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

Lecture - 05
TCP / IP Protocol Stack (Part II)

So, we continue with our discussion on the TCP/IP Protocol Stack. This is the second part of the lecture on TCP/IP protocol stack. So, recall in our previous lecture we talked about the overall TCP/IP protocol stack architecture, what are the different family members in the TCP/IP, notably TCP, UDP and IP.

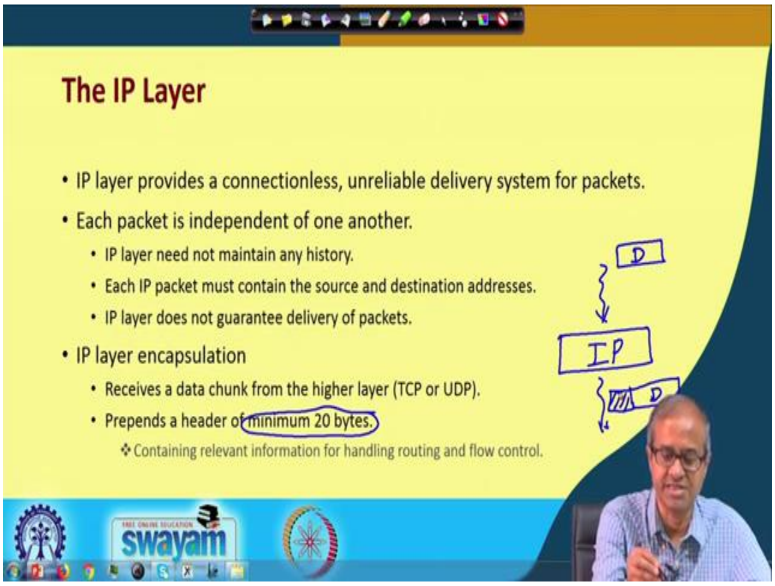
(Refer Slide Time: 00:43)

CONCEPTS COVERED

- ❑ IP Datagrams
- ❑ IP Header fields

Now, in this lecture we shall specifically be looking at some details about the IP packets, the IP formats, the IP header fields and so on ok.

(Refer Slide Time: 00:57)



The IP Layer

- IP layer provides a connectionless, unreliable delivery system for packets.
- Each packet is independent of one another.
  - IP layer need not maintain any history.
  - Each IP packet must contain the source and destination addresses.
  - IP layer does not guarantee delivery of packets.
- IP layer encapsulation
  - Receives a data chunk from the higher layer (TCP or UDP).
  - Prepends a header of minimum 20 bytes.
    - Containing relevant information for handling routing and flow control.

So, let us see talking about IP datagrams. Now, the IP layer what we mentioned? We mentioned that in the TCP/IP protocol stack, the IP layer is nothing but the networking layer in the stack.

Now, in the networking layer, the main responsibility is to route the packets that are flowing through the network, this is the main responsibility of the IP layer. Now, another thing I also mentioned, this we shall be discussing later, that IP is also responsible for breaking up a large packet into smaller packets if required.

So, broadly speaking the IP layer, it provides a connectionless and unreliable delivery system for packets. Essentially, it is a datagram service, it is a layer, which provides a mechanism for transmission and routing of datagrams from one node to another in the network in the internet. Now, in every intermediate node, there will be something called a routing table, these we shall be discussing later in detail.

With respect to the routing table, this IP packets will be coming, they will be compared against and they will be forwarded to one of the outgoing links in a suitable manner. Now, in the IP layer one thing we mentioned just like datagram, each packet is independent of one another, there is no relationship between successive packets.

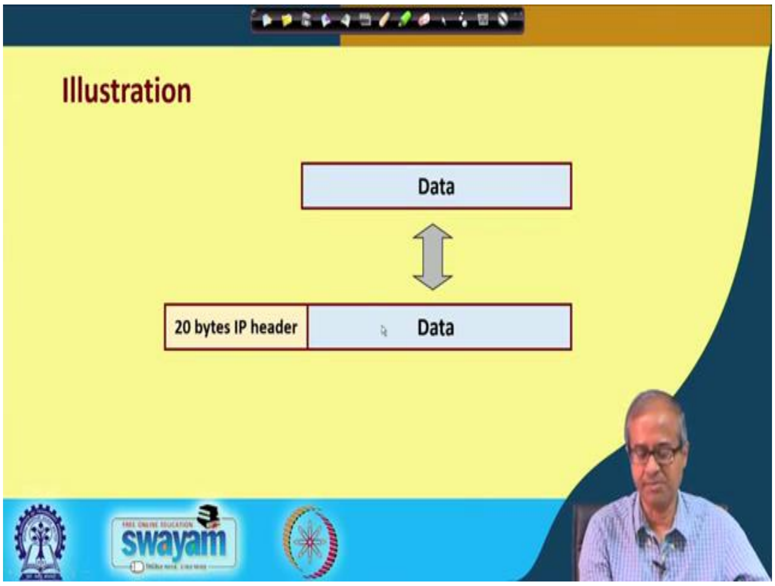
So, in that sense the IP layer does not maintain or need not maintain any kind of history and it must be provided with sufficient information, so that it can forward the packet correctly to the final destination. For that what is required, you need to provide the destination address. That must be there in the header and also it must contain the source address, because sometimes, some acknowledgment or some information needs to come back to the sender, that is why both source and also destination address are included as part of the IP address.

And, just like datagram service the IP layer does not ensure reliability and it also does not guarantee delivery of packets, some packets might get lost. In addition I told you, duplicates might get generated, packets may get also delivered out of order. So, all these things can happen. Now, other thing is, this IP layer provides some kind of encapsulation, which in our previous lecture, we have very briefly looked at by taking an example of that trivial file transfer protocol. We had seen that when a data chunk is moves from the higher layer to the lower layers in the TCP/IP protocol stack, some headers get progressively added.

Now, if you think of the IP layer, this is my IP layer and, some data packet is coming to the IP layer, some data packet, let us call it D. So, what IP will do, it will add some header to this data packet; it will add some header to this data packet.

And, IP layer adds minimum of 20 bytes of header, typically 20 bytes only, but in some times, it can be more than 20 rarely, but typically 20 bytes, 20 bytes of header is added to the data. And this includes source and destination address and some other information also ok, which helps in routing the packet and some other services this we shall see.

(Refer Slide Time: 05:33)



Illustration

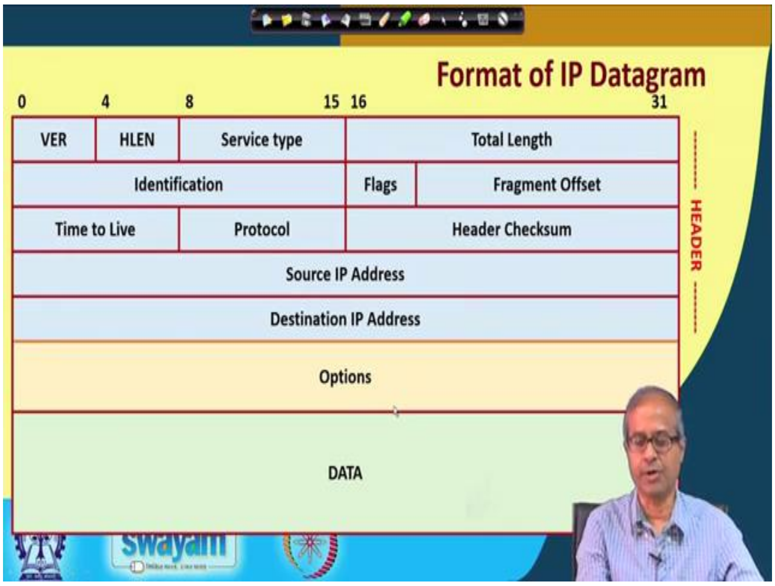
Data

20 bytes IP header | Data

So, just this is what I am talking about some data is coming to the IP layer and the IP layer is adding some 20 bytes of header to it.

And is then forwarding it to the lowermost hardware layer or the data link layer whatever you call ok.

(Refer Slide Time: 05:49)



Format of IP Datagram

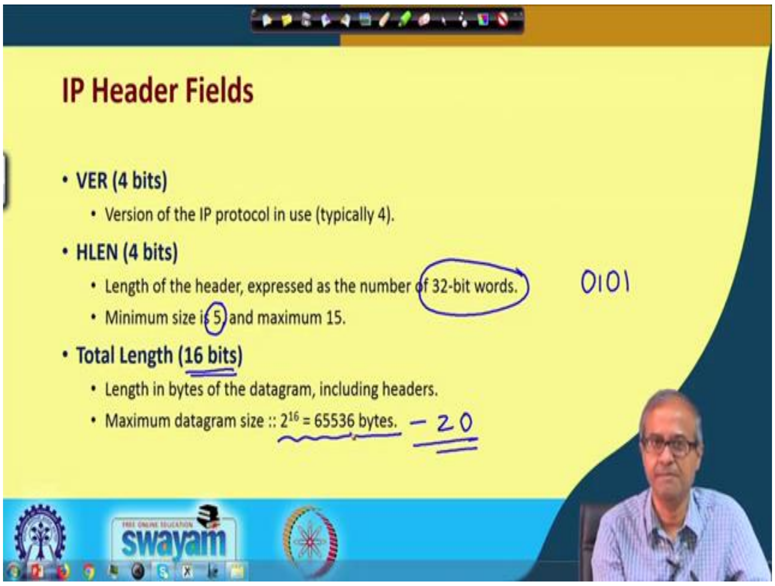
VER	HLEN	Service type	Total Length	
Identification			Flags	Fragment Offset
Time to Live		Protocol	Header Checksum	
Source IP Address				
Destination IP Address				
Options				
DATA				

This is an overall picture which shows the format of an IP packet. So, this is the format of an IP datagram. Where you can see, the first part of it here? These are the headers and this part is an optional header, you can have some additional header in some cases, but the first part of it is mandatory.

And, after that the actual data that you want to send ok. Now, let us see what are the fields that are there in the IP header? Let us look at it. The way I have shown this table is that you see these numbers on top, these are the bit numbers, you see it starts with 0, it ends with 31, which means in every row there are 32 bits, which means 4 bytes and there are 5 such columns you see 5 such columns.

So, the header is total 20 bytes, 4 bytes in each row, I am showing 1, 2, 3, 4, 5, 20 and in addition you can have some optional header fields also. Now, here we shall be looking at the function of these different header fields, there is a version, which will tell you about the IP version, this is header length, service type, total length of the IP packet, some identification of this packet, some flags, fragment offset, time to live, I will explain these things, protocol, header checksum and these are IP addresses. You see source and destination IP addresses are 32 bits long, they occupy entire 32 bits ok. Now let us see this different header fields one by one what they are.

(Refer Slide Time: 07:59)



IP Header Fields

- **VER (4 bits)**
  - Version of the IP protocol in use (typically 4).
- **HLEN (4 bits)**
  - Length of the header, expressed as the number of 32-bit words.
  - Minimum size is 5 and maximum 15.
- **Total Length (16 bits)**
  - Length in bytes of the datagram, including headers.
  - Maximum datagram size :: 2^{16} = 65536 bytes. − 20

0101

First we will indicates a version which is a 4 bit field, which indicates which version of IP you are using.

Now, here I am assuming this IP version 4, but there are some newer version of IP, version 6 which is also there, which is already being used in many places. We shall be talking about IP version 6 later, but normally for our normal computers which we use, there you use IP version 4. So, this first 4 bits will contain the number 0 1 0 0 which is 4.

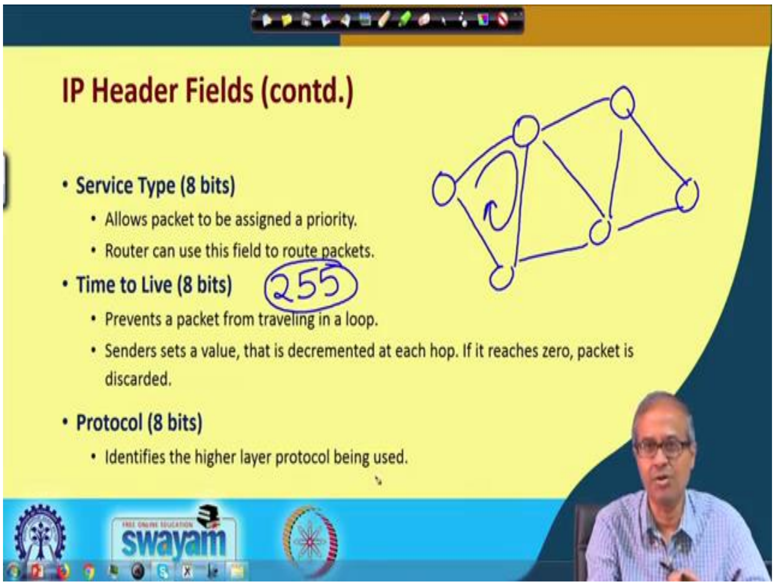
The next 4 bit indicates the header length, what is the size of the header? So, here the size of the header is expressed in multiple of 32-bit words. So, as you can see that the first five rows were there and there are some options.

So, if it is only 20 bits of header, this will contain 5, but it can go up to a maximum 15 because it is 4 bits, in 4 bits you can have 0 to 15. So, default is 5, 0101 which means 5 × 32 which means 160 bits ok. Then there is a field which is 16 bits long, this is total length, this indicates the total size of the datagram including the headers. Now in 16 bits what can be the size, 2^{16} is 65536 so, many bytes.

And, if you subtract the header suppose in the minimum the header can be 20. So, you go minus 20, that many bytes can be there as part of the data, that can be maximum size of the data in IP.

This is what IP supports.

(Refer Slide Time: 10:05)



IP Header Fields (contd.)

- Service Type (8 bits)
  - Allows packet to be assigned a priority.
  - Router can use this field to route packets.
- Time to Live (8 bits) 255
  - Prevents a packet from traveling in a loop.
  - Senders sets a value, that is decremented at each hop. If it reaches zero, packet is discarded.
- Protocol (8 bits)
  - Identifies the higher layer protocol being used.

Then there are some other fields which some of these are used very rarely. For example, service type, this is an 8 bit field which contains some additional information which allows a machine to tell, whether you are wanting some special kind of service. Like I mean whether you want to give some higher priority to your packets, well if you give some higher priority, then your packets will be routed first. You will get faster performance, you will get more bandwidth like that, but not all routers support this feature. So, this feature is again used very sparingly, normally you do not use this.

Now, there is another very interesting field, this is called time to live. This is an 8 bit field; 8 bit field means you can have a maximum value of $2^{8} - 1$, which is 255, this can be the maximum value. What this field contains, you see, you think of a network. These are all intermediate nodes, these are the routers let us say.

So, these IP packets, they will flow from one node to the other. Now, normally what do you expect that, you will be given the destination address, normally you would expect that the packet will be forwarded in the right direction and will finally, reach the destination. But, so many things can happen in the network over time, some link might go down, some networking node might get down, some network, the routing table may also get corrupt ok.

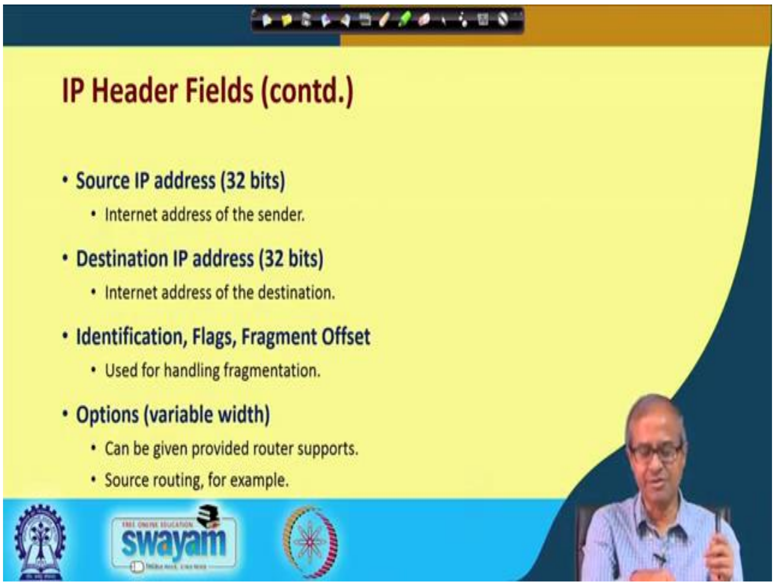
So, what might happen is that instead of getting forwarded in the right direction your packet might accidently get forwarded in the wrong direction also. Now, in the extreme case what may happen, your packet might go along indefinitely in a cyclic loop. Like for example, your packet might go along in the cyclic loop indefinitely, it will never go out.

If due to some reason, the routing table entries have changed or they have got modified in such a way that the packet will get forwarded like this only; now, you do not want such a situation to happen. Now this time to live is a field which is put in the header with an initial value. Now how much initial value that depends on the network, you can set that value suppose 50.

Say every time an a packet is forwarded from one node to the other the value in that packet is decremented by one, it becomes 49, next time 48, 47, 46, like that, like that whenever the value reaches 0, the packet is discarded.

So that you do not allow this indefinite cycling to go on, this is time to live, how long this packet can live ok, this is that. And, protocol is another field, this is also 8 bits, this will identify which higher layer protocol you are using. Like TCP and UDP are the most important protocols, they have a code, 8 bit code, there are other protocols also. In that case you will have that that particular protocol number in 8 bit which high level protocol is being used here.

(Refer Slide Time: 13:57)



IP Header Fields (contd.)

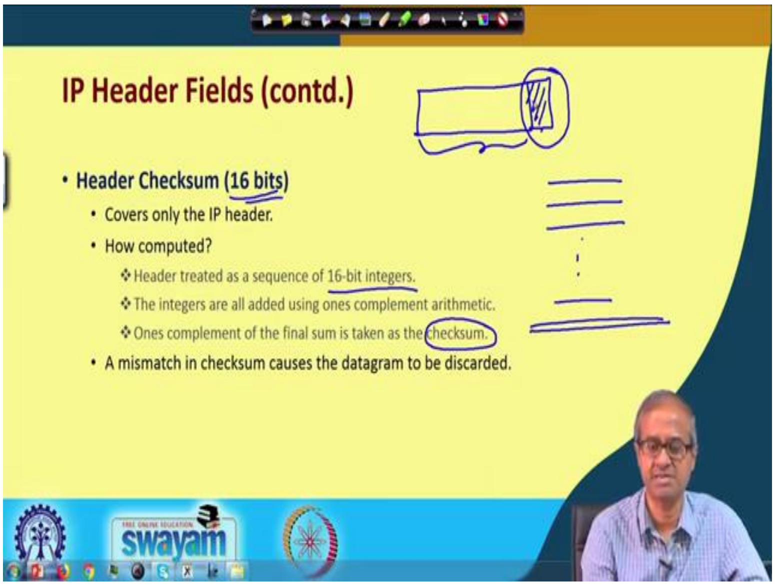
- Source IP address (32 bits)
  - Internet address of the sender.
- Destination IP address (32 bits)
  - Internet address of the destination.
- Identification, Flags, Fragment Offset
  - Used for handling fragmentation.
- Options (variable width)
  - Can be given provided router supports.
  - Source routing, for example.

Then of course, you have this source and destination IP addresses. As I said, these IP addresses are 32 bit addresses ok. Then, you have some other fields which we shall be discussing later, like identification, flags, fragment, offset. These fields are used for fragmentation, whenever the IP layer requires to break a packet into smaller packets and again it may have to combine those smaller packets together into a single packet, these are called fragments, fragmentation and reassembly. So, to support that these three fields are required, this we shall be talking about later, when you talk about fragmentation and reassembly ok.

And of course, we talked about the variable width options header, you can have some additional header fields you can add. These additional fields are required in some cases where some routers may be having some special support, like one interesting thing may be something called source routing. You see normally the way a packet will be routed, you leave it up to the IP layer in the routers ok.

You send a packet to the destination address and let the routers decide, the IP layers in the routers decide. But some routers may be having a feature called source routing, where you as the transmitter as this source know that this is the best route. And I want my packet to follow this route; that means, this source is specifying the route; that means, you to specify that route, you need a longer information to be specified. So, this options field can be used to specify the sequence of IP addresses of routers, which has to be followed to go to the destination, this is called source routing. So, these options are there.

(Refer Slide Time: 16:19)



IP Header Fields (contd.)

- Header Checksum (16 bits)
  - Covers only the IP header.
  - How computed?
    - ❖ Header treated as a sequence of 16-bit integers.
    - ❖ The integers are all added using ones complement arithmetic.
    - ❖ Ones complement of the final sum is taken as the checksum.
  - A mismatch in checksum causes the datagram to be discarded.

And, here lastly now for error checking, there is a field which is there, which is called header checksum. Like, whenever a packet is transmitted, like let us say I transmit a packet, you receive that packet. There may be a some error in transmission, some bits might get corrupt, 1 might get 0, 0 might become 1.

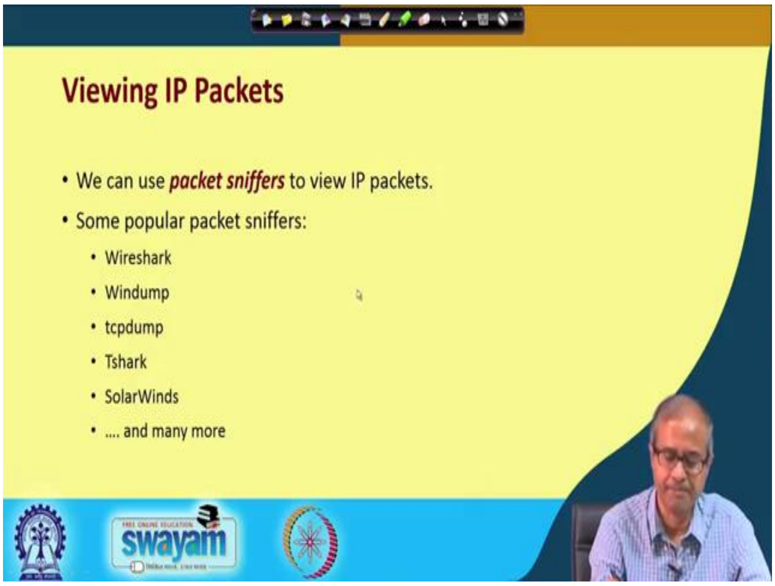
So, immediately whenever a node receives a packet, there has to be some simple checking whether there has been any error in transmission or not. And, this header checksum is used for that purpose, the idea is like this; suppose you have the header and you have one field in the header, this is the checksum, let us say this is the checksum.

Checksum is a 16 bit field. So, how do you calculate the 16 bit fields? And, this checksum is computed only on the header, you see the header, you treat it as a set of 16 bit numbers. So, I showed it as five 32 bit numbers; so, there will be ten 16 bit numbers minimum plus options. So, there will be several 16 bit numbers right like this.

So, what you do, you all add them up, you add up all the numbers using 1's complement arithmetic right. Just assume that these numbers are ones complement integers, you add them up. And, after adding them up you take 1's compliment of the final sum; 1's compliment means just to take complement 0 becomes 1, 1 becomes 0. That you take as the final checksum, simple yes that is some addition and then complementation, this can be done very fast, this is how this checksum is computed.

So, it is actually whenever a packet is received maybe by a computer, maybe by a router, this checksum computation is done typically automatically by the hardware. So, whenever a packet is received, the header fields are compared, the checksum is computed and the computer checksum is compared with the received checksum, whether they are matching or not. If, they match it is fine, if they do not match, you report that the packet has been corrupted, you reject the packet well.

(Refer Slide Time: 19:03)



Viewing IP Packets

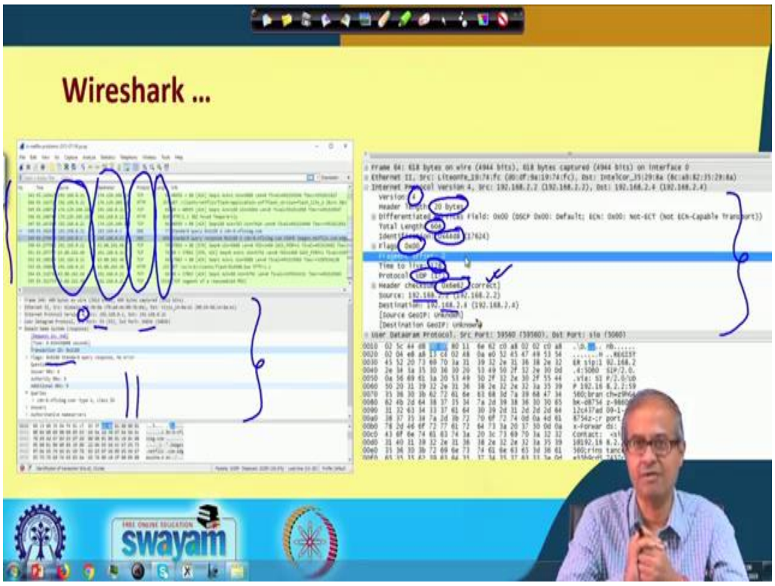
- We can use ***packet sniffers*** to view IP packets.
- Some popular packet sniffers:
  - Wireshark
  - Windump
  - tcpdump
  - Tshark
  - SolarWinds
  - .... and many more

Now, there is another thing which you shall be seeing much more during the demonstration sessions, you see this IP packets are something very interesting. There are so many fields and if you look at these fields, you may be able to understand; that means, what these packets are, what it means, what this fields mean and so on. There are a number of software tools which are available, many of them are free, these are called packet sniffers.

Now, using this packet sniffers you can view packets not necessarily only IP packets, but basically any kind of packets or frames that are flowing through the network, you can view them. And, you can install this packet sniffer software in your computer, in server, anywhere in the network. And whatever packets are flowing across the network interface of that place where you are installing, you can see all those packets, the information about those packets on your screen whenever you want to right.

And, these are the names of some of the popular mean packet sniffers which are available and here some of these you will also be seeing during the demonstration sessions later.

(Refer Slide Time: 20:27)



Wireshark ...

Here I am showing the typical screenshot of one of the packet sniffers called Wireshark, you see in this particular packet sniffer, you see there are several windows. In the first window out here, you see every line indicates some packet which has received. The font size are small, I do not know whether you are able to read it or not, here you can see what kind of protocol is being used.

You see some are TCP, HTTP, DNS, TCP and so on. You can see that what kind of packets are flowing through the network. So, whenever you give a command, you are doing a browsing the Internet some http packet will be generated, HTTP, Hypertext Transfer Protocol.

Depending on what you are doing that kind of packet will be generated and it will be immediately captured by the sniffer. And, you can see time that what time, what is the source, these are the source and source IP address, these are the destination IP address ok. This is the length how many bytes and some information.

Now, if you select one of the packets; if you select one of them on the window then out here you can see the details. For one of the selected ones you can see here, you can see this internet protocol version 4, UDP, port number, destination port number, DNS, you see flags ok. So, these informations are all there, whatever fields are there, you can see all the fields out here. And, just if you click on it you can see it on a separate window as it is seen here.

If you do a double click, you can also open a bigger window and see it like this. Like you can see, you see in the means, I mean, I told you about the IP fields, you can see here version 4, header length 20 bytes, total length 604, this is the id, I did not mentioned this is id, these are the flags, fragment offset, this is the time to leave, 128, I told about the time to leave. Protocol is the, UDP is the 17, the value 17 means UDP, this is the header checksum, which has been verified to be correct source address, IP, destination you see, means all these fields of the IP header you can view here if you want ok.

Here you can analyze the network traffic whatever is going on in the network? If, you want to analyze the kind of traffic that is going on, you can just capture the packets, then you can do an analysis using this kind of sniffer tools like Wireshark or any other tools ok. So, just I wanted to show you these are the tools available if you are interested you can also download these, these are available for free. You can download them on your machines run them and you can have a feel of these tools, how they work, ok.

So, with this we come to the end of this lecture. So, we shall be continuing with some more discussion on some other aspects of this TCP/IP protocol, because you see this entire networking today the Internet is based on this TCP/IP. So, unless you have a good knowledge about TCP/IP and how some of the things work out here. It will be really difficult to look at some of the advanced topics like, you like here you see, you are talking about hacking a network.

So, whenever you are talking about hacking, you are basically saying that I am trying to analyze some kind of network traffic and accordingly I am trying to do something so that I can break into a system ok. So, for this you need to understand some of the networking basics very well.

Thank you.