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

Lecture – 38 Wireshark Introduction

Welcome to this session on network security and forensics. Until now we have been talking about how to collect evidence. We saw that we might have to attach ourselves physically to the network, to collect evidence and if the evidences collected live; it is active evidence. If the evidences collected and stored or you take it from some storage medium, it is passive we can call it as a static or a passive kind of evidence.

Essentially we will look at one tool which is wireshark, which can be used to collect data packets, and if you remember the process what we have tried to do was that, we want to collect whatever is the data packet that is flowing in the network to which we are connected and then start gathering evidences. So this was one of our aims in this network forensics course.

So, what we will do now is, we will take a look at wireshark which is used for capturing the packets. This will a sort of hands on, so we are going to only give an introduction for about 15 to 20 minutes about this wireshark tool, so you need to work much more on various problems of how to collect data, how to collect series of data, how to identify the flows using wireshark. There are also other tools that such as tshark, but we will be mostly concentrating on wireshark because it is slightly easier to explain wireshark since it consist of a nice graphical user interface. This wireshark tool is actually found in Kali Linux which probably you are very familiar with and which you have been using. So, you need not separately install this and if you have Kali Linux you can start using wireshark. You can invoke wireshark by doing the following operation, you can do a pseudo command and then invoke wireshark.

(Refer Slide Time: 02:22)

	Recording	Ele Edit Jobs Help njsraman@njsraman-NP-Pro@ook-450-60:>5 sudo wjreshark
Pause recording		njsranan@njsranan-HP-PrcBook-450-G0:+5
Enable recording hotkey totkey: 🔋 Ctrl + [Shift + Alt + Super + R v	
Information tal time: 0.04.59 95 nr. 30.00 95 nr. 30.00 24 nr. 1546/340 24 nr. 1546/340 16 name: Worsh. 31.mp4 16 size: 12.MB 16 size: 26 kbb/s	Preview Preview frame rate: 10 C Note Previewing requires extra CPU time tespecially at high frame rates).	
	Log	
PageNecoro stantuutput) s PageRecord Stantingut) sta X11ingut Init] Using X11 sl PageRecord Stantingut) sta X11ingut IngutThread] Ing	zanteo surput arting input hared memory tred input ut thread started.	
Cancel recording	Save recording	

(Refer Slide Time: 02:23)

a Marcille Activity		Expression
Bell proc to Reportant	9	
Capture		
Walling for startup		
Finding tocal interfaces		
A second s		
Learn Guide - Wiki - Questions and Answers - Mailing Lists		
the count of the second second second		

So, this is in Ubuntu systems for other systems find you have to become a super user you become a super user and try to invoke wireshark. Otherwise what you can do is, you can get necessary permissions to access the Ethernet interface and then use wireshark. So once you invoke wireshark you get a nice GUI something like this.

(Refer Slide Time: 02:41)



So, if you look at this it says that error during loading and it says do file has been disabled due to wireshark running as super user ok. You can just ignore it for the time being; if you want to find out what this is this messages is just go to the web and check. Now our idea here is to capture the evidence so we are not bother about all these messages. So, what we will go do is, we will once we will just press and then once we go here; you see this is the main screen of wireshark and it might vary depending on what version of wireshark you use, but in general wireshark you will have these kind of screens, where you can just open the previously captured packet or you can capture the packets live ok.

So, in this case for example, we are using wireshark 2 dot 2 dot 6 probably you could get some other version of wireshark which is in your Kali Linux. Now the idea is we want to capture the data from Ethernet interface ok. So in at least in this case; if you have a wireless interface you could that wireless interface will also be shown here. So the idea is that we will go ahead and capture data; so what I do what we will do is, we will just go ahead and then double click on this eth 0 to capture the data. So here we are double clicked it. And once you are double clicked it you see that wireshark has started to capture the data packets.

(Refer Slide Time: 04:01)



Now if you look at this, the wireshark actually I mean it divides I mean one of the things that you should be careful is that whether I will get all these three partitions of the screens etcetera it depends on what is the configuration you are going to do for wireshark, we will look into it shortly.

But then in general; once you open wireshark, it will have three partitions the first partitions shows the packets, the second partition shows each of the packets and the third partition shows the values in hex. So if you look at this part, you see that its displays the number of packets that are collected; so and the number of packets that are displayed. So if you look at this since its live, you see that the packets are increasing will collect about 100 packets and then stop. So we can go ahead and stop wireshark after collecting about 100 packets.

So, now we have stopped wireshark and once we have stopped capturing ok; so this gives you if you look at this, we have different so it starts the packet numbers are sequence something like this, that is the sequence number 1, 2, 3, 4 5, 6, 7, 8, 9, 10 etcetera. This is the packets, the independent packets that are flowing and get captured by wireshark. The second one is its shows the time; time stats in the current time.

So, these 0th pack I mean the first packet is cart at the 0th time, I mean this can be changed so we will see how this can be changed; then it talks about the source address, the destination address, the protocol, what is the kind of protocol that it has captured,

then its finds out the length of the packet and then what is the info that it can decipher. See many a time we might have to do this hard work of inferring what is there inside the packet, but to some extent this wireshark provides you some basic information about the packets that it tries to interpret ok.

For far fyrer (g. Cyature Javáre Statutes Telephony Wretes Teles Telephony Brutes Teles Telephony Brutes Telephony Brutes

(Refer Slide Time: 06:13)

Now moving on, so if you look at this you have various options you can open files you can save as file so what we are going to do right now is, we just have to whatever packets that we have captured, we have captured about 100 packets. We will just store it in the slash tmp directory ok. So just I mean this just a experiment purposes.

So, this is how you actually acquire the data. So what we are doing right now is, there are various formats you can even zip it etcetera, but we will for the time being we will just capture it in p cap n g format. So I am just over writing a file, if you want I can create a new file or you can over write file. So in this case for example, I am over writing sample wireshark p cap n g So, I am just replacing that file and the data gets saved. So, what I can do right now is that, I can actually having saved the data, I can now remember we have told you that we should always work on a copy of the data and not on the real data, that is why we are actually storing the data.

So, then this is these are all the packets that this guy captured. So as I told there is time field, source filed, destination protocol, length and info etcetera ok. The protocols so one of the things that you should be aware is that as network forensic expert, we should be

aware of many of the protocols that are there in internet. We will tell you I mean where this information of the protocols can be found. One of the advantages of using wireshark is that, you can actually if for example, the protocol is not a standard protocol; that is proposed by intent engineering task force and it is a proprietary protocol. If you get information about this proprietary protocol, you can actually writes scripts in a language called lu a; Lua and then you can actually integrated with wireshark.

So, many a time you might have to do this, because while doing forensic you might have to do lot of scripting and that is one of the reasons in the previous course we thought about one scripting language, which is the shell scripts. You might have to writes code in python or ruby or I mean the Lua programming language anything I mean you might have to be familiar with at least one of these programming languages. So that you can write your own protocol interpreter and integrated with wireshark ok. We will see how these things are done, but not in detail, but we will be doing some case studies and we will we shows how this things are done ok.

Now, what is tells you that, you should be aware of the almost all the protocols that are there in the internet and which is quite impossible. So you should have the ability to understand the protocol and then try to write scripts, whereby the protocols can be interpreted ok; that is more important.

So, in this case for example, we can see a RIP version 1 protocol and RIP version 2 protocol and as I told you this the second window here, the second partition here actually shows what is being carried by this packet. So if you look at this it has a Ethernet frame, and it has a so it has a internet protocol version 4. So IP is being used and if you see this is the broadcast address for the subnet. So there is the router that is broadcasting its routes and it uses the use UDP and it uses the routing information protocol. One of the things that you should note in this is that if for example, I just open this routing information protocol, now I can go on expanding this routing information protocol is that if you go to the corresponding RFC which that the current RFC numbers usually changes as they make updates to protocols.

So, you can just go to IETF website and then check what is the current RFC for this. Now in that they would have given a packet format for RIP. So, this so RIP requires and then there is a response and things like that; so this is RIP version 1 and then it tells you what are all the IP addresses that it is carrying essentially the routing table that it is carrying or the routes that are reachable. So you will gets this kind of format and wireshark helps you identify read this things in a much better way by interpreting it with the standard diagrams that is given in the RFC s ok.

(Refer Slide Time: 10:57)

4	Sample	eWireshark.pcapng	- + 1
File Edit View Go Capture Analyze Statistics Telephon	y Wireless Tools Help		
ք 🗏 🦉 😁 📑 🗋 🗙 🌀 🧣 🌩 🚆	i i i 🕺 📕 🔍	Q, Q, II	
Apply a display filter <ctrl></ctrl>		•	• Expression +
No. Time Source Destinatio	n Protocol	Length Info	
1 0.000000000 192.168.1.245 192.168	1.255 RIPv1	106 Response	
3 0.083890272 192.168.1.193 239.255.	255.250 SSDP	215 N-SEARCH * HTTP/1.1	
4 0.1965/9330 192.168.1.249 239.255. 5 0.225306716 D-L1nk 00:e7:45 Spanning	-tree-(for-brid. STP	213 N-SEARCH * HTTP/1.1 60 Conf. Root = 32768/0/00:21:91:00:e7:3c Cost = 0 Port = 0x8009	
6 0.901844576 fd00::ca9c:dcff:fea3:a1. ff02::fb	MONS	106 Standard query 0x0000 PTR 249.1.168.192.in-addr.arpa, "QM" question	
7 0.901885744 192.168.1.250 224.0.0. 8 0.902033782 192.168.1.249 224.0.0.	251 MONS	86 Standard query 0x0000 PTR 249.1.168.192.1n-addr.arpa, "QM" question 106 Standard query response 0x0000 PTR, cache flush church.local	
9 1 084948239 192 168 1 193 239 255	255.250 SSDP	215 N-SEARCH * HTTP/1.1	
10 1.098697130 192.168.1.121 192.168. 11 1 107483030 103 148 1 346 316 316	1.255 UDP 166.366 CERB	86 57621 - 57621 Len=44 313 M.CEABCM + WTTD/1 1	
Frame 2: 166 bytes on wire (1328 bits), 166 bytes cap Ethernet II Sec. Eliferen 4, 06 57 (c.8.0) de et de 160 5	tured (1328 bits) on inter	Tace 0	
 Internet II, Src. Elitegro_ac.w.or (ca.wc.oc.ac.wo.d) Internet Protocol Version 4, Src. 192.168.1.245, Ost. 	224.0.0.9	aa se aa aa aa aa	
 User Datagram Protocol, Src Port: 520, Dat Port: 520 			
Destination Port: 520			
(Decision Status: Unverified) [Stream Locken: 1] • Rowling Information: Protocol			
1010 01 00 5+ 00 00 09 c8 9c dc A4 90 67 08 00 45 c0 1010 00 98 bf 79 40 00 01 11 16 c5 c0 a8 01 f5 e0 00 1020 00 09 02 08 02 08 21 85 25 c2 02 00 00 ff ff 1010 00 03 06 68 00 15 54 53 cc d0 00 00 00 00 00			
0040 00 00 00 02 00 00 00 00 00 00 00 00 00	mining manner		
0000 01 Fe 00 00 00 01 00 02 00 00 ac 10 b3 00 FF FF 0000 FF 00 00 00 00 00 00 00 00 01 00 02 00 00 c0 at			
0070 01 00 ff ff ff 00 00 00 00 00 00 00 00 10 00 02			
0000 00 00 c0 as 69 00 FF FF FF 00 00 00 00 00 00 00 00 00			
DOut On Od de de e9 f1			
0 7 (enath (udo length), 2 butes		Parkets 100 - Displayed 100 (100 0%)	Profile Default
a sector of the sector of the			

So here we are talking about the route matrix and if you click on one of this route matrix for example, that will actually go ahead and for example, I mean if you look at this we are looking at the RIP v 2 packet now; the RIP v 2 packet has more information than the RIP v 1 packet essentially I mean it has more information and what we do is now if you just click on one of the one of the what should I say the fields ok.

For example we are clicking on source port field 520. Now once you click on this field look at this here ok. So there is a blue color that is so he tells you that this hex value corresponds to this source port field. So either you can click on the values here, then the values will be highlighted or if click on the values here the values will be highlighted.

So, this tells you that in the packet, this field corresponds to the source port. Similarly I can identify other stuff for example, the destination port is also identified 0208 and then if you look at this I mean remember this is in hex, so 84 is 16 into 8 plus 4 that is so 132 so that is 128 plus 4 that is 132. So these are all hex numbers whereas, this guys clearly for easy readability it is printing it in decimal format. Of course, hex numbers are printed

as 0 x, so if you look at this the checksum is this and the checksum it says is unverified which is so similarly you can identify for internet protocol internet protocol we are using IPv 4 and looking at the IP address, we can say IPv 4 this is an IPv 6 packets ok.

SampleWireshark.pcapng	- +
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help	
🖌 🔲 🥂 🙆 💳 🗋 🎗 🖉 Q 📥 🛎 🌋 📥 🧮 🖉 🖉 🔲 🔲 Q Q Q 💷	
Apply a display litter «Ctrl-/»	 Expression
a. Time Source Destination Protocol Length Info	
1 0.000000000 192.168.1.245 192.168.1.255 RIPv1 106 Response	
3 0.083890272 192.168.1.193 239.255.255.250 555P 215 H-SEARCH * HTTP:	4.1
4 0 196579330 192 168 1 249 239 255 255 250 SSDP 213 M-SEARCH * HTTP/	/1.1
5 0.225306716 D-L1nk_00:e7:45 Spanning-tree-(for-brid. STP 60 Conf. Root = 327 6 0.001844536 fabb - robr deff fact at ff02 - fb 8065 106 Standard marks	768/0/00:21:91:00:e7:3c Cost = 0 Port = 0x8009 0x6000 PTR 340 1 168 193 in addr area, 70M* cumiling
7 0 901885744 192 168 1 250 224 0 0 251 KONS #6 Standard query C	0x0000 PTR 249.1.168.192.1n-addr arpa, "QM" question
8 0.902033782 192.168.1.249 224.0.0.251 MDNS 106 Standard query r	response 0x0000 PTR, cache flush church.local
9 1.084948239 192.168 1.193 239.255.250 550P 215 M-SEARCH * HTTP/ 10 1.084948239 192.168 1.131 193 148 1.555 100 46.53631	/1.1
11 1 107681038 183 188 1 348 318 318 318 318 365 365 365 CC10 313 M. CEAD'W + WTTD	4.1
Frame 2: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits) on interface 0	
<pre>titernet II, Src: Elitegro_a4.99.57 (C8.9c.0c.a4.99.67), Dit: IPv4ecast_09 (01.00.5e.09.00.09) Internet Protocol Version 4, Src: 102.148.1.245, Dat: 224.0.0.9</pre>	
0100 # Version: 4	
 Differentiated Services Field: Oxc0 (DSCP: CS6, ECN: Not-ECT) 	
Total Length: 152	
Identification: 0xbf29 (48937)	
Fragment offset: 0	
Time to live: 1	
Protocol: KDP (17) Mandar charkson: Briffit funlidation disablad]	
[Header checksum status: Unverified]	
Source: 192.168.1.245	
Destination: 224.0.0.9	
00 01 00 5e 00 00 09 c8 9c dc a4 90 67 08 00 45 c0^gE.	
10 00 98 07 29 40 00 01 11 10 05 00 88 01 15 60 00	
30 00 03 09 68 00 10 5a 43 3c cd (00 00 00 00 00hZC <	
40 0000 00 02 00 00 00 00 00 00 00 00 00	
00 01 fe 00 00 00 01 00 02 00 00 ac 10 b3 00 ff ff	
70 01 00 ff ff ff 00 00 00 00 00 00 00 10 00 02	
10 00 00 c0 a8 69 00 ff ff ff 00 00 00 00 00 00 001	
00 01 11 11 00 01 ec c6 e2 09 28 1e c6 d0 71 71	
The second secon	Roden 100 Protocol 100 (100 Ph) Roden
A shier se as the appendix the bacterial	Packets: 100 Unsprayed: 100 (100.0%) Prome: Default

(Refer Slide Time: 12:29)

So, we can identify what type of protocol that we have captured. So it also tells you that there is a spanning tree, so there is dealing link switches actually getting a spanning tree protocol STP trying to identify the spanning tree in order to avoid cycles etcetera. So, in this way you will be able to interpret many of the fields, but for this you might have to have your RFC next to you to understand what this field means and all those specific details ok. Now why are these specific details needed because you might have to present a report and in the report you might have to underline what are these fields and why this fields correlate as evidence ok.

(Refer Slide Time: 13:24)



So, similarly I mean like what we have done this can be done for each of the protocol that it carries. So in this case for example, if we I mean if we can identify we can just press our button on some particular hex value then we will able to see what that value is for example, 11 is actually 17 so it tells you that the IP is carrying a UDP protocol on top of it.

So, in this way we will be identify we will be able to each packet can be analyzed. Now let us try to see how this analysis can be done for example, let us say that I want to track what are all the data packets that are emanating from one machine particular machine. So in this case I can actually do a sorting based on the machine id. So if you look at this what I have done right now is I have just click on the source tab and its source according to the IP addresses. So in this way I will be able to capture what are all the data packets that are gone from a particular source machine. Now similarly I can do it so if you look this here you see that 192 dot 168 dot 1 dot 245 has communicated with 192 and sent a broadcaster request and it is also sent request to a RIPv 2 request to 224 dot 0 dot 0 dot 9 and so on.

So, in this way I will be so 249 is talked with so many other routers ok; now remember all these data can actually be presented in the form of graph and then we can later go on find out the connectivity of the network; who are all the people that I have try to contact and all those things. So this actually spans I mean this goes into something known as network discovery part. So, each data that is available here can be used for very many purposes, but we just wanted to present you that what is the kind of information that wireshark provides you.

Now, so you can sort according to the source address or you can sort according to the destination address or the protocol. So if you look at this we have this many types of SNMP protocols; so it SNMP protocol there is 0 dot 10 has generated lot of traps so and it also tells you at what time the straps are generated and so on.

So, and you can even sort according to the length of the packets; so it provides you huge amount of search opportunities sorting opportunities etcetera I mean the idea is that by see if you just put the data in a different format you may come to a as different conclusion ok. So this so because of this distance provides you all the facilities that are needed ok.

(Refer Slide Time: 16:10)

	4			SampleWireshark.pcapng		- + x
	file !	(dt	gen Go Capture Analyze Statistics Telephony Wireles	is Tools Help		
	4	Ш.,	Main Toolbar Eiter Toolbar	<u>.</u>		
	App	ely a c	Wireless Toolbar		Expressi	ion +
	No.	T	🖌 Status Bar	Protocol Length Info		
		1020	Packet List	RIPv1 106 Response RIPv2 166 Response		_
	e	10	✓ Packet Details	555P 215 H-56ARCH * HTTP/1.1		
		5 0	✓ Packet Bytes	-brid. STP 60 Conf. Root = 32768/0/00:21:91:00:e7:3c	Cost = 0 Port = 0x8009	
		6 0 7 0	Time Display Format	Date and Time of Day (1970-01-01 01:02:03.123456)	Ctrl+Alt+1 "QM" question "QM" question	
		80	Name Resolution	Year, Day of Year, and Time of Day (1970/001 01:02:03.123456)	th.local	
		10 1	geen +	Time of Day (01:02:03.123456)	Ctrl+Alt+2	
		Sourc		Seconds Since 1970-01-01	Ctrl+Alt+3	
		Ad	Expand All Ctrl+Right	Seconds Since Beginning of Capture	Ctri+Alt+4	
			Collapse All Ctri+Left	Seconds Since Previous Captured Packet	ColeAles	
	· Int	ernet	Colorize Packet List	LITC Date and Time of Day (1970-01-01 01 02 03 123456)	Ctri+Ait+7	- 1
		0100	Çolo ng Rules	UTC Year, Day of Year, and Time of Day (1970/001 01:02:03.123456)		
		Diffe	El Basine Columna	UTC Time of Day (01:02:03.123456)	Ctrl+Alt+8	
	-	Ident	Messe Cours Corrantere	Automatic (from capture file)		
	' '	rage	Chan Rucket in New Window	Seconds		1
		Time Proto	Beload as Eile formatil arthure Chile Shift at	Tenths of a second		
		1010	Reload Ctrl+R	Hundredths of a second		
	0000	01 0 00 c	9 26 21 00 00 01 11 1 1 1 1 0 1 0 at 01 c1 of ff	Milliseconds		
	0029	ff f.	a ee Ba 07 6c 00 b5 1a df 4d 2d 53 45 41 521.	Microseconds		
	0040	41 5	3 54 3a 20 32 33 39 2e 32 35 35 2e 32 35 35 05T: 23	Nanoseconds		
	0050	20 3.	2 35 30 3a 31 39 30 30 0d 0a 4d 41 4e 3a 20 .250.19 3 73 64 70 3a 64 69 73 63 67 76 65 72 22 0d "ssdp:d	Display Seconds with Hours and Minutes		
	0076	64 6	f 58 3a 20 31 0d 0a 53 54 3a 20 75 72 6e 3a . MX: 1. 9 61 6c 2d 6d 75 6c 74 69 73 63 72 65 65 6e dial-mu	. ST: urn: Ltiscreen		
् 5	0000	20 6	f 72 67 3a 73 65 72 76 69 63 65 3a 64 69 61 -org:se	vice dia		
	0010	20 4	7 6f 6f 67 6c 65 20 43 68 72 6f 6d 65 2f 36 Google	Chrone/6		
	0000	33 2 61 7	r 30 Ze 33 32 33 39 Ze 38 34 20 57 69 6e 64 3.0.323 7 73 0d 0a 0d 0a 0d 0a 0d 0a	9.84 W1/d		
	0 ?	He	ader checksum (ip checksum), 2 bytes	Packets: 100	Displayed. 100 (100.0%) Profile.	Default
-	_	-				

Now, what are the other things that this tools; so if you got each one of the tab that is given so if you look at this you see that there is a packet list, there is a packet details and packet bytes; now this correspond to packet list, this corresponds to packet details and this corresponds to packet bytes. So, in this way so you can enable or disable whatever you want so and if you look at the time field I set the field started with 0, but actually what you can do is if you go to this time display format then give this gives u lot of ways see what we have done a second since beginning of capture that is what we have done.

So, what you can do is we can even do with time of the day date and time of the day etcetera. So this provides you lot of facilities you can even do 10s of a sectioned 100s of a second and milliseconds and etcetera ok. So if you look at I mean please go through all these tabs and then try to find out you can see for easy readability this also you can also color the packets; so if you look at this we have just pressed on this tab to color the packets and you can also remove the coloring by clicking on it ok. So, now the colors have been improved.

(Refer Slide Time: 17:20)

4	SampleWireshark.pcapng	- + x
File Edit View Go Capture Analyze Statistics Telephony Wire	reless Jools Help	
A main Toolbar		
🖌 filter Toolbar		
Apply a C Wireless Toolbar		Expression +
No. Ti 🖌 Status Bar	Protocol Length Info	
1 Packet List	RIPv1 106 Response RIPv2 166 Resonne	
Packet Details	552P 215 H-5CARCH * HTTP/1.1	
Packet Bytes	SSOP 213 M-SEARCH * HTTP/1.1 For LL STP 50 Conf. Boot a 32146-00-01 St. 50-00 al 30 Cont. 6 Port a 0.0000	
6 0 Time Display Format	MDNS 106 Standard query 0x0000 PTR 249.1.168.192.in-addr.arpa, "QM" question	
7 0 Name Resolution	MDMS 86 Standard query 0x0000 PTR 249 1 168 192 In-addr arpa, "QM" question MDMS 106 Standard query resonnse 0x0000 PTR carbe flush rhwrch local	
- 91	SSDP 215 M-SEARCH * HTTP/1 1	
e goon	1000 BE CTETH , CTETH 1 gen AA	
· Source Egrand Southman South Fight		1
Ad Expand All Ctrl+Right	address (factory default)	
Collapse All Ctrl+Left	bs (unicast)	
Colorize Packet List	255.250	
0100 Coloring RdRs		
Diffe Colorize Conversation	· T)	
Total D Resize Columns Ctrl+Shift+R	R	
Flags Internals		
Frage Show Packet in New Window		
Proto Reload as File Format/Capture Ctrl+Shift+F	4	
Reload Chi+R		
0000 01 00 c0 11 11 10 10 10 10 10 10 10 10 10 10 10	····k os(E.	
0020 Ff fa ee 8a 07 6c 00 b5 1a df 4d 2d 53 45 41 52	1	
0030 43 48 20 2a 20 48 54 54 50 2f 31 2e 31 0d 0a 48 CH *	HTT P/1.1H	
1050 2e 32 35 30 3a 31 39 30 30 0d 0a 4d 41 4e 3a 20 .250.	1190 0. MAN.	
0060 22 73 73 64 70 3a 64 69 73 63 6f 76 65 72 22 0d "ssdp	prdi scover	
0010 64 69 61 6c 2d 6d 75 6c 74 69 73 63 72 65 65 6e dial-	-mul tiscreen	
2 0000 2d 6f 72 67 3a 73 65 72 76 69 63 65 3a 64 69 61 -org:	ser vice dia	
1000 20 47 6f 6f 67 6c 65 20 43 68 72 6f 6d 65 2f 36 6000	gle Chrone/6	
00c0 33 2e 30 2e 33 32 33 39 2e 38 34 20 57 69 6e 64 3.0.3	3239 .84 WInd	
C 7 Mander checksom in checksom) 3 hoter	Reckets 100 Desclared 100 (100 001)	Budda Datailt
W # Header Checksoni up CheckSonit, 2 Dytes	Packets 100 Displayed 100 (100.0%)	Frome Deraut

(Refer Slide Time: 17:25)

4		SampleWire	eshark.pcapng	- + x
file fdit Yew 🚾 Capture Analyze Sta	atistics Telephony Wireless To	ols Help		
Go to Packet	Ctrl+G	0 Q	Q. II	
Apply a display 👄 Next Packet	Ctrl+Down			Expression +
No. Time 🔶 Previous Packet	Ctrl+Up	Protocol Leng	th Info	
1 0.00000 First Packet	Ctrl+Home	RIPv1 RIP-1	106 Response	
r 3 0.00021	Ctrl+End	5569	215 M-SEARCH * HTTP/1.1	
4 0.19657 5 0.22530 Next Packet in Conversat	tion Ctrl+.	5550	213 H. SEARCH * HTTP/1.1	
6 0.90184 🌰 Previous Packet in Conve	ersation Ctri+.	MONS	100 Standard query 0x0000 PTR 249.1.108.192.in-addr.arpa, "QM" question	
7 0 90188 8 0 90203 T Auto Scroll in Line Cathor	ra	RONS	86 Standard query 0x0000 PTR 249.1.168.192 In-addr arps, "QM" question 106 Standard query campaign 0x0000 PTR carbs flush church local	
9 1.084946239 192 168 1.193	(19 (35 (35 (34	5500	215 M SEARCH * HTTP/1.1	
10 1.098697130 192.168.1.121	192.168.1.255	LOP	86 57621 - 57621 Leni44 513 M. 6640ru X. WTT0/1.1	
 Jaterine Protocol Writes 4, Sec. 192. 030 Version: 4 031 . Washer Length: 20 bytes 01freenablate Services 1810: 0400 10estification: 042222 (0761) 11agi: 000 11agi: 000 11agi: 010estification 11agi	168.1.199. Dst: 239.255.255.2 (5) (05CP: C50, (CN: Not-(CT) (1540)44)	50		
3000 01 00 5e 7f ff fa 20 25 64 36 7b e	eo 08 00 45 00 % 05(
0010 00 c9 26 21 00 00 91 11 00 12 c0 a	a8 01 c1 of ff	SI AR		
0030 43 48 20 2a 20 48 54 54 50 2f 31 2	2e 31 0d 0a 48 CH * HTT P/1	1		
0050 2e 32 35 30 3a 31 39 30 30 0d 0a 4	35 2e 32 35 35 051 239 250 4d 41 4e 3a 20 .250.190 0.1			
0060 22 73 73 64 70 3a 64 69 73 63 67 1	76 65 72 22 0d "ssdp:d1 sco	er".		
0060 64 69 61 6c 2d 6d 75 6c 74 69 73 6	63 72 65 65 6e dial-mul tis	reen		
2 0000 2d 6f 72 67 3a 73 65 72 76 69 63 0 0000 6c 3a 31 0d 0a 55 53 45 52 2d 41 4	05 3a 64 69 61 -orgiser vice 47 45 4e 54 3a 1:1. USE 8.4	dia		
10000 20 47 6f 6f 67 6c 65 20 43 68 72 6	of 6d 65 2f 36 Google Chri	ne/6		
0100 6f 77 73 0d 0a 0d 0a	20 57 59 56 54 3.0.3239 .84 0v5	#1/40		
0 7 Header checksum (ip checksum), 2 byte	es		Packets 100 Displayed 100 (100 0%)	Profile Default

So now, you can also travels from one packet to another packet using the go tab or we can also do a filters ok; this is very important because this wireshark dumps lot of packets and we do not need all the packets I mean what if I am going to analyze who is a person who is talking to another person why do you need RIP packet? So in order to do this what we can do is we can apply filters ok. This filter just tells you that only those packets that satisfy the filter condition or displayed the rest of the packets are not shown.

(Refer Slide Time: 17:31)



So, there are many ways to apply filter; one is you can go and type in this apply a display filter column ok, or you what you can do is you can on a selected packet you can just apply a filter, and in which case you see this it just tells you all the routing protocols like RIPv 1 and RIPv 2.

(Refer Slide Time: 18:10)



So, what we are saying in these cases if whatever is if the IP source address is 192 dot 168 dot 1 dot 245 just display those packets.

So, essentially we can know that this guy is a router and this is router trying to I mean people who are configured the router; now look at the information that we can get we can find out that the people have configurate both RIPv 1 and RIPv 2. I do not know for why they also done OSPF ok. So this tells you that this guy configured for OSPF and for example, if none of the other routers are there in OSP are used other using OSPF, you can actually disable this protocol otherwise this hello packets just occupy bandwidth.

So, even some kind of performance optimization can if information can be obtained by using wireshark ok. This is very important tool that the network administrators used and it also used very largely by forensic experts and if you want to clear of just press that cross button. So it clears off and then it again stats is playing almost all the packets. So, this is one of the ways by which you can actually capture the packets that are flowing in the network.

(Refer Slide Time: 19:34)

4	SampleWireshark.pcapng - + >
File Edit View Go Capture Constraints Telephony Display Fiters Display Fiter Macros	wreters look gep
Apply a dispury inter	Lind *) Expression *
No. Time 4 Source Apply & Infor 214 15/14/14/13 (216) Projectal and a State Projectal and a State 214 15/14/14/13 (216) Projectal and a State Environment and a State 217 15/14/13/16/14/13 (216) Environment and a State Environment and a State 217 16/14/16/14/14 (217) Environment and a State Environment and a State Environment and a State 218 16/14/14/14 (217) 10/14/14 (217) Environment and a State Environment and a State	Spectral per LM spec Statewerty Protocol 94 Statewort 4.00 Stateworty Protocol 94 Statewort 9 Statewort 95 Statewort 9 Statewort 96 Statewort 10 Statewort 96 Statewort 10 Statewort 96 Statewort 10 Statewort 96 Statewort 10 Statewort 96 Statewort 11 Statewort
. 0	nugor wanyski (freidry defnall) Address (nicast) Abi 144.1.255 Not-ECT)
	Na θ θ Na θ Na
• Header (hecksum (ip checksum), 2 bytes	Packets 100 Displayed 100(1000%): Dropped 6(00%): Load time 0.6.4 Profile Default

The next one, we can look at is so you can capture, you can do analysis, you can get some kind of statistics. And if you look at what is happening is that you can also look at for example telephony for telephony, what are all the certain kinds of protocols ok. So for telephony protocols you can actually specifically do the capture those particular packets ok.

(Refer Slide Time: 19:57)

4	Sample	Wireshark.pcapng	- + x
File Edit Yiew Go Capture Analyze	Atostics Telephony Wireless Tools Help		
S X 🗇 🚍 🛛 🖢 🖿 🔪	Capture File Properties Ctrl+Alt+Shift+C Resolved Addresses	Q. 11	
Apply a display filter «Ctrl-/»	Protocol Hierarchy		Expression +
No. Time • Source 28 6.136784541 192.168.1.165 29 8.223069126 0.LLrA.09.e7.45 30 7.23645750 192.168.1.108 31 7.709273155 192.168.1.245 32 7.95662266 0.0.0 33 8.215562981 192.168.1.108	Conversations Endpoints Packet Lengths policy Response Time Service Response Time OHCP 1800075 Statistics OHC ARC Programs 2 2098et Called Call Called Call Physe Cognition I Files Cognition I	905 Hdo 180 Organization LAN spic Discovery Protocol 60 Conf. Root = 22/26/2009 (21.91.00.07.30) 229 # SUACC = NTEV.1.1 94 PULP Conf. Transaction 10 Gen2(1978) 229 # SUACC = NTEV.1.1 221 # SUACC = NTEV.1.1 219 # SUACC = NTEV.1.1	
34 8.222312854 0.0148,09.2745 35 8.688193534 192.168.1.210 36 8.769496024 192.168.0.10 37 9.212823020 192.168.1.108 38 16.3135338 103.188.1.108		 60 Con7. Root = 32788/20/00/21:93:00:07/30; Cost = 0 Purt = 0x8009 60 IS54 = IS54 Lene8 154 Tray Iso 3.6 1.4 I.3183.1.1 1.3.6.1.4 I.3183.1.1.1 23 8.55864 * HTP/1.1 23 8.65844 * UTP/1.1 	
 Source: Llitegro_s4:90:67 (c8:% Address: Elitegro_s4:90:67 (c6:%) 		mitj	
 Different Protocol Weiland, J. St. 1 0100 -: Weiland : 4 0101 = Weider Length: 20 byl b)fferentlated Services Field: 0 Total Length: 92 Identification: Outch8 (56504) + Flags: 0x82 (Don't Fragment) 		,	
Time to live: 64 Protocol: UOP (17) Header checksum: Uxd7d) (validat:	HTTP2 Sametime		
0000 FF FF FF FF FF FF C8 9c dc a4 1	TCP Stream Graphs	,	
0020 01 FF 02 08 02 08 00 48 88 5e 0	UDP Muticast Streams		
0040 00 01 00 02 00 00 00 00 00 00 00 00 00 00 00	Web Statistics	All Addresses	
1050 00 00 00 00 00 10 00 02 00 00 (1050 00 00 00 00 00 00 00 00 00 01	Pve statistics	IP Protocol Types	
		Source and Destination Addresses	
0 7 Header checksum (ip.checksum), 2 b	ytes	Packets 100 Displayed 100 (100 0%) Dropped 0 (0 0%) Load tir	ne 0.0.4 Profile Default

You can do lot of statistics for example, I can find out the IPv 4 statistics for all address, then for telephony, then wireless, then you can have some ACL rules etcetera access control list etcetera.

(Refer Slide Time: 20:11)

				Sampl	leWireshark.pcapng	- + x	I
	dit Yiew Go Capture A	nalyze Statistics	Telephony Wireless Too	is Help			I
	Open	Ctrl+O	• * X * T	= 0	0.0.0		I
	Open Recent		- H - F - H	= •	ч.ч.Ш.		ł
1	Merge					Expression +	I
i.	Import from Hex Dump		Destination	Protocol	Length Info		
	Close	Ctrl+W	192.168.1.255	08-LSP-D.	188 Dropbox LAN sync Discovery Protocol		I
			spanning-tree-(for-brid. 239.255.255.250	5509	60 Conf. Koot = 32768/0/00.21.91.00.07.3C Cost = 0 Port = 0x8009 213 M-SEARCH * HTTP/1.1		1
	Saus As	(his thill at	24.0.0.5	0585	94 Hello Packet		I
	State Bo	011340(13	255.255.255.255	DHCP SSDD	342 DHCP Discover - Transaction ID 0xe3c31d70 213 M.GEARCH * WTTP/1 1		I
	File Set		spanning-tree-(for-brid.	STP	60 Conf. Root = 32768/8/00:21:91:00:e7:3c Cost = 0 Port = 0x8009		I
	Export Specified Packets		192.158.15.255	LOP	60 1534 - 1534 Leni8		I
	Export Packet Dissections		239.255.255.250	5509	213 M SEARCH * HTTP/1.1		L
	Export Packet Bytes	Ctrl+H	210 244 244 240	0010	515 M. READYW X WTTD/1 1		1
	Export POUs to File		199.67)				
	Export SSL Session Keys		Globally unique address	(factory	default)		ł
	Export Objects		Individual address (un)	cast)			I
	Riel	ChileR	45. Dst: 192.168.1.255				I
	par.	0.110					
	* 0	CINEQ	C58, ECN: Not-ECT)				
	Identification: Owdeb8 (54	6504)					I
	+ Flags: 0x02 (Don't Fragmen	nt)					I
	Fragment offset: 0 Time to live: 64						I
	Protocol: UDP (17)	_					I
	Header checksum: 0xd7d3 (v	ralidation disab	1ed]				1
1	100 FF FF FF FF FF FF CB BC	dc a4 99 67 08	00 45 c09	.E.			T
	10 00 5c 0c 08 40 00 40 11	88 9e 02 01 00	10 CO AN				I
1	130 60 60 60 60 60 60 60 60	00 00 00 00 00	00 00 00				
- 8	H0 00 01 00 02 00 00 c0 A8 H0 00 00 00 00 00 10 00 02	01 00 00 00 00 00 00 00 c0 a8 69	00 00 00				
R.	00 00 00 00 00 00 00 00 00	00 01					
- 51							
							I
							I
							l
0	Header checksum lip.check	isum), 2 bytes			Packets: 100 Displayed: 100 (100.0%) - Dropped: 0 (0.0%) - Load time	e 0.0.4 Profile Default	l

As I told you one of the things and after that you can go ahead and quit it. Now if you remember we had actually stored the packet now we showed only the lives. Now we can go and store the packet and then we know we knew that, we are storing the packet as p cap and if you double click on that, so you are able to retrieve all those packets whatever you have stored in the previous sessions. So this gives you a very good interface for capturing packets.

Thank you very much.