the 2 clients are going to get connected.

The indexing server will tell this guy, the other guy is on this port and this IP address you will tell him this guy is on port and IP address, then they will make a connection; when this information is transacted, only indexing server is knowing. And you have faith on indexing server if indexing server is spoof, what it can do is: it can now use a media gateway in between route, this call to this and this guy route call here and it can do the tapping of the call.

So all the snooping will be done can be done in that way. So, if you can and this can be done if you temper with DNS. So, DNS is most... So do not believe on DNS; DNS only gives from a name and IP address. Once an IP address comes, then you have to further verify that guy. Whether, that server is authentic or not authentic. So, how that will be done. You can log give a login and password, but how that guy will give a login and password to you, gmail cannot login into you. And you cannot... so gmail has to remember all login passwords for so many users and provide them, a credentials no it is not going to be done that way.

So, we use something called certificate. So, you have all of you have actually Id cards coming to this concept. These Id cards are usually signed by dean of student affairs and a dean of students affairs Id cards by director. Director's Id cards are signed by somebody I think chairman of the board, his Id card must have been signed by the somebody in the ministry and his Id card ultimately it goes to the president of India; who is the first citizen and we assume that everybody knows, how this signature of president of India looks like.

Since the well known signature, well known public key or we call it. So, it is a extremely important. Similarly it is like bank notes, bank note is nothing, but a piece of paper or a technically speaking; what is important is there is a promissory note written on that and there is signature by RBI governor. So, I can also prepare a note and I can sign with my signatures, you would not accept it. It is only RBI governors, because essentially he also has been given an authority through a certification chain. So, this is what we call concept of certification chain and there is always going to be somebody who is a master, on whom everybody is going to have the faith. So, this is what we call certification

authorities is being created from root onward, till the user's certificate; which is going to be used for authentication for puzzles both ways.

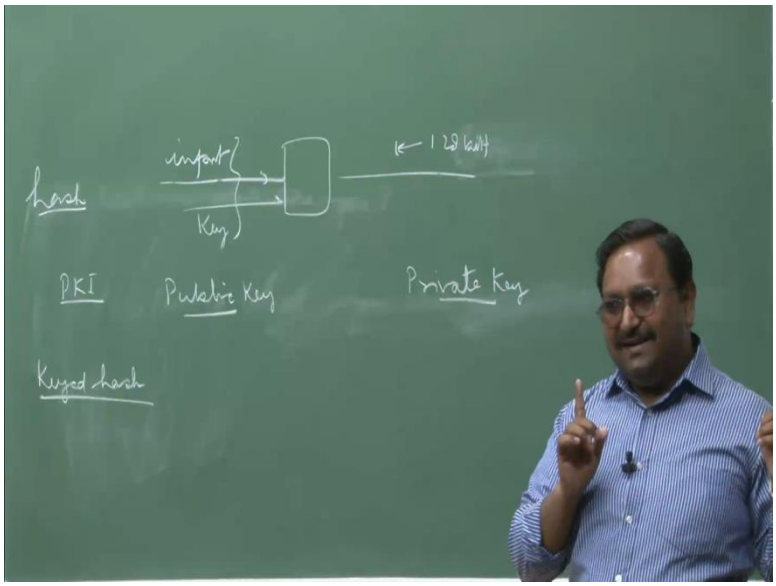
Now when do you talk to another guy which is I am also from IIT Kanpur, you also say I am from IIT Kanpur how you verify? Both of we exchange, show yours Id cards. So, you know exactly his name is x y z and that is written on the Id card, your name is a b c because that is written on the Id card. So you cannot fool each other, verify each other's Id card both of them signature of dean of student affairs perfectly fine.

Both of if you are aware of the signature of that is also, you verify. Unfortunately he does not have a photocopy or a copy of dorsa certificate. So, if dorsa changes you do not know if you do not know the dorsa signatures, you are doomed. But somehow you can go to website and verify signature and so on. If the verification can go till the route, it will be fine.

Everybody need to know only the root authority signature. So, if this is the essentially trick which will use. So, router authorities for security certificates are existing; 1 of them is actually very sign then many of them are there, version is the most popular. So, lots of agencies become certification authorities. And you can apply for a certificate and you will get a certificate, but what is a certificate now?

So, we use something called PKI; public key cryptography infrastructure actually in this case. There is only 1 problem here the certificates when they are revoked and if you cannot verify the revocation list, then there is an issue. We will also face the same problem, actually here nobody has a solution, but we assume that most likely problems will be very, very rare. And whenever you have a chance of revocation make sure, the certificate validity period has to be small. So, there is a very small vulnerability period which will be there. Only very few people will be vulnerable in that case.

(Refer Slide Time: 25:01)



So, in PKI; public key cryptography infrastructure we call it. We can always generate 2 keys I a m not going to going to details of how this is done, but this is technically possible. So, there is key pair: public and private key. So, there is method by which this can be done. The important thing is, I can generate something known as keyed hash. So, before this what is a hash? Hash is you take in whatever is the box which is this the hashing function, you push in whatever is the input it can consist of any number of a strings it is a material. And there is a attractive procedure by which you can compute and you will end up in getting a say 128 bit hash. So, 128 bit code you will get after this.

So, all possible message is which are there or sequence of strings can be mapped on to 2 rise to power 128 possibilities, if it is a 128 bit hash. So, many messages will map onto the same hash, but I need to have only 128bit that is important; message size is size is independent. Now 1 of the important thing there is something called key dash, I can input something.

And I can also input a key and I can generate a keyed hash. Even if the message is same if I keep on changing my keys, my keyed hash will also be different. So, very simple thing if you remember certain key I also know that key we share certain key and you send a message to me, message is not encrypted. But you attach the key generate keyed hash, you send me the message, you send me the keyed hash anybody can know it see no issues.

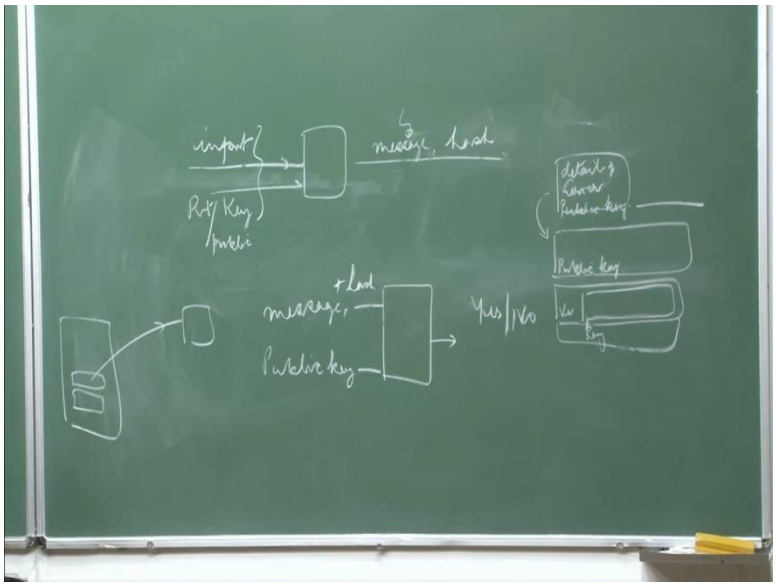
Once it comes to me, I also know the same shared key. So, I can put that key along with the message, generate the hash this hash will be the same; which you have generated in this and get to me. Somebody tempers the message in between; message now contains message plus the hash. If hash is modified it will not match, I will discard the message it has been tempered; message is tempered, then also hash will not match.

He cannot temper with the message cannot generate a new hash, because no only 2 persons are knowing the key. So, this is going to ensure the integrity between the 2 people who share a secret. I am only talking about, a shared secret common key; I am not come to this public and private thing.

My question is how we exchange the keys? That is not required; I am coming to that situation. I can verify message you are send by you, I have not still come to public and private key business. So, this is what we call integrity check and if you want to actually do encryption, you can actually you also use this key to do this encryption. A methods available, so you will get a encrypted message and on encrypted message or encrypted message; you also generate a hash.

So that hash an encrypted message also can be sent you can verify because, you know the key. No tempering has been done first thing, you have to always even for encrypted message. If you are sending an encrypted message, for me and there is no hash. For me to damage the system is very simple I can change some characters will decrypt you will get a wrong message; you have to always check also the integrity, before you do decryption. Encryption only hides a message, encryption does not check integrity. So, integrity check is separate and security separate the 2 separate things.

(Refer Slide Time: 29:29)



So, this temper proofing now if I these 2 systems, I can always use a private key here. I will get a message plus keyed hash this can be transported I can again push in whatever is this message, which has been received. And I can put now the public key; only thing which I can verify, I cannot generate the hash. Because if I can generate the hash, very well I can modify the message generate a new hash instant actually.

Hash can only be generated by private key or the other way around actually; at the this we private key or if it is it can be public key. So, if I am using a private key for generating the hash, I can only verify the integrity with this I cannot generate hash. So, integrity check will be done here that ensures, if public key can be used to verify that the proper hash; hash is actually proper.

Whatever is the current existing hash is coming is proper or not proper only that thing can come. I cannot generate with this actually the same hash otherwise, technically there is no security.

(( ))

When this message plus hash these 2 things are going to pumped in. I put it public key I can get an answer yes or no. Whether the match happens or does not happen. So, what the yes and no if it is yes, it is says whatever was the corresponding private key indeed that was used to generate the hash for this message plus hash combination.

Sir, this has been encrypted with the private key?

No, it is not done it is available it is open. But if you for example, temper with the message; you do tempering with the message in between. Though incorrect message plus hash when you will generate, I know what is the hash bits. I can this whenever I will use public key, it will give an answer no. I cannot generate hash with public key if it is already generated by private key; I only can get the answer yes and no.

How we are getting yes and no?

The way you do CRC.

What is hash?

Hash is a mapping is a non-linear mapping. Many to 1 map, many to fixed number of mapping for example...

(( ))

How does hash is different from signature; where in signature we also do the signing when the private key encrypted by the public key.

Hash is signature only, hash is signature only.

Yeah, right.

(( ))

Right, right, right.

Public private actually hash is keyed hash are slightly different because of this. So, you cannot generate keyed hash if you generate with public key, you have to verify with this private. If you generate with private, you have to verify with public.

(( ))

There is a pair. This actually more computationally complex, that is only the problem.

We are encrypting the public key?

Whenever I am sending a message to you and I want only you to understand the message. I am not talking about, encryption as of now. Encryption will never be done with a private key I have to tell you. 1 very important thing if I start doing a encryption of long message with the private public key pair, it will be computationally heavily inefficient it is never done.

So what is the way it is done is, I will send you a symmetric key; which is used for encryption of a proper of the message. Key is not known to you, for that a small key only I will use your public key for doing the encryption. Nobody else knows your private key. So, other people cannot decide for what was the key transferred to you. With your private key you will not decipher first of all decipher the key.

Then with that key many message will be decrypted. So, message can be securely send to you, but you have not try figured out that whether the guy who has send you the message is authentic or not, you have not verified that. Only thing I have ensured it as only the right guy can actually get the message out. If you also have to identify whether I am the right person who have send the message, that is another operation which is need to be done on this.

So, I have to sign this whole thing with my private key, you will know my public key. And with that you will verify whether it is the I actually who has signed at thing or not, but you also are having a faith. Because whatever, I because my public key is do not do everybody. But private key is not own, but how do you know that my public key is correct?

So, I am now am into that certificate thing what technically certificate is. So, this is what we call basically this mechanism will be used for encryption, this will be used for authentication both ways. Now what is certificate? Certificate usually will contain what we call details of say server for example, it can be anything. So, but mostly when you go for www.gmail.com certificate says, this certificate is for www.gmail.com this is having this particular IP address.

And this certificate will also contain a public key of this server, private keys only known to this server. Public key of this server, then it will also contain that it is being signed by VeriSign; VeriSign's public key must be there with you ok. It must be there with you or

if you have only VeriSign, but this guy has bought something from MTNL for example, or BSNL.

It has bought the certificate it will say this is thing signed by a certificate of BSNL; BSNL public key is this and BSNL certificate has been signed by VeriSign. VeriSign signature will be there and VeriSign public key can also be there. And I can always verify that this thing will be a stored certificate in your browser, it always updates periodically.

In windows Microsoft, windows it is update whenever the windows update will come that time these set difficulties will route are difficulties we call them; they will get updated. Whenever you will update your Firefox that time, Firefox internal certificates get updated. You can even install your own certificates; say if you have faith you accept it. For example if you do STTPS for Brihaspathi sink of Brihaspathi.

It gives you a certificate that is, certificate is not signed by VeriSign, your browser will immediately given warning. This certificate is not signed by VeriSign, this is a only self signed certificate would you like to accept it. Do you have faith on the server on which are connecting. So, in that case usually my purpose is not for server authentication or this thing because, I am not maintaining those kind of critical informs is not a financial transaction for that matter.

You are only look at course material, but you want your password should not be seen by some intermediate guy who is sniffing on the channel. So, that certificate once it comes. So, public key of the server is available, private key is with the server; which I have actually installed in a file and that file is only read only for that user it is not visible to anybody else I have hidden that actually there. So, these 2 things public private key will be used to create a shared secret between these 2.

So, what happens is they actually exchange some random strings and these random strings are past I think an encrypted as well as encrypted both ways. And both of them will, then compute using these keys, shared secrets which have been transacted. Common shared secret, actual shared secret is never transacted over the thing they compute it. And both of them, will in turn will compute the same key and that is what will be used to create a secure channel.

So, nobody can temper, nobody can estimate what is being transacted. So, before you do login and password currently if you do not use STTPS, you login password can be seen by anybody any router or any switch. If you use a STTPS, then it cannot be and it is STTPS. If that is kind of mistake they do not want then those people would not do. I am sure then that STTPS, so I have been using that.

I think most of the servers are STTPS, now this is the kind of certificate which will look like. So, you have already logged into something you know the, what is the destination IP address from source IP address, from which it is coming this is also URL; which was used and this guy will also you will can send him a random string and ask him, you kindly sign it and send it back to me, I will verify with this thing.

So, that is possible or what you do is you say ask him to send me a random string as well as signed version of it. So, he sends a unencrypted and the hash of that. These 2 can only be generated by that person nobody else can generate, third party cannot generate. Because this will not match with this public key and that way you know this gmail; actually is gmail.com.it is not somebody else.

So, that is a very simple check which can be done and that is how you will authenticate the server? And it is always you will fulfill the chain till you find out that ultimately somebody has signed this certificate chain, whose public key is there in your key store you know certificate is stored. You can actually see all these tokens these are known as built in tokens, security tokens in the browser. And check these need not be a browser, this can be even you are software which can do this job.

So, Brihaspathi sink client actually does that it is a java client written in java, which all is a actually also does the same thing; it checks for the server. So, we do check the authenticity and is especially we have been actually using a trick that is not that anybody can install indexing server and they will you can start using it you cannot, because we always run a master, where all indexing servers in the world has to be listed.

Otherwise, you cannot join a Brihaspathi sink clients session actually it is not possible. So, we have done the system in that way. So, I think this is what will be the basic frame work which will be used. Now there is a another important we talking about peer to peer systems. So, 1 way as I told I will now use this security certificate mechanism for authentication with clients.

So, Skype I think technically as we are doing because of the way it behaves that gives a gas most likely, it is using a very similar system. First time when he will start the client, you will always get a login in password. If you actually have a certificate issued to you by a MTNL, where the certification cam be verified till the root certifying agency. Then it is fine you can install guy, other guy can verify from a certificate; your certificate can be verified through a certification chain. And he knows you are a b c, other guy also does the same thing and you know he is also x y z.

But that actually means, every user in this world require a certificate and not only require a certificate you also has to remember a long private key. So, remembering password itself is difficult if is more than 15 or 16 characters. So, you have to have some sentences you will take first and last, but all combinations. And you remember in that way, by using everybody has it is own algorithm of remembering the passwords. So, 128 bit hash key if you are going to remember is going to be help for every persons.

So, usually you get a USB stick or something as you have we are the private key is stored if that is lost you are gone. And then what you will do, if it is lost because, now private key is has been compromised. So, we create what we call CRV. Certificate Revocation List; CRL we call it. Now this is maintained at the certification authority whichever is there.

So, whenever this certificate we present it your browser if it is connected directly to the net, will actually try to connect to this server which is signed this find out the CRL there. In that it will keep on doing it till it goes to the route and a route will maintain a CRL. It will verify that none of the certificate should be there in that Certification Revocation List, certificate expiry will be heart of the design. And since, it is already signed by certification authority it cannot be tempered with that actually. And usually the procedure is the when you want to generate a certificate; you can do it on your machine actually.

So, you can put all the entries and everything and there is a open SSL utility which comes, it is available on all next boxes. So, you can generate your own certificate and you will generate your own private and public key, private key will be there in your file. Now this certificate you can submit to the site, you have to also submit all your documentation they will verify these addresses and everything all things are authentic.

And, then he will sign with his private key and signed certificate copy he will send; which then you can use your own certificate also, you will be signing with your own private key also. To ensure even, so the others guy cannot temperate and this whole package itself will be signed, by somebody else. And he will attach his own certificate and sign this is a way everybody keeps on doing it; you have the complete chain because of that.

It is your choice people can have is actually forms few days to 12 years 13 years. So, that depends if you know that there is a possibility of tempering with certificates, certainly you will actually keep a period to very, very small. So, if you keep it 1 day suppose even if you lose the key, we will generate another certificate next day. So, only fed that period say you lost it in the say in the afternoon. So, midnight anyway it will expire.

So, 12 hours is your vulnerability period. In fact, there has been a case once where, a routes certificate I think of in Holland was compromised and that was I think 1 of the biggest disasters, which happened. That was the only incidence, which I am aware of where the routes certificate was compromised; by hackers. And that is it if that can be done if you know the private key, it is ok, then you can be very nasty. They can be the private key is there in side, once the private key is lost it is gone.

If you are not remembering it you have not written on a sheet of paper it is gone. Best is you remember in your mind, unless you use your memory it will be there. And nobody can read your mind, till that time is fine. So, CRL is 1 of the key components so in fact, some times what happens if you lose internet connectivity, you are using a certificate, it will actually give you a warning; if a browser has been configured properly, that I am not able to verify CRL.

So, certificate which we are using may not be correct, especially if you are reading some document and that requires a certificate to be used to match the take care of the integrity of the document. For integrity verification typically this problem comes, when you take the Ethernet port out if is not on the net. It is always dependent on the network for doing CRL check. Now, this is I think is the only problem and we do not know the solution of this.

So, usually whenever you will buy certificate, you have to pay I think 2000 something rupees in India. And any 1 of us can buy certificate we have to submit our pan card copy,

identification document, everything and then they will give you the certificate. In fact, if you are going to run a company; it will be mandatory for you. Because, your income tax returns you cannot file unless you have that certificate with you.

So, it has not been made mandatory for the employees as a as of now, but may be in few years down the line it will be made. So, we have to also buy certificates. So, now this is a problem, so everybody will not have certificate. So, how this Skype works now question is, this there is to be some internal solution. So, you have at 1 end login and password, other end you have certificates; you can make some kind of a hybrid combination.

So, what is Skype does whenever you install a Skype, Skype has it is own built in route authority. It is not VeriSign, it has its own built in authority comes program and hope fully this will never be compromised. If that gets compromised then of course, it will be big trouble for Microsoft as of now earlier it was Skype. So, the way it is done is whenever you will login for the first time, you do not have a certificate installed. Certificate remains you till logout from the system, once you logout one, once you be technically exit actually and terminate or else your session expires.

So, you will actually do a login this will go to a server, this is a hard coded server; again inside this Skype client. In fact, there is a list of servers which is maintained indexing servers; you go there you do login password it is maintains a secure connection STTPS or connection most likely. And login password is sent you are verified from the database, whenever you will change a password it should actually communicate back to the server.

So, once you do this login this will generate a certificate for you, whose private key will be stored in your client certificate will be present it and this certificate will be self signed here itself and it will also be signed by the private key of the Skype; public key of a Skype is already available to all clients. So, once it is done your certificate and gets installed there.

Now when to be Skype clients want to talk to each other, they can authenticate very easily. In fact, every device, every Skype client or a Skype super node or super peers we call it actually have a certificate. So, I also hack the code of the Skype put everything and then I start running it and I feel that this is Skype will contact me, this Skype will contact me know if this would not happen. They will not recognize me actually, because I do not have a certificate installed; which is signed by a Skype's central server.

But the beautiful thing is if this server is down I am not on the internet, I am on LAN I can a still connect if somehow I can find out the other Skype here. I can authenticate with each other using certificates. Now this is also the genesis of Brihaspathi phone design, which is peer to peer server less LMS system. So, is similar structure also actually we have conveyed there was no other option for me.

I tried all kind of possibilities if they exist, I figure out login passwords are not going to be possible. If a large number of users all across the world maintaining a server and authenticating everybody at every time, I did to maintain large infrastructure in a peer to peer system. And that will not be financially viable, but this makes a things financially viable; only thing is that I show a very short thing.

So, periodically depending on how many users are active and how fast I can, how much load a indexing server can take. We ensure that time out or the expiry period of the certificate is updated and when certificate expires you have to login again. And get a new certificate a Skype actually does this. Once in a while is Skype will logout and then say ask you to login again; this behavior I have observed actually I do not know whether you have observed or not.

But this is the most likely reason for this thing. So, now when we will build up a system, most likely this is what is a technology which we will be using… So, advantage users need not buy a security certificate from a certification agency; by pass certification agency all together. So, this is like everybody knows that the director here you need not go to the president of India and issue certificate Id cards and we all recognize there through those I d cards, is a local authentication, group authentication inside system ok.

So, I think now we understand that the servers have the server can be authenticate it and how the users can be authenticate. Authorization issue yes, that is 1 more authorization in case of sip systems usually done through sip private staff. Because is usually the service provider, which will primitive and you have to pay to for those services. First usually it will be all kind of services it is a peer to peer.

So, whatever the 2 guys which agree on this kind of service, which they want they can actually a connect with that kind of service. And of course, you should also think slightly bigger it is not only voice calls I can ask for a another peer it is actually technically sip is

creation of sessions between peers. So, I can ask for even for a virtual machine being provided by the other peer to me.

So, I can give the compute jobs, he does the computation returns back the result to me. I can make multiple connections or multiple sessions and as a ask virtual machines from lot of people. So, computing itself can be sold; I can do transactions with other peer. It is like it is a market place, you go and you want buy something a service.

So, you can buy a service from the way you buy why buy from websites, you can buy directly now from the peer clients. But for initial setup you require a sip. So, the tomorrow's is on Saturday's lecture if time actually permits will actually move and look in to sip setup mechanism. So, how the sip actually goes we will look in to the sip header structure and the interpretation of sip header.