

Information security - IV
Prof. M J Shankar Raman
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

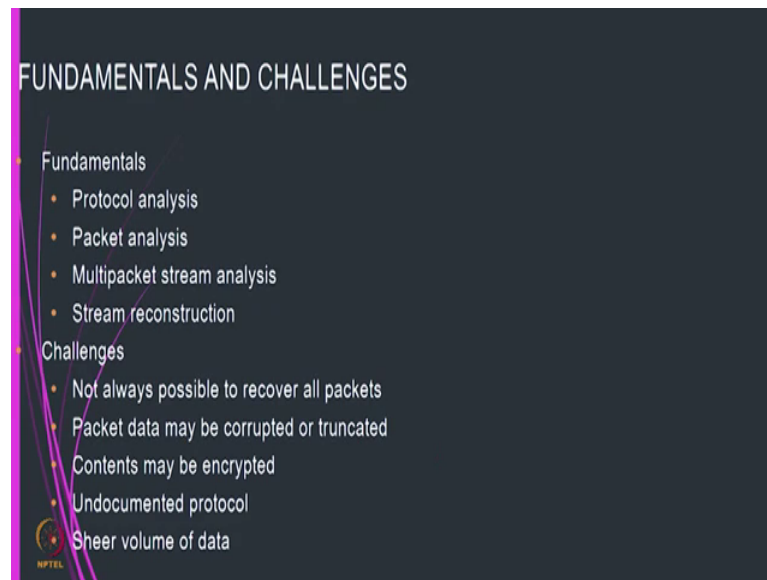
Lecture – 39
Packet Analysis
Network Security and Forensics

In the last session, we had a look and the wire shark tool. We request that you go through other sources for understanding about wire shark, because we will be using this wire shark tool extensively sometimes if you are not going to use command line you can actually use the wire shark tool to achieve the same effect as what we do in command line.

So, what are all the things that we can do with a wire shark tool or other tools? We will also be introducing some other tools like t shark etcetera and if you want to install them just go out and install all these tools are available many of them are open source and almost all of them are open source. So, you can go out and install it in your machine, and we suggest that you use a Ubuntu machine not windows because all these tools are easily downloadable and then you can just go out and install and then since you will have administrator permission, even if the machine goes off you can just get it recovered.

One of the things that we would suggest that you also use a virtual machine to try out all these things, because if there is a problem then virtual machines can be required much better than the real machines anyway.

(Refer Slide Time: 01:25)



Now, we will do some to some something on packet analysis. What we have been doing is that, we have been capturing packets using wire shark. What do you do after capturing the packets? So, there are actually four activities that we can do. One is the protocol analysis. So, we saw that one of the fields that were shown by a wire shark was the protocol.

So, we can do some kind of protocol analysis using wire shark, then we can also do packet analysis ok. So, what kind of packet what is the data it carries etcetera then we can also do multi packet stream analysis because if you take internet the packets actually traverse not as continuously, but in small chunks the size is determined by what the routers feel should be their correct size ok. What happens is that if you just take 5 or 6 packets, they will be connected together and that is called as a stream ok. So, multi packet and then we can find out what are for example, a protocol like sip, it has a control plane and a data plane which is separated because when you dial a number ok.

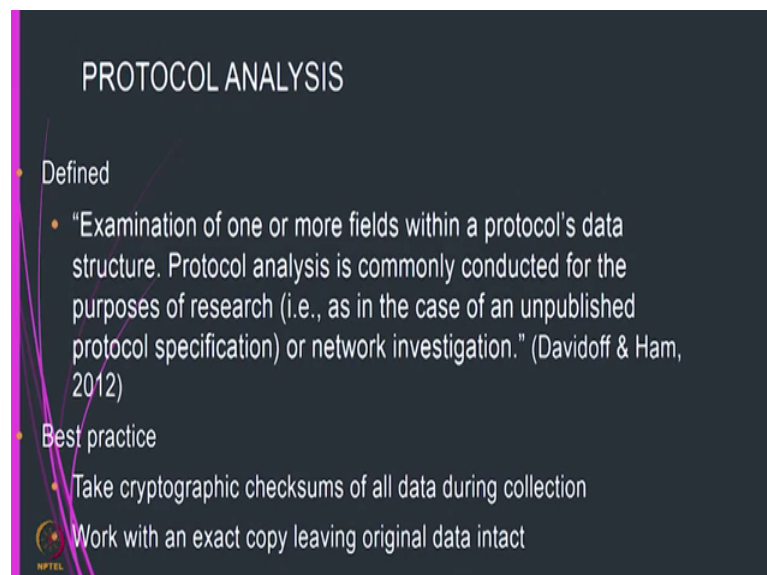
So, what happens is that, it the dialer goes to a separate server and once the connection gets established then the part that is taken by the data is different. So, if you want to collect a forensic evidence of voip, then you have to correlate the between the control plane where dial things happen and the data plane where the actual talk gets exchanged. So, all these things then you have to reconstruct these two together. So, these are all the fundamental things that you will do after collecting the evidence or gathering acquiring

the evidence. So, there are a lot of challenges in this activity the first and foremost is that not possible to recover all the packets and sometimes when you are collecting you may not have collected it properly. So, the package could have got corrupted or it could have got truncated or the third possibility is that the contents might be encrypted. So, you might need some kind of encryption key to identify what encryption it is and or decrypt the packets many of these service providers not. So, the content providers actually use some undocumented protocol.

Now, this is where the whole challenge arises for example, whatsapp use it its own encryption strategy which is which does not let out what interruption strategy it is using. So, some if a network forensic expert might have to crack these things, with the help of hackers or even if he is capable of doing it; so that is say other problem there could be undocumented protocols that are being used by organizations. Many organizations use this kind of undocumented protocols for security purposes and the last the final one is the volume of data.

I think we had discussed it about how much data flows through a router and how difficult it is to store all this data and work with it. First let us see what is protocol analysis ok.

(Refer Slide Time: 04:28)



PROTOCOL ANALYSIS

- Defined
 - "Examination of one or more fields within a protocol's data structure. Protocol analysis is commonly conducted for the purposes of research (i.e., as in the case of an unpublished protocol specification) or network investigation." (Davidoff & Ham, 2012)
- Best practice
 - Take cryptographic checksums of all data during collection
 - Work with an exact copy leaving original data intact

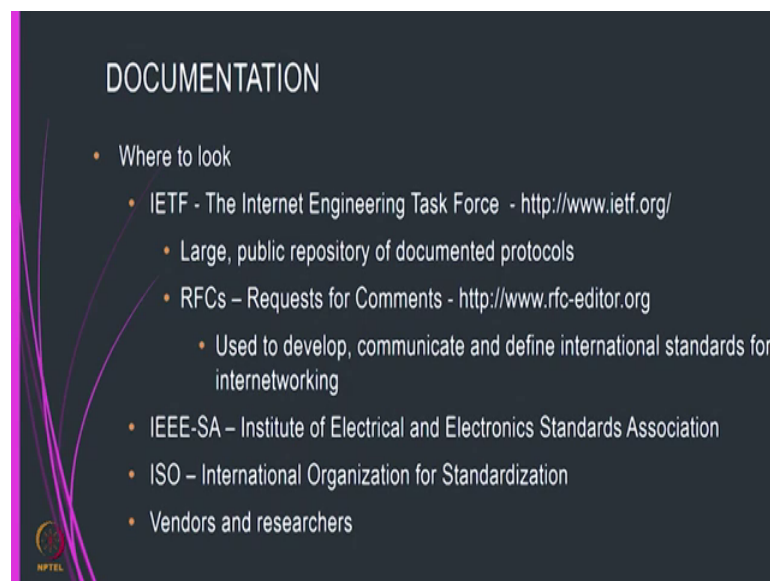
NPTEL

It is defined as an examination of one or more fields within a protocols data structure. A protocol analysis is commonly conducted for purposes of research as in the case of unpublished protocol specification or network investigation.

So, as I told you if there are certain organizations, which actually have unpublished protocol specification, we might have to decrypt it. One of the examples was AOL America Onlines chat protocol ok. So, there is a sort of proprietary and, but people actually broke it. So, one of the things in network security is, if you try to keep the algorithm closed and people will try to break it. So, usually the algorithm becomes open, but the way you transfer the keys etcetera becomes closed and. So, the best practices that to take cryptographic checksums of all data during collections ok. So, that tomorrow after you decrypt and then if you are able to find the encrypted a you see that the checksum matches, and as we had discussed earlier and we had done it in wire shark also in the previous session you have to work with an exact copy of the original data intact.

So, that is why they actually store the data and then use it; for doing protocol analysis which is the best place.

(Refer Slide Time: 05:53)

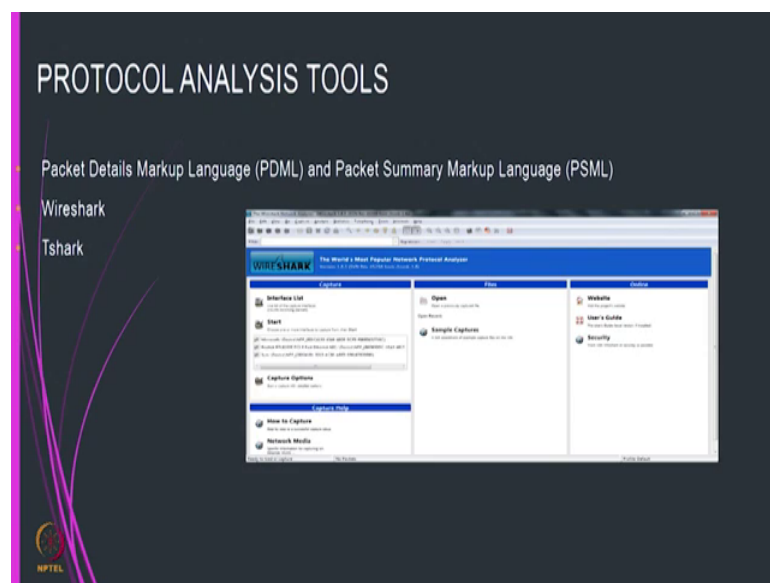


So, many of these protocols are published by the internet engineering task force. I think we had heard about it in previous sessions in information security three course, it is a large public repository of documented protocols, they have a structure of writing the protocols and all these protocols are written in text files. So, that its easy for you to any editor to take a look at these kind of files ok, they are published as requests for commands or RFCs. These RFCs are regularly updated. So, having an accept when it becomes a standard, when it becomes a standard they give RFC number for it for

example, IP TCP all these have some standard numbers ok, but for other protocols until the concurrence is obtained. So, they it does not until it becomes a standard then particular number is not applicable. Other than ietf there are ISO there then the international institute I mean institute of electrical and electronic standardization then I triple E is one place ok.

But sometimes I triple E documents you are pay money to get the protocol standards and the best place for searching all these things is vendors and researches.

(Refer Slide Time: 06:59)



What you can do is that there are many protocol analysis tools, now one of the things that you can do is see wire shark captured captures the packets in a particular format. I think we actually stored it as a p cap ng format. There are some markup languages one is known as the packet details markup language and the packet summary markup language ok. So, the these are the I mean the advantage of using PDML and PSML is that if a parsers for this then you can write code much easier to identify and then automate many of these things. There is also this command line t shark ah but we will just and you can actually convert wire shark file to PDML or packet sml, PSML etcetera.

(Refer Slide Time: 07:56)

LANGUAGE AND PACKET SUMMARY MARKUP LANGUAGE

- PDML
 - Expresses packet details for Layers 2-7 in an XML format
 - Example:
 - `$ tshark -r capturefile.pcap -T pdml > xml.pdml`
- PSML
 - Used for most important details about a protocol also in XML
 - Example:
 - `$ tshark -r capturefile.pcap -T psml`
 - Part of the NetBee library – support packet processing

<http://www.nbee.org/doku.php>

NPTEL IMAGE'S CLIPPED FROM WORK CITED

```
<pdml>
<packet>
...
</packet>
<packet>
...
</packet>
...
</pdml>
```

```
<psml>
<structure>
...
</structure>
<packet>
...
</packet>
<packet>
...
</packet>
...
</psml>
```

The PDML expresses packet details of layers 2 to 7 the ISO standard OSI standard ISO OSI standard.

It actually shows that in the form of an XML package. So, it is shown on the right hand side that PDML and then each one of this is a packet and then one of the ways you can do it is that, you can use a tshark tool and then read the pcap file. So, in this case you have put a pcap file as an example, and then translate it to PDML you might have to write you to some files for XML XML dot PDML or whatever it is to ensure that you can store these things. Similarly for PSML you can do the same thing if you see minus r is to read the file and then change it into this particular format and then you can. So, this is the structure of the PSML format. So, the point is that you convert it to whatever format to which you are comfortable and it also depends on what are you trying to analyze if you are going to write a lot of scripts then converting to these format is a good idea ok.

(Refer Slide Time: 09:09)

TSHARK

- Same functionality as Wireshark using command-line interface
- Basic commands
 - \$ tshark -r capturefile.pcap
 - Capture file
 - \$ tshark -n -r capturefile.pcap
 - Disable network naming resolution to show IP addresses and port numbers, -n
 - \$ tshark -r capturefile.pcap -T pdml
 - Select output format using t flag
 - \$ tshark -r capturefile.pcap -T fields -e frame.number -e ip.addr -e udp
 - Prints a specific field, -e flag
 - \$ tshark -r capturefile.pcap -d tcp.port ==29008 , http
 - Decode as, -d
 - \$ tshark -r capturefile.pcap -R 'ip.addr == 192.168.1.1'

So, t shark is similar to wire shark, but its a command line interface ok. So, these are all some basic commands for t sharks. So, in case you do not have a command line interface available. So, probably you can use t shack and this is almost the same features of wire shark, but wire shark is much more easier to use that is why we actually demoed wire shark. So, this is just given for your reference if you want to try out this we you are welcome to go ahead and try using t shark. Essentially the functionality that is provided by t shark it is also available in win shark.

(Refer Slide Time: 09:40)

TSHARK DISPLAY

```
C:\Windows\System32\WindowsPowerShell\v1.0\PowerShell.exe
0.000000 Apple_c6:108:1d -> Broadcast ARP 52 Gratuitous ARP for 140.211.183.238 (Request)
1.024889 Apple_Ba:c1:60 -> Broadcast ARP 52 Gratuitous ARP for 140.211.183.238 (Request)
3.377941 IntelCor_10:6c:42 -> Broadcast ARP 52 Who has 140.211.154.110? Tell me (Request)
4.483227 IntelCor_10:6c:42 -> Broadcast ARP 52 Who has 140.211.154.110? Tell me (Request)
5.427888 IntelCor_10:6c:42 -> Broadcast ARP 52 Who has 140.211.154.110? Tell me (Request)
5.939210 IntelCor_10:6c:42 -> Broadcast ARP 52 Who has 169.254.95.54? Tell me (Request)
10.239936 Apple_3c:16f:5c -> Broadcast ARP 52 Gratuitous ARP for 140.211.155.218 (Request)
10.956804 SeikoEps_2f:e0:61 -> Broadcast ARP 64 Gratuitous ARP for 140.211.182.224 (Request)
11.079246 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
11.079110 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
13.311881 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
13.516618 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
14.847853 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
15.257441 SamsungE_14:96:0f -> Broadcast ARP 52 Who has 140.211.182.224? Tell me (Request)
16.486224 Apple_e6:147:3c -> Broadcast ARP 52 Gratuitous ARP for 140.211.154.60 (Request)
15 packets captured
C:\Program Files\Wireshark
```


are doing all these things. So, after that you have to search for relations and then that will become the evidence file, then you have to organize it there will be huge volume of this ignorance file, then you have to again include the structure, I mean you might have to collect the metadata for this, and then collect this as evidence whether then you have to form certain hypotheses and then reevaluate the hypothesis. And then finally, using correlation and other things, things you just have to prove that is something. So, this is the work of a forensic expert.

(Refer Slide Time: 11:53)

PROTOCOL IDENTIFICATION

- Look for common binary/hex/ASCII values that are associated with specific protocols
 - Ex: 0x4500 marks the beginning of an IPv4 packet
- Use information in the encapsulating protocol
 - Ex: Byte 9 of the IP header indicates protocol, 0x06 corresponds with TCP
- Use port numbers for TCP/UDP
 - Ex: port 443 indicates TLS/SSL, check to see if packet is indeed encrypted
- Analyze the function of the src or dst server
 - Use IP address and do a WHOIS lookup
- Look for recognizable protocol structures
- Refer to RFCs

Handwritten notes on the slide: "IP (tcp/udp)", "SNMP", and "SN".

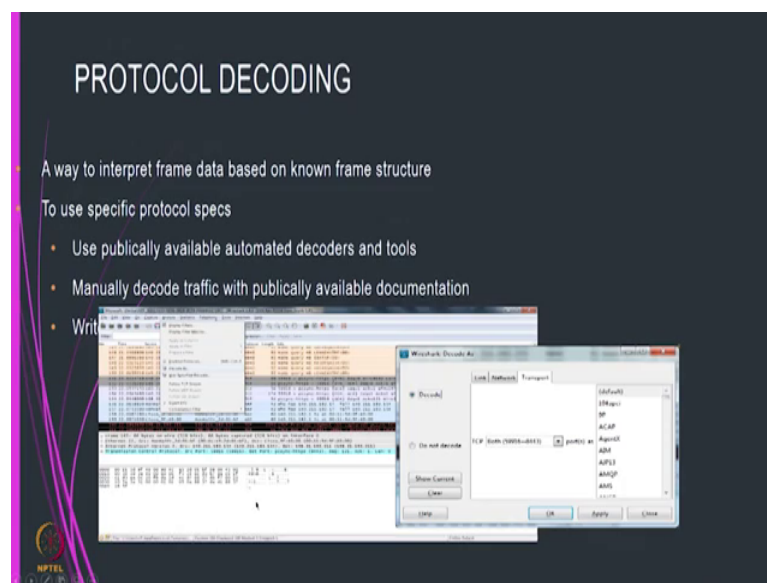
So, how do we usually identify protocols? Use of the protocols we will have some kind of signatures say for example, a marks like 0 x 4500 marks the beginning of an IPv 4 packet. I mean similarly a pdf document starts with a certain marker ok. So, in this way you would be able to identify what kind of protocol is going on in the network ok. So, sometimes for example, in SNMP do you they allow company specific management information? So, it will be then in that case you might have to identify that its a company specific information and then go to the company's website to find out what information they are presenting ok. So, here are some ways in which the protocol can be identified, one is use information the encapsulating protocol. So, for example, if the standard says that IP can encapsulate TCP or UDP this also, but for the time being.

So, now what will happen is that one of the fields will identify ok. So, within the IP packet one of the fields will identify whether it is a TCP packet or a UDP packet. I mean

you can observe it in wire shark I think we showed one example of a UDP packet in wire shark ok. The next one that I can do is I can look at port numbers. So, for example, port 443 is specifically SSH or TSL or SSL secure socket layer protocol then 53 is dns these are known as standard ports. So, you could using these kind of standard ports you could identify what is the kind of protocol packets that are flowing in the network. The other thing that you can do is you can use the IP address and you something known as who is lookup I mean. So, what you do is this is a public website ok. So, if I give a public the if I give the public IP, it will tell you what is its domain name for example, Google dot com it has a particular public IP etcetera or finally, if you cannot identify any of these please take a look at the RFCs because some people implement protocols which are not yet a standard, but they find it convenient to use so.

So, these are all some of the ways by which you can identify protocols; obviously, I mean there are all some known methods, you could also try your own innovation to find out other methods of identifying the protocols.

(Refer Slide Time: 14:28)

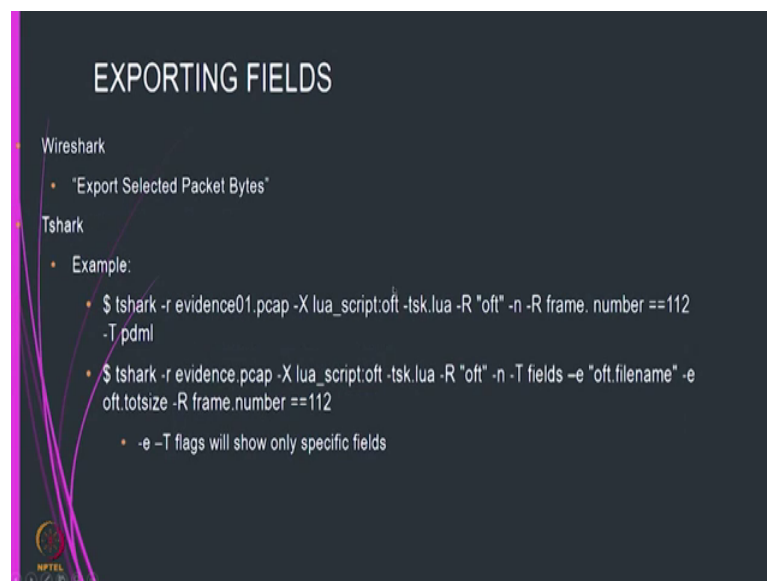


Now, once I identify the protocol, if the standard is known then I have no problem in decoding it ok. For example, a protocol sees the wire shark itself has a decoder. So, here if you see this we have a way by which you can decode the protocols. So, if you look at this you go to analyze and then you go to decode as, then it tells you I mean how you need to decode this protocol etcetera. So, you can use the publicly available automated

decoders and tools, and manually decode traffic with publicly available documentation. Now what can you automate the process yes you can. So, you this is where learning some of the programming languages helps you or you could make yourself familiar with python or ruby or so, so what you can do is if I get to know the publicly available information of a protocol.

So, how the protocol behaves, then I should write a protocol interpreter in one of these languages. If you write it in lua you can just straight away attach it to wire shark and then in that way it can be used and you can also publish it as a documentation so that others can also use it. So, this is something you might have to work on when you are doing when you come across each problem means we cannot generally say that you got to write these things. Well well known protocols are documented already by wire shark so, we can use this.

(Refer Slide Time: 15:53)



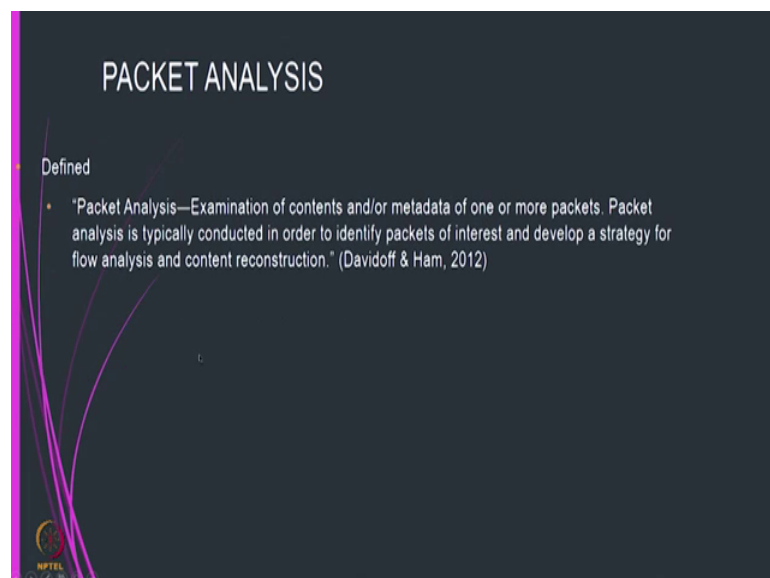
One of the things this after this, that you can have to up download specific selected package bytes.

So, this is just to establish something for example, if I carry a document then I have to download all the package that carry a document. For example, I am sending a word document from our organization to some other place secretly I mean hopefully without any encryption I mean. So, once I do this, what will happen is that then, I have to select all those packets which carry this documentation ok. It will be as carried as a payload

and then I have to only select those kind of documents and then put it together. So, here you can either use wire shark to identify or you can use a long command and if you are a geek, then you try to use a long command line something like this and then translate it to PDML and from PDML you do lot of work so.

So, this kind of translation can be done with t shark for example, this is a I use the pcap file and then I used a lua script and then what I do is, I capture from a particular frame number and then write it into a PDML format. So, in this way once I write it in a PDML; then, I can write xml parser xml kind of parser to get the data and other information.

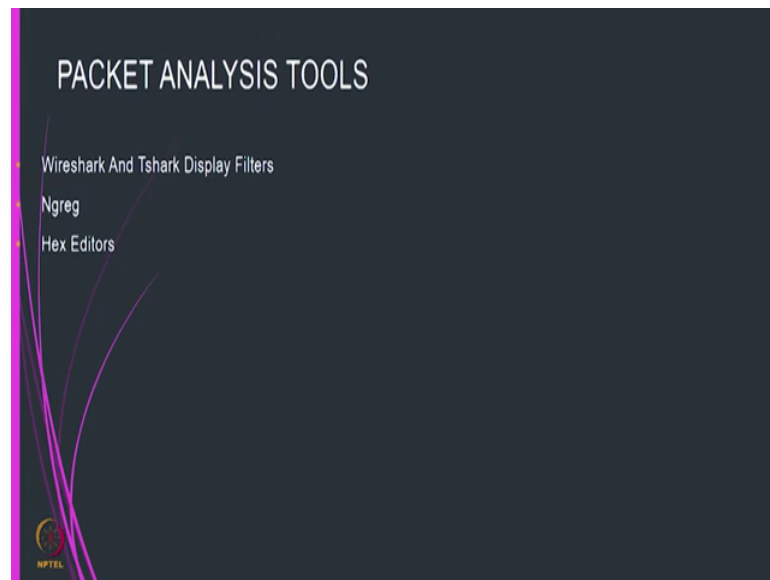
(Refer Slide Time: 17:15)



Now, after I do a protocol analysis I can also do a packet analysis ok. A packet actually goes into the contents of the packets ok. So, it is the examination of contents under metadata of one or more packets, the packet analysis is typically conducted in order to identify packets of interest and develop a strategy for flow analysis and contents reconstruction.

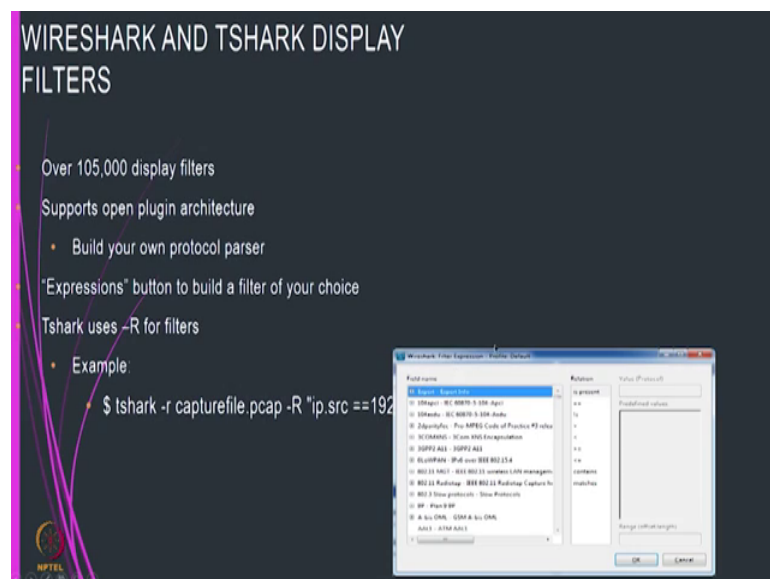
So, as I told you before suppose I send a word document the word document will not be sent in one packet. So, it will be split across many packets ok. So, once I identify that in one of the packets I have a word document, then I have to find out what are the packets that are sent before and then the packets that are sent after, and then I try to match all these data together and then try to recreate the word document that was sent. So, this is an example of a packet analysis.

(Refer Slide Time: 18:04)



Similarly, even in packet analysis we can use wire shark or a t shark, other than this you may doubt use some kind of hex editors, there are many of them available with Kali Linux and you can also use something like n grep etcetera.

(Refer Slide Time: 18:20)

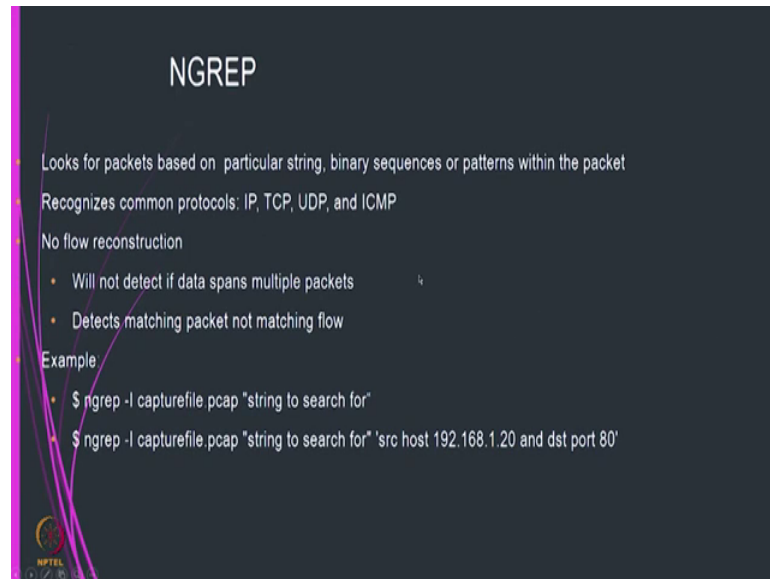


So, here is an example of how wire shark filter expression can be done. So, if you look we had looked at filters we told you that we can apply filters, and if you look at wire shark it has given almost I mean for almost many of the standard protocols it has display filters ok. If since wire shark is also open plug in architecture, you will actually be able

to build your own protocol parcel and you can also build expressions of your choice etcetera.

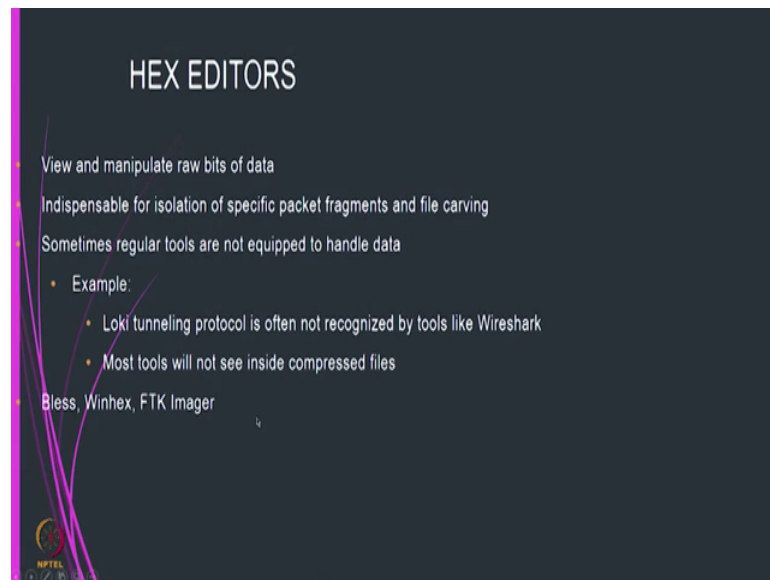
So, here is an example of how to use t shark also. So, here it tries to identify all those packets with IP source address is 192.168.1.something ok.

(Refer Slide Time: 18:58)



So, as I told you n grep can also be used a looks for a package based on pattern similar to grep, I think if many of you are familiar with grep f crappy grep and all those things n grep is used to work on a pcap files ok. So, here is just like grep if you are familiar with grep you should be able to understand about n grep it recognizes a common protocols and it tries to extract the strings that are present in any of this non protocols coming back to hex editors ok.

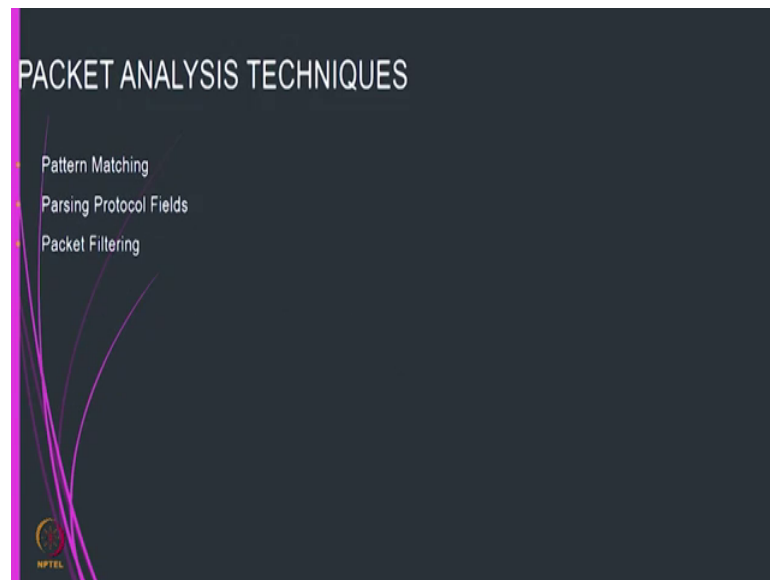
(Refer Slide Time: 19:26)



So, there are these types of hex editors that are available the hex editors actually manipulate this is this can be for some kind of advanced forensics because these hex editors work at the range of bits and bytes. So, you should want to do something known as a file carving. So, you might have to build this kind of hex elements put together and then if the packets are fragmented, then which is the header which is the byte and which hex byte should come here and so on.

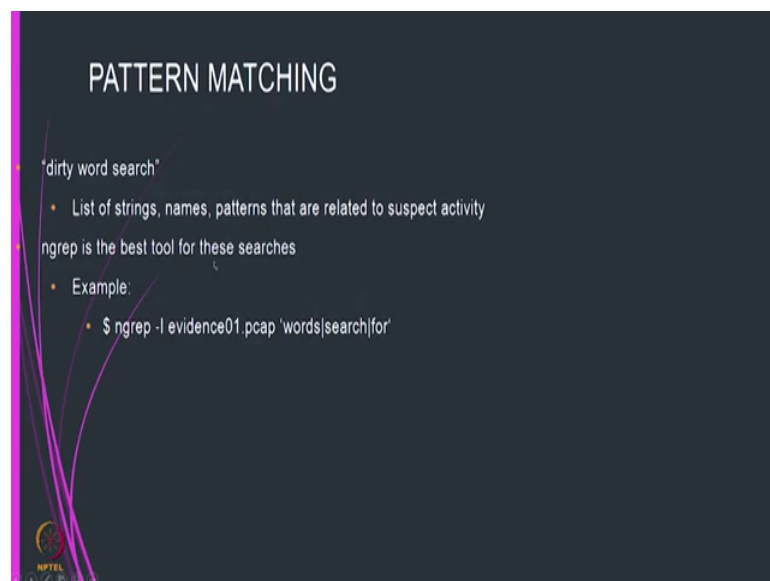
So, it is sort of detailed analysis for which you would be using all these things and one of the example is most tools will not see a inside a compressed file.

(Refer Slide Time: 20:13)



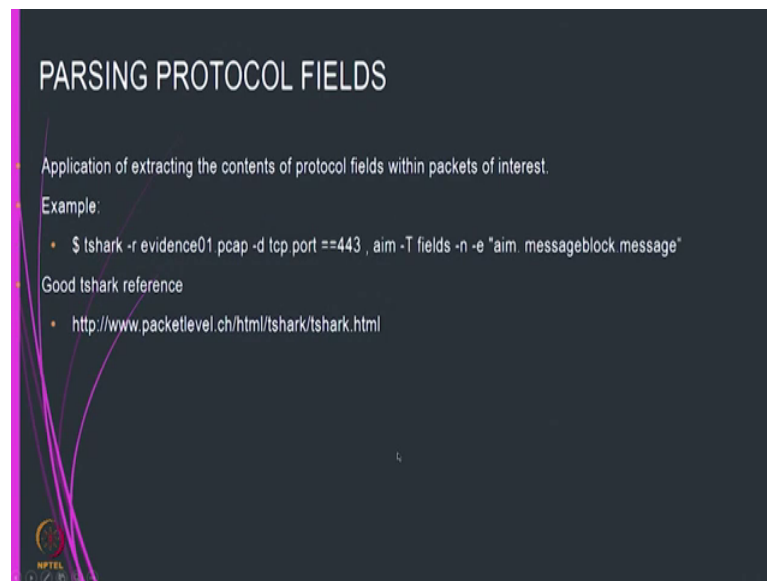
So, in that case you might have to look at this x then take the compress file out then decompress it and so on. Some of packet analysis techniques includes one is pattern matching and then parsing protocol fields and then packet filtering. So, pattern matching is similar to n grep where you search for a particular pattern, we also saw that how to parse protocols fields because protocols fields are given certain length for example, IP is 20 bytes header ipv 420 bytes header and so on and then you might have to filter the packets to identify what type of packet it is.

(Refer Slide Time: 20:42)



So, here is an example of a pattern matching for example, I can use the n grep then I can look at the strings, the names patterns most of you are familiar with grep I think you should understand what this is because n grep you take the evidence file and then search for the words that you want to say and the output will be displayed particular packets could be due days whether it is available.

(Refer Slide Time: 21:01)



So, similarly for parsing protocols fields you can use t shark or you can use wire shark straight away. So, with t shark you might have to give all these command line parameters etcetera.

(Refer Slide Time: 21:13)

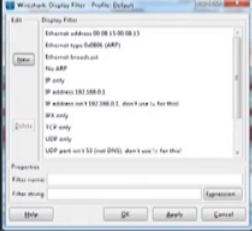
PACKET FILTERING

...the art of separating packets based on the values of fields in protocol metadata or payload.'
(Davidoff & Ham, 2012)

Use tcpdump with a BPF filter to dump out suspicious conversations

- Example using IP addresses
 - `$ tcpdump -s 0 -r evidence01.pcap -w evidence01-talkers.pcap 'host 64.12.24.50 and host 192.168.1.158'`
 - type EN10MB (Ethernet)

Use Wireshark



And with packet filtering its art of separating package based on the values in the field in the protocol metadata or payload ok. One of the tools we can use is TCP dump with Berkeley packet filter. I think we discussed this in the one of the previous sessions. So, TCP dump can be used to dump the packet and the best tool we just use wire shark ok.

So, this brings us to end of this module where we are looking at packet analysis.

Thank you.