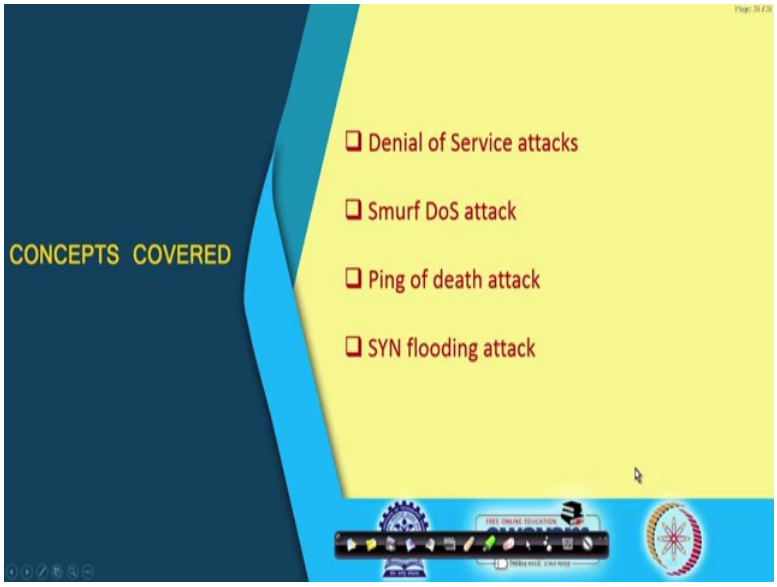
Ethical Hacking
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 38
Network Based Attacks (Part - I)

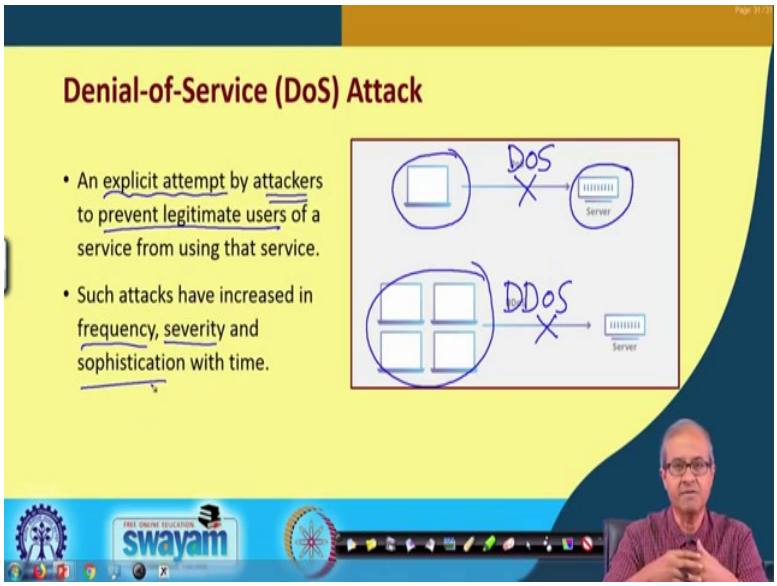
In this lecture, we shall be talking about some of the actual attacks that can be mounted on a network infrastructure; Network Based Attacks, the first part of it. Now, when we talk about penetration testing in a network; obviously, network attack is the most important thing that comes to your mind. Well, you will see the tools, the demonstrations. You must have already seen some of them, but here let me tell you some of the basic ideas and technologies that are exploited, that are used to mount this kind of attacks so that you will have an idea, how this attacks are carried out and how we can possibly stop these attacks, ok.

(Refer Slide Time: 01:06)



Now, in this lecture specifically I shall be talking about Denial of service attacks and specifically three types of such attacks; one is called Smurf DoS attack, other is called a Ping of death, it is not exactly a denial of service. But trying to bring down a system and other is SYN flooding attack which we mentioned very briefly earlier when we are discussing about the TCP connection establishment protocol.

(Refer Slide Time: 01:39)

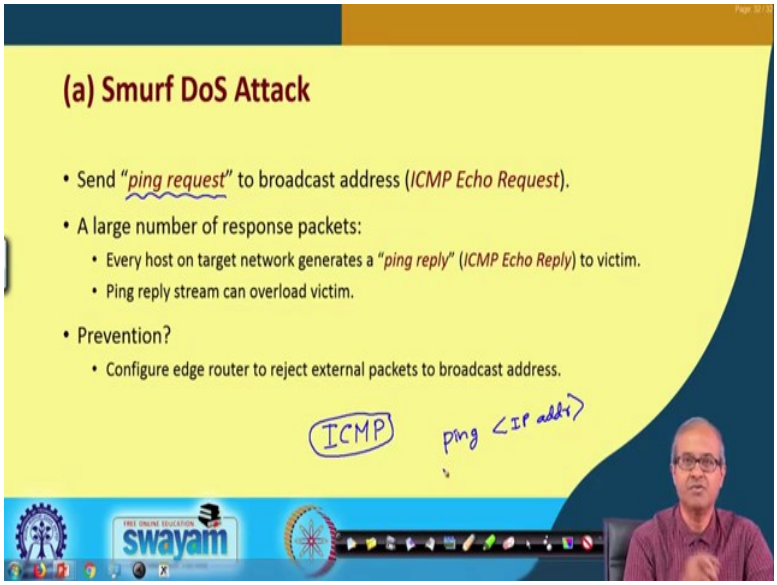


Denial of service attack, we briefly mentioned earlier. Now, what it is exactly? It is an explicit attempt by an attacker or a group of attackers. It can be a single person; it can be a group of persons. They are trying to attack some network infrastructure, some service. And what is the objective, to prevent legitimate users from accessing that service. So, if you look at this picture, you see here I have some kind of a server, let us say.

The server is providing some service; it can be web server, it can be mail server, it can be whatever. It can be an application server and here, we have a user who is sending some request for some service. Now, if there is a denial of service attack, DoS attack that is mounted here; what will happen is that legitimate user will not be have, having access to this service. Now, this kind of attack can be made more dangerous and effective by having some kind of distributed denial of service which means not only one, but several computers can mount such an attack and can bring the server down.

And these kind of attacks have been reported over time. They are pretty frequent in that sense and they have increased in frequency, severity and sophistication where the persons who are attacking now they have a set of tools at the disposal which are extremely sophisticated. So, as a method of defence, if you are the owner of that kind of service, you have to protect yourself adequately so that this kind of attacks cannot be easily mounted.

(Refer Slide Time: 03:57)

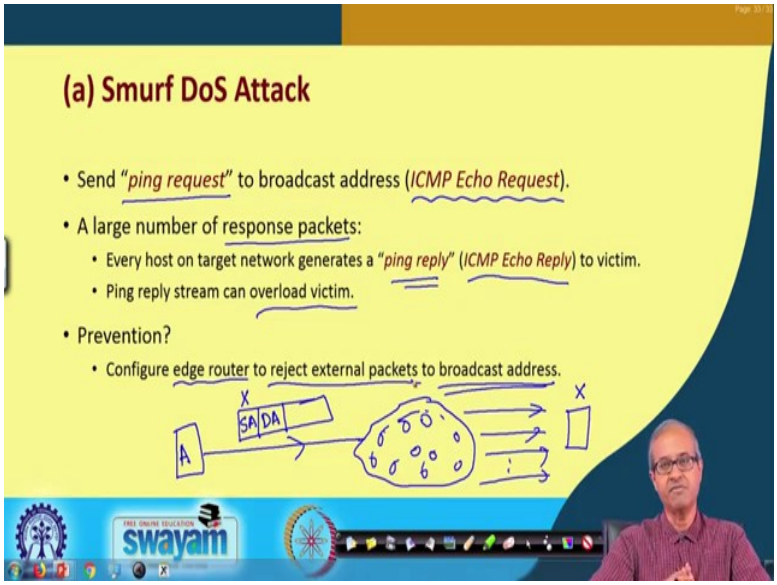


Now, let us look at the first kind of attack called Smurf Denial of Service attack. Now, let me tell you how this attack happens? Now, you see this ICMP is one protocol I talk talked about Internet Control Message Protocol. The ICMP is a protocol that is part of the TCP/IP protocol suit. What it does? It normally sends some kind of error messages between machines. If there is something wrong, it can send an error message to all the persons connected in a network so that others can also know.

While using ICMP you can also try and find out whether some host or some machine is currently available or up or not, you can send something called an ICMP Echo Request; the other person if it is available, it will send back an ICMP Echo Reply. This is one of the features of ICMP. Now, this kind of an attack exploits that this ICMP echo and reply packet response feature that is available in TCP/IP.

You see there is a command called ping which is available on machines. Ping actually generates this ICMP echo request packets. You normally give a command ping with an IP address. So, what will happen? Such an echo request packet will go to that IP address and a response will come back and what we will get? Whether the response is coming back and how much time it is taking for it to come back. Normally, these packets are sent repeatedly one after the other and it will getting continuously, some statistics that whether the network is up and how much time it is requiring to reach and the response come back, ok, fine.

(Refer Slide Time: 06:16)

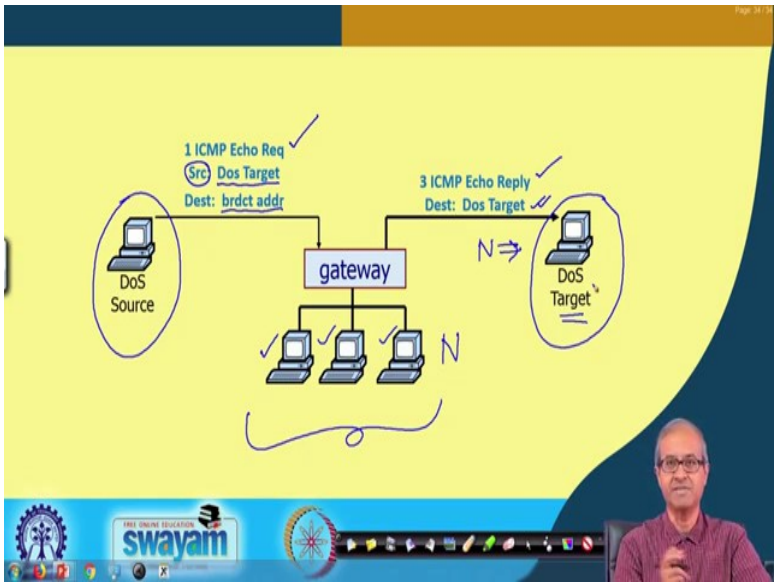


Now, this ping request packet as I said is nothing but this ICMP echo request. In this attack, what is done? This ping request packet is sent to a broadcast address. You look at IP address. I tell you in the host part, if I put all one. It refers to as a broadcast address. Now, suppose I have a network and I want to attack one particular host. Let us say this host is X and this is your attacker, let us see. So, what the attacker can do? Attacker can send a packet to this network with a broadcast address.

So, the destination address will be a broadcast address means which will be sent to everybody; all the computers in this network, this packet will be delivered. But in the source address part the attacker has done some mischief. He has done something called IP spoofing which means instead of the address of A, he has substituted this with the address of X. Let us say there are 10000 computers here in this machine. So, if such a broad cast packet comes, all this 10000 machines will be sending back the echo reply packets. But they will not be sent to A, but they will all be sent to X.

So, X will be flooded with echo reply packets, ok. So, there will be a large number of response packets which are ping reply or ICMP echo reply, which can overload the victim. Now, how you can protect this kind of an attack? Well, one simple way is to configure the edge router in every network to not allow packets to go out which have broadcast address as the destination. To reject external packets with broadcast address as destination well, that is possibly an attempt to mount this kind of an attack, if you can stop such packets, then this kind of an attack will not be easy to mount, right.

(Refer Slide Time: 08:52)

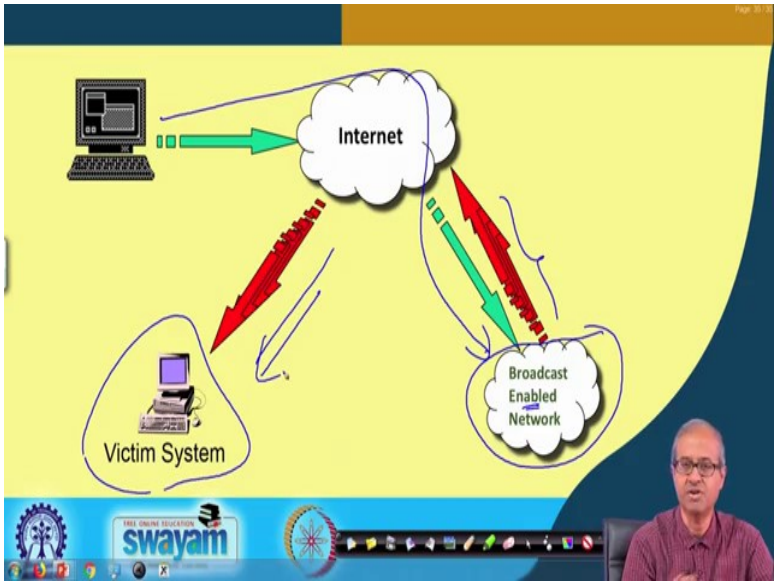


So, this is pictorially what I have just now mentioned. Let us say this is the host which an attacker tries to attack and the attacker is sitting here and this is a network. Let us say there are three computers here, the small example. So, what this attacker does? It will send one ICMP echo request packet with this source address as spoofed, using IP spoofing. So, it will change it to DoS target which is this DoS target and destination will be a broadcast address of this network.

So, it will be reaching all three computers and they will all be sending echo reply packets and they will all be targeted to DoS target. It will be 3 ICMP echo response with the destination DoS target. So, if there are n number of machines in this network, there will be n packets which will be targeted to this. So, this is one simple way to overload a server with junk packets, large number of packets.

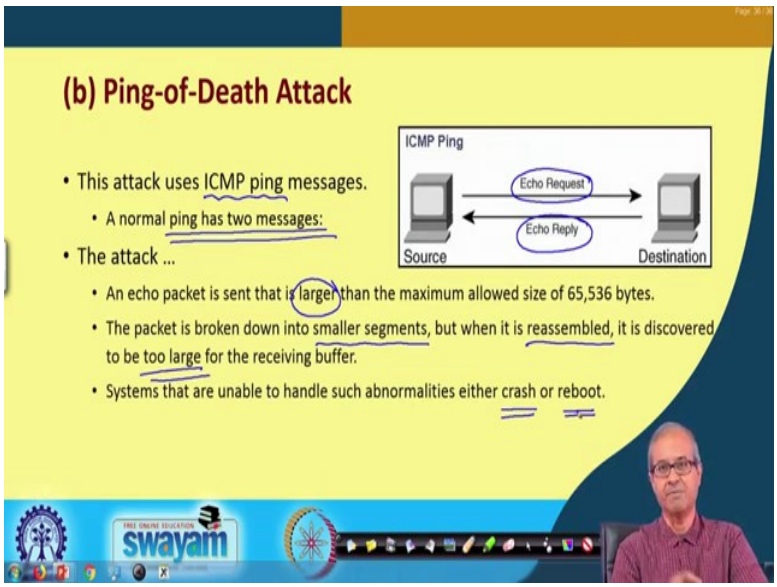
See the idea is that if a large number packet reaches the destination which we are trying to attack, then any legitimate user who is trying to send a packet to access a service will find that the link has become very slow. Because the servers network means network buffer may become full with the incoming packets and the legitimate packet, legitimate packets which are coming, they will find that the buffers are already full and those request packets will be discarded. So, this is like a denial of service.

(Refer Slide Time: 10:51)



So, this is another pictorial depiction, where here we have the attacker. This is your victim system and this is a network which allows broadcast enabled packets to come. Well, if you want to stop this, your network can reject any incoming packet with broadcast address as a destination. Suppose it is not so. It is broadcast enabled network. So this fellow will be sending a packet here and broadcast enable packet will be generating a large number of packets mocked by these red arrows to this victim system. Just what I said, this is the idea.

(Refer Slide Time: 11:40)



Now, this another kind of our attack, this I mentioned. This exactly not a denial of service, but a way to bring a machine down; the idea is like this. This is called ping of

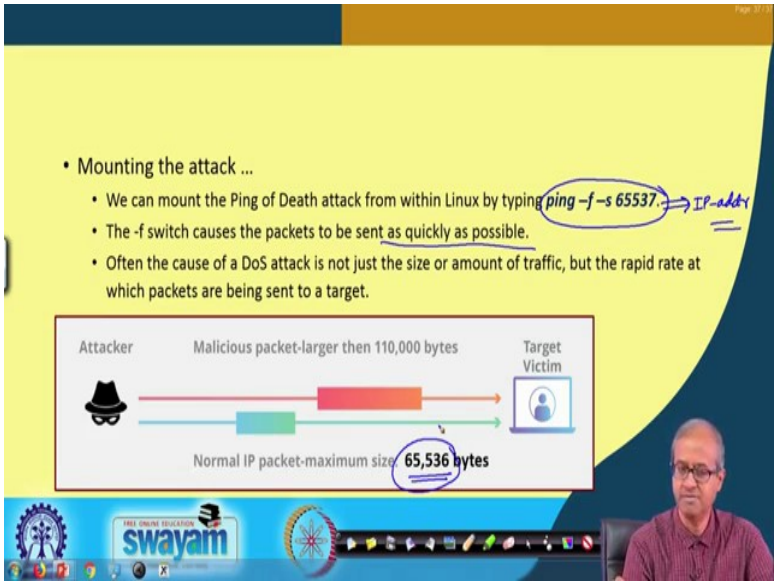
death. Here also we use ICMP ping messages. Now, a normal ping will generate two messages. As this picture shows, if you give a ping from a source to destination, they will be an echo request followed by echo reply.

Now, this echo request and echo reply packets are typically small packets. They do not contain too many bytes. So, it is assumed that it will be certainly less than the maximum allowed size of 64 kilobytes, 65,536. But in this attack, what is done? A echo packet is deliberately sent which is larger than this maximum size. So, what will happen? Because it is larger there will be a fragmentation, multiple IP fragments will be created and they will be sent.

So, they will be smaller segments, the fragments, but at the others side when it there reassembled, it will be discover, discovered that the total packet size will be greater than 64 kilobytes and it cannot be delivered to the ICMP destination, because it can only receive packets up to a maximum of 64 K.

So, there are systems where if such a large packet comes, which is too large to handle, then this, then the software at the receiving end may either crash or the system may reboot. But of course, not all systems are like this. If the systems are not well configured, they might get crashed or reboot, if a two large packet comes and tries to reach an application at the destination. This is the basic idea here.

(Refer Slide Time: 14:07)

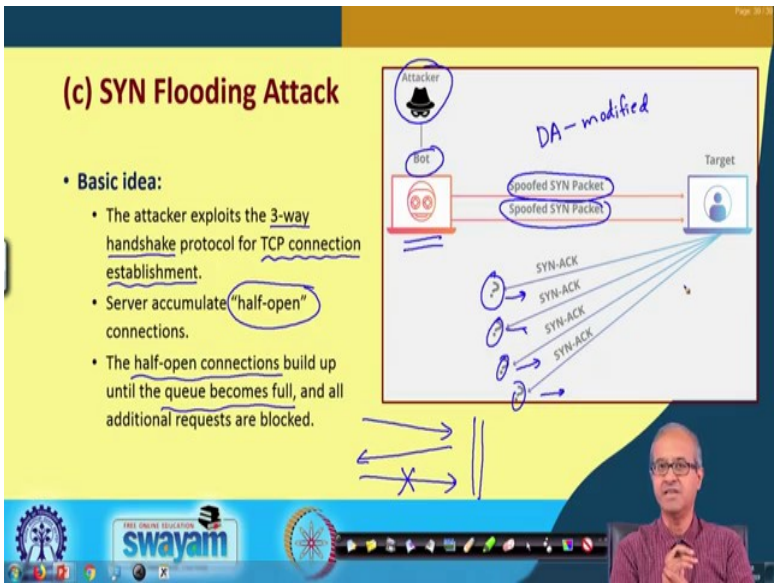


Mounting the attack as I said, so what is the idea? So, on a Linux machine for example, you can give a ping command, of course, followed by an IP address. There will be an IP address. After this IP address with some flags $-f - s$ 65537. So, 65537 is a size which is greater than 65536 that is 64 K ok. This will mean that you are sending a ping packet to this IP address with size is this and $-f$ is a switch which tells that the packets to be sent as quickly as possible, fast.

So, attacker can send such a packet, larger whatever size it is, greater than this to the target victim, but normal IP packet maximum size is 65536. So, if it is greater than that, so this kind of ping of death attack may happen, because it will broken up into smaller packets to reach the target. They will be reassembled and the size will be found to greater than 65536, right.

So, this is how the attack is mount. But one thing you remember. Just mounting this attack is not the only thing, but that means, the amount of traffic, but the rate at which the packets are being sent that is more important. Because if the packets is sent with some gaps, then maybe the buffer overflow will not happen at the receiver end. But if they sent too fast, then the buffer will get filled up very rapidly and at the end there will be some kind of buffer overflow and no new packet new packets can be accepted ok; fine.

(Refer Slide Time: 16:10)



Now, let us come to SYN flooding which we briefly mentioned earlier. Now, in this SYN flooding attack what we really do is that we try to exploit the TCP connection establishment protocol, some weaknesses there. So, you recall TCP connection is established using a 3-way handshake protocol. This is a SYN, followed by a SYN acknowledgment followed by another SYN packet sent. So, there are 3 packets that are exchanged between a source and a target, before a TCP connection is established.

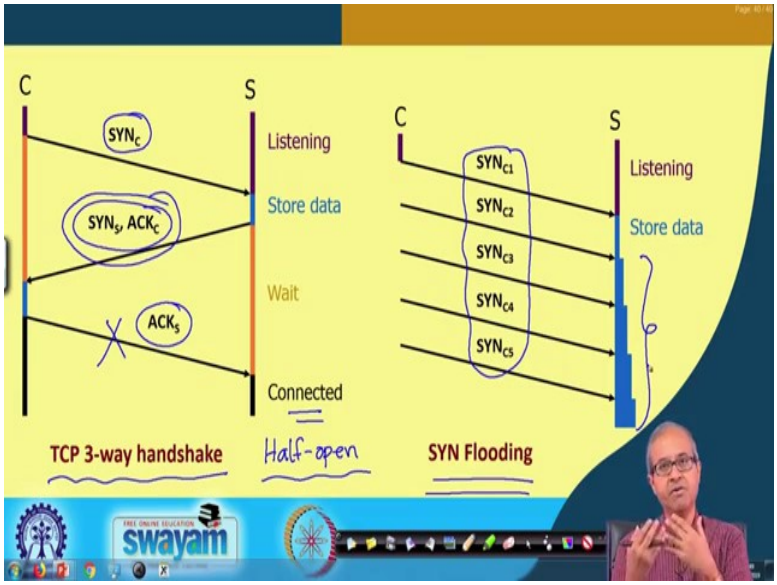
Now, if the third message does not go like, there is one packet coming, one packet coming back, another packet going. If the third message is not send, third packet is not sent, then the server side will maintain this so called half open connection at least till a timeout interval. But this third packet not going this can be a deliberate attempt from the point of view of the attacker, ok. This half open connection, if a large number of such requests are coming similar, but the third packet is not coming back to the server.

This half open connections will be building up at the server side and the pending request queue will ultimately become full and after that all additional requests that are coming in, they will get blocked right. So, pictorially it can be shown as follows. There is an attacker which can control a bot. Bot is like an auto means, it is an autonomous system. It is an automated software which will be doing certain things repeatedly, which may be a part of a malware or a virus which was injected by the attacker. That is called a Bot.

Now, the attacker is controlling a bot. This is the bot. The bot will be sending spoofed SYN packet. Spoofed SYN packet means the destination address, they are modified. Because if they are not spoofed, all these responses will also be coming back to this bot, but bot is the giving arbitrary IP addresses as the source address. So, the responses will be going to some other places, but because the some other places they are not the ones to initiate the connection, they will not obviously be sending back the third response back.

So, they will not be sending back any response to these SYN-ACK packets and the target will be having so many half open connections accumulating over time and this kind of SYN flooding will take place.

(Refer Slide Time: 19:47)

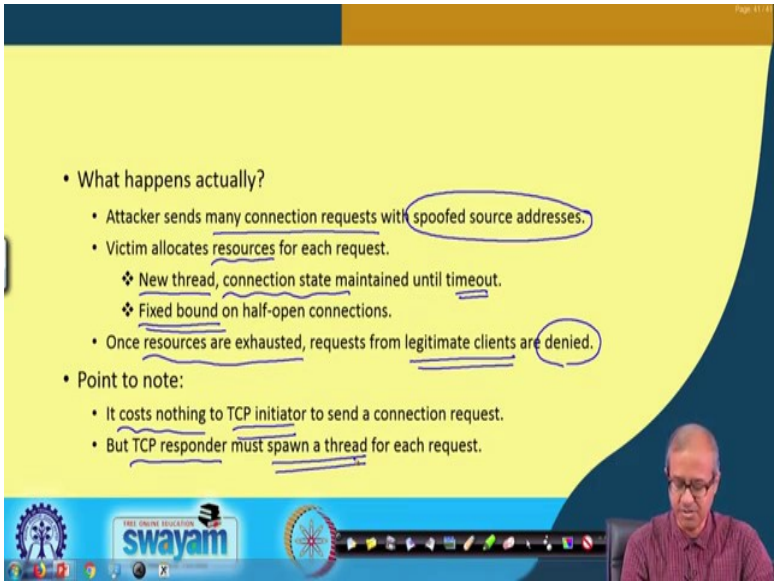


So, let us look at the protocol diagram, the message exchanges. This is the 3-way handshake protocol in TCP. As I said from client to the server, first a SYN packet is sent. Server sends back a SYN and ACK packet both the flag set and the client will finally, send back an ACK packet and the connection will get established. But if this third packet is not send back, then there will be a half open connection.

So, what will happen is that the client will be sending back many SYN packets, where the IP addresses are spoofed. So, the server is sending back this SYN-ACK packet all right, but they are being sent not to the client, but to some other dummy IP addresses which will not be sending back these third ACKs. So, what will happen? The buffers will get accumulated the size. This half open connection information will get accumulated and this is what is referred to a SYN flooding.

So, there will be a situation, where this server will not be able to accept any further connection because the buffer is full and if any legitimate connection is also coming, they will also get rejected, ok. This is denial of service.

(Refer Slide Time: 21:25)



And so, whatever I have said is actually mentioned here. The attacker will send many connection request with spoofed source addresses, this is important. So, IP address spoofing is carried out so that the responses will not be sent to the attacker, but to some other places. Victim will allocate resources for each request, because these are TCP request coming, TCP connection requests. So, temporarily some information has to be maintained in a buffer or a table, ok.

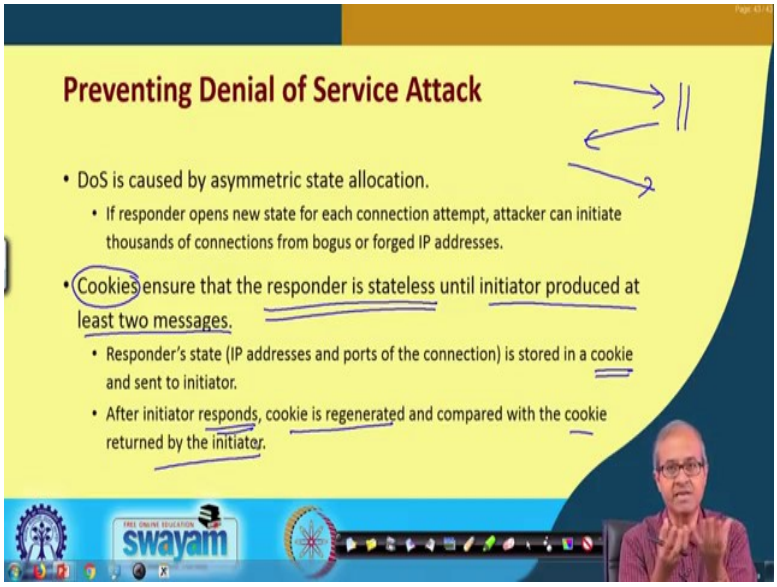
A new thread has to be created also, because each incoming TCP connection request has to be handled by a separate thread. So, whenever there is a connection request coming, a new thread is also created. So, the server also creates a new thread and the connection state is maintained half open connection till a timeout period elapses, beyond which of course, the connection will be rejected. This will be removed from the table.

The table allows a fixed number of half open connections to be maintained, but if the table gets exhausted, no further requests will be accepted and even legitimate client requests will be denied. So, the point to note is that when the attacker mounts the attack, it is costing nothing to the attacker which is the TCP initiator. Just a packet is being sent no buffer space is being reserved at the attacker site.

But the responder, the receiver is spawning a thread in response to every request and also reserving some entry in a table or a buffer to maintain the state of the connection. So, there some cost incurred at the receiving side, there is an asymmetry. The sender is not

incurring any cost only the receiver is incurring the cost that is why this kind of attack is possible.

(Refer Slide Time: 23:41)

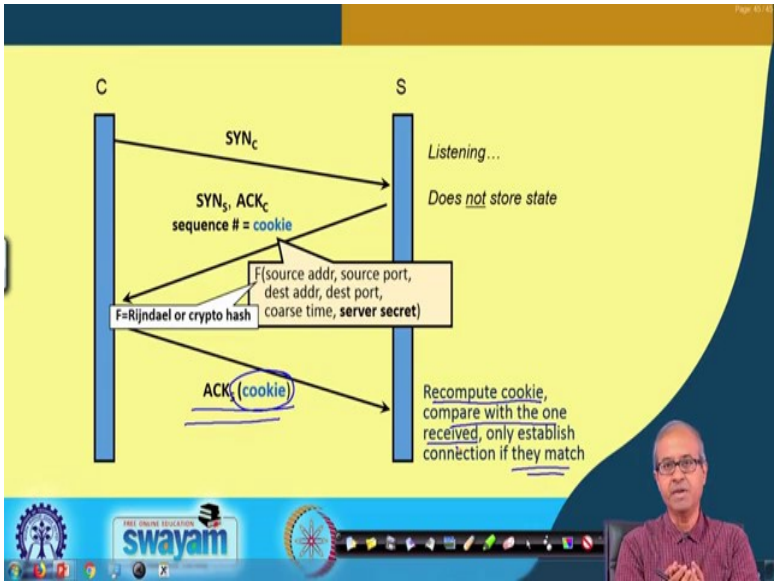


Now, to prevent this kind of an attack, as I said the reason for this attack is asymmetry in state allocation. If the responder opens new state for each connection attempt that is what it happens here. The attacker in the, can initiate 1000s of connection from forged IP addresses, spoofed IP addresses and mount this kind of an attack. Now, one solution says you can use some cookies. So, what the cookie does? Cookie means some information that is temporarily maintained in the server, which is relevant to some connections.

These cookies will ensure that it, responder is not maintaining any state until the initiator produces at least two messages. This one and the second message also. So, it will not maintain state of the half open connection, it will store it only in a cookie. Cookie is much less of an overhead, only a few bites ok.

So, it will be store in a cookie and sent to the initiator and only after the respond is obtained from the initiator, the same cookies regenerated and compared with the cookie written by the initiative whether they are the same connection coming from the same place. If they match, then the connection is accepted and a new thread will be generated. Otherwise no new thread will be generated. Something like this can be done, ok.

(Refer Slide Time: 25:33)



I am just showing with the help of an example here, diagram. These are client, this is server. So, the first packet SYN packet goes. This server responds back with a SYN-ACK and it does not store state in tables, but rather it stores it in a cookie, ok.

Now, this cookie can be containing some information, some sequence number is sent back to the client. It will be a function of the source address, port number, destination number, time and including some server random number it can generate. It is like a challenge response. This server is sending back to the client and this function f can either be a cryptographic encryption process using AES or it can be a cryptographic hash function.

So, what will happen at the end? The client will be sending back an acknowledgement that will also contain the value of the cookie. Now, if this packet was sent to some arbitrary place to, the cookie will not be sent back. So, only if there is a match, the cookies recomputed and compared with the one received and if the match only, then the connection will be established and a new thread will be created. So, in this way the amount of resources that will be accumulating in the server for this kind of request, will be minimized and this kind of attacks can be mitigated, ok. This is the basic idea.

So, with this we come to the end of this lecture. In the next lecture, we shall be talking about some more kinds of attacks including distributed denial of service attacks, exactly how they are mounted and how we can stop them from happening.

Thank you.