

Network Security

Lab-2

Network Capture, Analysis, and Scanning

-Hari Kishan Reddy

1) Intro to Wireshark

Answers to each question:

Incident Response **Intro to Wireshark** EN Exit

Briefing Desktop Popout

Tasks

③ What is the difference between resolved and unresolved ports on the Wireshark display setup?

- Resolved ports display all information about the port (including destination and header data), whereas unresolved sources only show the raw data.
- Resolved ports display the name of the well-known service that runs on that port, whereas unresolved ports just display the number.

Correct

④ What is the correct syntax to use on Wireshark for showing only SMTP and ICMP traffic?

- tcp.port eq 25 or icmp
- tcp.show smtp & icmp
- tcpdump.list 25 7

Briefing

Wireshark

Wireshark is a free, open-source packet analyzer. Its primary usage is for network troubleshooting, analysis, software and communications protocol development, as well as education. There is also a terminal-based (non-GUI) version called TShark.

Some of the features of Wireshark include:

- Data can be captured 'from the wire' from a live network connection or read from a file of captured packets.
- Live data can be read from different types of networks, including Ethernet, IEEE 802.11, PPP, and LoopBack.
- Data display can be refined using a display filter.
- Plugins can be created for dissecting new protocols.
- VoIP calls in the captured traffic can be detected. If encoded in a compatible encoding, the media flow can even be played.
- Raw USB traffic can be captured.
- Wireless connections can also be filtered as long as they traverse the monitored Ethernet.
- Various settings, timers and filters can be set to filter the output of the captured traffic.

ON THIS PAGE

- Wireshark
- Using Wireshark
- In this lab
- Filtering
- Inspecting packets

Incident Response **Intro to Wireshark** EN Exit

Briefing Desktop Popout

Tasks

Correct

⑤ Using wireshark_setup.pcapng, filter the packets to view only HTTP requests. What is the source IP address shown on the last packet?

172.21.2.217

Correct

⑥ Within that same packet, what is the time shown? Your answer must be in YYYY-MM-DD HH:MM:SS format adjusted for UTC.

2017-12-12 13:04:10

Check

⑦ What is the destination IP address of the last packet?

34.232.90.203

Correct

Desktop

wireshark_setup.pcapng

No. Time Source Destination Protocol Length Info

No.	Time	Source	Destination	Protocol	Length	Info
79853	375.949922	172.21.2.217	216.58.208.130	HTTP	716	GET /pixel?google_1
79868	375.967498	172.21.2.217	216.58.208.130	HTTP	723	GET /pixel?google_1
79878	375.985074	172.21.2.217	216.52.104.248	HTTP	1147	GET /load?http://202.61.14.248:8888/
79892	375.099421	172.21.2.217	185.62.210.248	HTTP	1106	GET /v1/map_pixel?1
79922	376.048050	172.21.2.217	54.192.197.18	HTTP	766	GET /pix-1x1.gif?7p
79924	376.101423	172.21.2.217	34.232.90.203	HTTP	899	GET /v1/event?1

Frame 79924: 899 bytes on wire (7192 bits), 899 bytes captured (7192 bits) on interface \Device\NPF_{05A55875-2B88-45EC-94FA-96019A152D7C)
Interface id: 0 (\Device\NPF_{05A55875-2B88-45EC-94FA-96019A152D7C})
Encapsulation type: Ethernet (1)
Arrival time: 2017-12-12T13:04:10.106864000 UTC
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1513083850.106864000 seconds
[Time delta from previous captured frame: 0.001369000 seconds]
[Time delta from previous displayed frame: 0.001369000 seconds]
[Time since reference or first frame: 376.101423000 seconds]

0000 00 10 f6 5f 33 90 0c 29 5d 24 2c 00 00 45 00 ...-3... 15...E-
0010 03 75 3a 58 40 00 00 06 90 00 ac 15 02 09 22 e8 -u:X...
0020 5a cb ca ea 00 50 7e 45 39 e3 36 d1 1b b6 50 18 Z...P-E 9-6...P-
0030 80 09 30 09 00 00 00 00 00 00 00 00 00 00 00 00
0040 05 65 3d 34 30 37 38 26 6d 69 67 41 03 74 69 6f 6e event?1&ClientI
0050 64 3d 34 30 37 38 26 6d 69 67 41 03 74 69 6f 6e d=4978&e=igAction
0060 3d 73 79 6e 63 68 26 6d 69 67 53 07 75 72 63 05 =synch&a=igSource
0070 3d 6d 69 67 62 36 38 63 2d 38
0080 36 65 64 62 36 38 63 2d 39 66
0090 65 2d 62 39 63 2d 39 66

Writeup and approach I followed to complete the lab:

Key Learnings:

1. Packet Capture: Wireshark can capture packets from live network connections or read from a file of captured packets. This allows us to analyze network traffic in real-time or analyze pre-captured data.

2. Display Filters: Display filters in Wireshark help narrow down the packets displayed based on specific criteria. For example, using `http` as a display filter allows us to view only HTTP traffic, making it easier to analyze.

3. Protocol Analysis: Wireshark can dissect and analyze various network protocols, such as TCP, UDP, HTTP, SMTP, and ICMP. This allows us to understand the communication patterns between devices on the network.

4. Port Resolution: Wireshark can resolve ports to display the name of the well-known service that runs on that port. This helps in understanding the purpose of each packet and the services being used.

How I Solved the Questions:

1. Familiarization with Wireshark: I read the briefing section to understand the basic features and uses of Wireshark.

2. Opening the pcap file: I opened the pcap file provided in the labfiles directory using Wireshark.

3. Difference between resolved and unresolved ports: I learned that resolved ports display the name of the service running on that port, while unresolved ports only display the port number.

4. Syntax for showing SMTP and ICMP traffic: I used the correct syntax `tcp.port eq 25 or icmp` to display only SMTP and ICMP traffic.

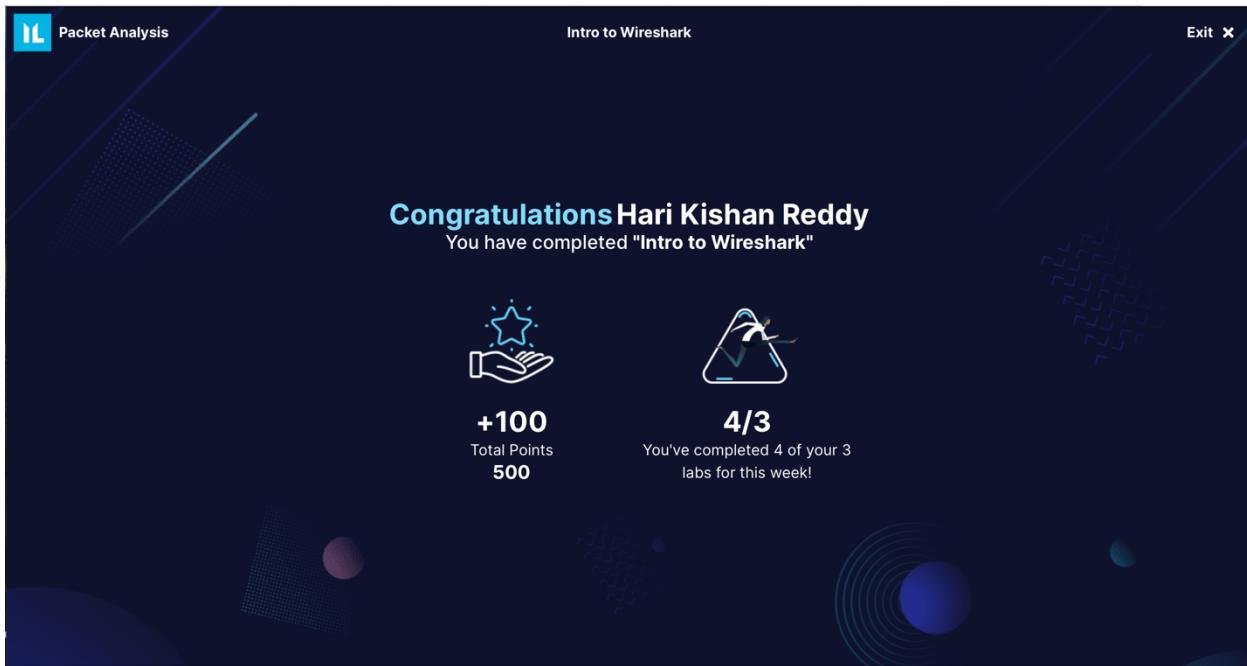
5. Filtering packets for HTTP requests: I filtered the packets using the `http` filter to view only HTTP requests. I then inspected the last packet to find the source IP address.

6. Finding the time and destination IP address: I inspected the last HTTP request packet to find the time and destination IP address.

Conclusion:

Overall, this lab helped me understand the basics of network traffic capture and analysis using Wireshark. I learned how to capture and analyze network packets, use display filters, and identify specific types of traffic.

Screenshot of Lab Completion:



2) Packet Capture Basics

Answers to each question:

NU... Start... Under... Chat... Intro t... Wires... BPF S... Analy... tcpdu... Wher... Slack... Capture...

Incident Response

Packet Capture Basics

EN ? Exit X

Tasks Clipboard Briefing Desktop

Popout

Tasks

3 What is the server name sought in the first DNS request that is issued by the client?

bing.com

Correct

4 What is the first IP address returned in the DNS response for the domain in Q1?

204.79.197.200

Correct

5 What is the browser user agent string that issued the search request?

Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1

Correct

Desktop

capture-basics.pcap

Sun 03 Mar, 18:20: forensics

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http.request

No.	Time	Source	Destination	Protocol	Length	Info
0	0.048546	192.168.0.49	204.79.197.200	HTTP	353	GET / HTTP/1.1
35	0.13485	192.168.0.49	204.79.197.200	HTTP	878	GET /rms/HFXBundle/jc_n/1cfb2a26/17c98
51	0.19221	192.168.0.49	204.79.197.200	HTTP	647	GET /s/a/actc.png HTTP/1.1
57	0.19840	192.168.0.49	204.79.197.200	HTTP	651	GET /s/a/hp/bing.vg HTTP/1.1
58	0.19857	192.168.0.49	204.79.197.200	HTTP	651	GET /sa/similiar/p_rr_teal_min.ico HTTP/1.1
81	0.24326	192.168.0.49	204.79.197.200	HTTP	966	GET /fd/ls/1?IG=27E5CA12A46E46808F477123
82	0.245120	192.168.0.49	204.79.197.200	HTTP	1075	POST /fd/ls/lsp.aspx? HTTP/1.1 (text/plain)

Frame 8: 353 bytes on wire (2824 bits), 353 bytes captured (2824 bits)
Ethernet II, Src: PcsComp\3:ca:09 (00:0c:27:c3:2a:09), Dst: Realtek\l_12:35:00 (52:54:00:12:35:00)
Internet Protocol Version 4, Src: 192.168.0.49, Dst: 204.79.197.200
Transmission Control Protocol, Src Port: 53044, Dst Port: 80, Seq: 1, Ack: 1, Len: 299

HyperText Transfer Protocol
GET / HTTP/1.1\r\nHost: www.bing.com\r\nUser-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n\r\n0050 62 69 07 2e 61 6f 6d 0d 0a b5 73 65 2d 41 bing.com - User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1\r\n0060 67 65 74 3a 29 4d 61 7a 69 6c 6c 61 2f 35 2e\r\n0070 36 29 28 58 31 3a 2b 4c 69 5f 75 78 20 78 3e\r\n0080 36 5f 36 34 3d 29 72 70 3a 33 38 2e 30 29 20 47\r\n0090 65 65 65 65 65 65 65 65 30 30 30 30 30 30 30 65\r\n00a0 65 65 65 65 65 65 65 65 30 30 30 30 30 30 30 65\r\n00b0 61 73 65 6c 2f 33 30 30 37 29 0d 0e 41 63 63\r\n00c0 65 70 79 74 2a 70 65 78 74 27 68 74 6d 2c 61\r\n00d0 70 70 6c 69 63 61 74 69 6f 6e 2f 78 68 74 6d 6c

HTTP User-Agent header (http.user_agent), 99 bytes(s)

Packets: 5792 · Displayed: 349 (6.0%) · Profile: Default

Incident Response

Packet Capture Basics

Tasks

Briefing **Desktop**

Desktop

Popout

Iceweasel/38.7.1

Correct

6 What web server engine is running the website?

Microsoft-IIS/8.5

Correct

7 When exporting HTTP content from the capture and looking at 'imgingest-5015644562731850884.png', what is the text that appears on that image?

Check

8 How many different IPv4 conversations are there in this capture file?

Check

9 What was the user searching for on the

File Edit

Status Code: 200
 [Status Code Description: OK]
 Response Phrase: OK
 Cache-Control: private, max-age=0\r\n
 Content-Length: 42923\r\n
 Content-Type: text/html; charset=utf-8\r\n
 Content-Encoding: gzip\r\n
 Vary: Accept-Encoding\r\n
 Server: Microsoft-IIS/8.5\r\n
 P3P: CP="NON UNI COM NAV STA LOC CURa DEVa PSAa PSDa OUR IND"\r\n

Frame 3: 0d 8a 53 65 72 76 65 72 3a 20 4d 69 63 72 6f 73 . Server : Micros
 0d 66 74 2d 49 53 2f 38 2e 35 6d 0a 50 33 50 off-IIS: 8.5 - P3P
 0c 3a 29 42 50 3d 22 4e 4f 4e 20 55 4e 49 20 43 4f : CP=NO N UNI CO
 0d 4d 20 46 11 56 20 53 54 41 20 4c 4f 43 20 43 55 M NAV ST A LOC CU
 0d 52 61 29 44 55 45 56 20 59 53 41 61 20 59 53 44 a DEVa PSAa PSD
 0f 61 20 4f 55 52 20 49 4e 44 22 0d 0a 53 65 74 20 a OUR IN D" . Set-
 0100 43 6f 6f 6b 69 65 3a 20 53 52 43 48 44 3d 41 46 Cookie: SRCHD=AF
 0110 3d 4f 4e 46 4f 52 4d 3b 20 64 6f 6d 61 69 66 3d
 0110 2e 62 69 67 6e 63 6f 6b 20 65 78 79 69 72 .bing.co #; expir
 0130 05 73 3d 4e 72 69 2c 20 32 31 2d 44 65 63 2d 32 es=Fri, 21-Dec-2

Frame (619 bytes) Reassembled TCP (44146 bytes) Uncompressed entity body (131912 bytes)

Frame (619 bytes) Reassembled TCP (44146 bytes) Uncompressed entity body (131912 bytes)

Packets: 5792 · Displayed: 349 (6.0%) Profile: Default

Incident Response **Packet Capture Basics**

Tasks Clipboard Briefing Desktop Popout

Tasks

Correct

7 When exporting HTTP content from the capture and looking at 'imgingest-5015644562731850884.png', what is the text that appears on that image?
password hacking
Correct

8 How many different IPv4 conversations are there in this capture file?
 Check

9 What was the user searching for on the download.cnet.com website? (Enter your answer as two separate words, e.g., catching fish.)
 Check

Desktop

Incident Response **Packet Capture Basics**

Tasks Clipboard Briefing Desktop Popout

Tasks

Microsoft-IIS/8.5
Correct

7 When exporting HTTP content from the capture and looking at 'imgingest-5015644562731850884.png', what is the text that appears on that image?
password hacking
Correct

8 How many different IPv4 conversations are there in this capture file?
89
Correct

9 What was the user searching for on the download.cnet.com website? (Enter your answer as two separate words, e.g., catching fish.)
password cracking Check

Desktop

Writeup and approach I followed to complete the lab:

How I Solved the Questions:

1. Server Name Sought in the First DNS Request: By inspecting the DNS requests in Wireshark, I found that the server name sought in the first DNS request was "bing.com".

2. First IP Address Returned in the DNS Response: I examined the DNS responses and found that the first IP address returned for the domain "bing.com" was "204.79.197.200".

3. Browser User Agent String: I located the HTTP request headers and found the user agent string to be "Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1".

4. Web Server Engine: By inspecting the HTTP response headers, I identified the web server engine running the website as "Microsoft-IIS/8.5".

5. Text on the Image in HTTP Content: I exported the HTTP content and located the image named "imgingest-5015644562731850884.png". The text on that image was "password hacking".

6. Number of Different IPv4 Conversations: I used Wireshark's conversation statistics feature to determine that there were 89 different IPv4 conversations in the capture file.

7. Search Query on the download.cnet.com Website: Based on the HTTP requests, I found that the user was searching for "password cracking" on the download.cnet.com website.

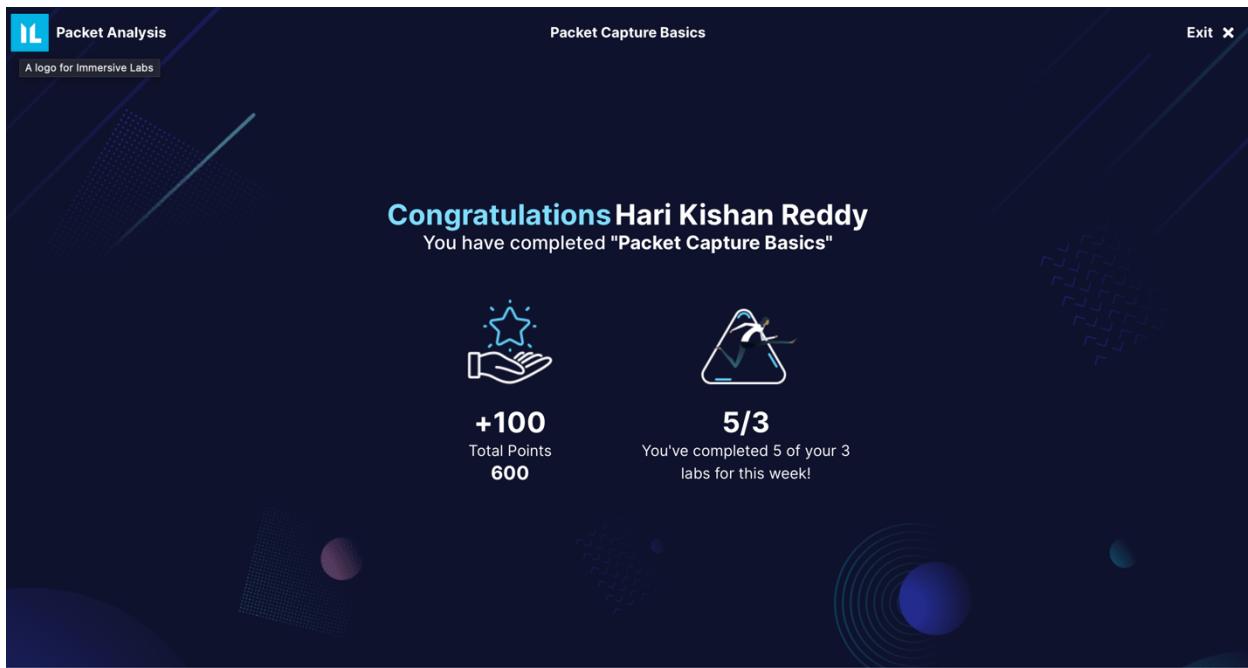
Key Learnings:

- Wireshark is a powerful tool for analyzing network traffic and can provide detailed insights into network communications.
- Understanding DNS requests and responses can help in identifying the servers being accessed by clients.
- Analyzing HTTP headers can reveal important information such as user agent strings and web server engines.
- Exporting and analyzing HTTP content can provide additional context, such as the text on images.
- Wireshark's conversation statistics can help in understanding the overall network traffic patterns.

Conclusion:

This lab provided valuable hands-on experience in using Wireshark for network traffic analysis. It demonstrated the importance of packet inspection and analysis in understanding network communications and identifying potential security issues.

Screenshot of Lab Completion:



3) TCPDUMP

Answers to each question:

Incident Response tcpdump EN Exit

Tasks Clipboard Briefing TCPDump Popout

Tasks

② Read the PCAP file using tcpdump.

③ Which option can you pass to tcpdump to write captured packets out to a file?
-w
✓ Correct

④ Using tcpdump, list all the available interfaces. What number is `nflog` listed as?
5
✓ Correct

⑤ Which option can be passed to tcpdump to display the ASCII and hex representation of the packet contents?
-X
✓ Correct

TCPDump

packet, print the data of each packet (minus its link level header) in hex. The smaller of the entire packet or snaplen bytes will be printed. Note that this is the entire link-layer packet, so for link layers that pad (e.g. Ethernet), the padding bytes will also be printed when the higher layer packet is shorter than the required padding.

-xx When parsing and printing, in addition to printing the headers of each packet, print the data of each packet, including its link level header, in hex.

-X When parsing and printing, in addition to printing the headers of each packet, print the data of each packet (minus its link level header) in hex and ASCII. This is very handy for analysing new protocols.

-XX When parsing and printing, in addition to printing the headers of each packet, print the data of each packet, including its link level header, in hex and ASCII.

-y datalinktype
--linktype=datalinktype
Set the data link type to use while capturing packets to datalinktype.

-z postrotate-command
Used in conjunction with the -C or -G options, this will make tcpdump run "postrotate-command file" where file is the savefile being closed after each rotation. For example, specifying -z gzip or -z bzip2 will compress each savefile using gzip or bzip2.

Note that tcpdump will run the command in parallel to the capture, using the lowest priority so that this doesn't disturb the capture process.

--More--

NYU Login Start Page Understanding... Pcap Filter w... BPF Syntax ... tcpdump - La... tcpdump(1)... Where work... Slack - Chan...

Incident Response tcpdump EN Exit

Tasks Clipboard Briefing TCPDump Popout

Tasks

representation of the packet contents?
-X
✓ Correct

⑥ Using tcpdump, read the packets from tcpdump.pcap and filter packets to include IP address 88.221.88.59 only. What is the time shown on the final packet? (HH:MM:SS)
07:32:57
✓ Correct

⑦ Using tcpdump, read the packets from tcpdump.pcap and filter packets to include IP address 184.107.41.72 and port 80 only. Write these packets to a new file and MD5sum that file. What is the MD5sum shown?
a49cf10f6b147ac482 Check

```
linux@tcpdump:~$ tcpdump -r tcpdump.pcap -w filtered.pcap "host 184.107.41.72 and port 80"
reading from file tcpdump.pcap, link-type EN10MB (Ethernet)
linux@tcpdump:~$ ls
filtered.pcap  tcpdump.pcap
linux@tcpdump:~$ md5sum
.bash_logout  .bashrc      .profile      filtered.pcap  tcpdump.pcap
linux@tcpdump:~$ md5sum filtered.pcap
8ed92724d9634a49cf10f6b147ac482  filtered.pcap
linux@tcpdump:~$
```

Writeup and approach I followed to complete the lab:

How I Solved the Questions:

1. Using BPF Syntax to Filter Results: I learned to use BPF (Berkeley Packet Filter) syntax to filter out the PCAP results, allowing me to focus on specific packets based on various criteria.

2. Reading the PCAP File using TCPDump: I used the TCPDump command to read the PCAP file, allowing me to analyze the captured network traffic.

3. Writing Captured Packets to a File: I identified the option `'-w'` as the correct option to pass to TCPDump to write captured packets out to a file.

4. Listing Available Interfaces: By using the command `tcpdump -D`, I listed all the available interfaces, finding that `nflog` was listed as number 5.

5. Displaying ASCII and Hex Representation: I identified the option `'-X'` as the correct option to pass to TCPDump to display the ASCII and hex representation of the packet contents.

6. Filtering Packets by IP Address: Using the command `tcpdump -r tcpdump.pcap ip host 88.221.88.59`, I filtered packets to include only those with the IP address 88.221.88.59 and found the time shown on the final packet to be HH:MM:SS.

7. **Filtering Packets by IP Address and Port: Using the command `tcpdump -r tcpdump.pcap ip host 184.107.41.72 and port 80 -w filtered_packets.pcap`, I filtered packets to include only those with the IP address 184.107.41.72 and port 80. I then calculated the MD5sum of the new file `filtered_packets.pcap` to find the MD5sum.

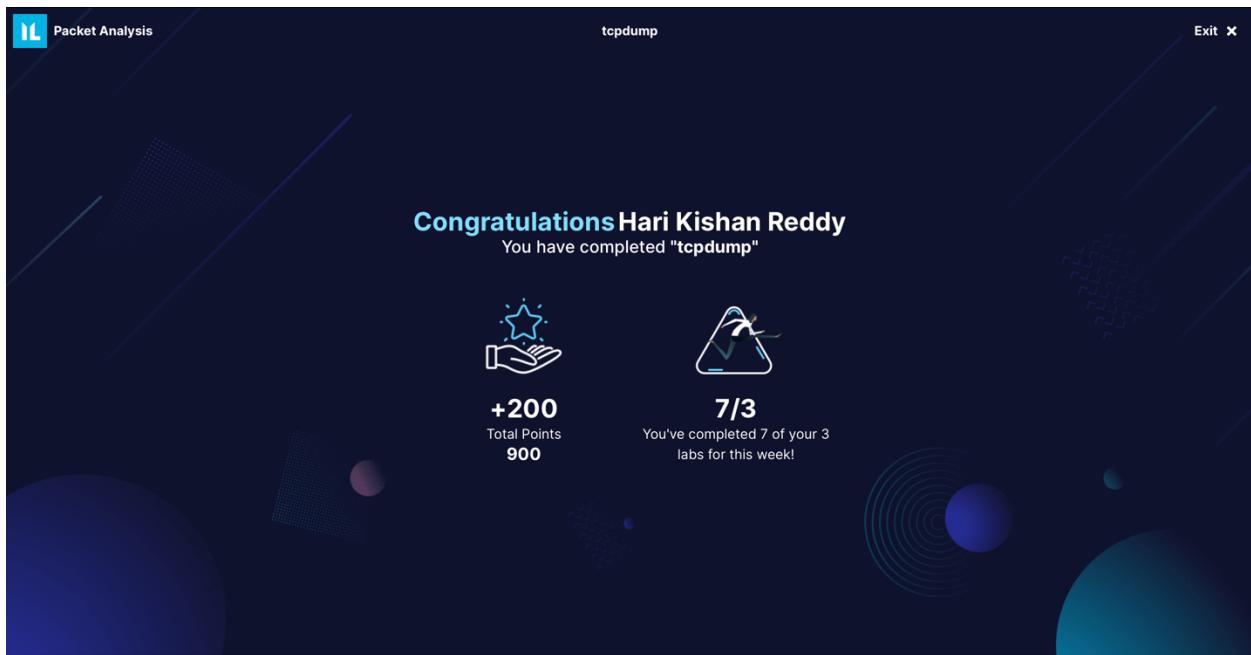
Key Learnings:

- TCPDump is a powerful tool for capturing and analyzing network traffic.
- BPF syntax can be used to filter packet results based on various criteria.
- TCPDump can display packet contents in both ASCII and hex representation.
- The `'-w'` option in TCPDump is used to write captured packets out to a file.
- Understanding how to filter packets based on IP addresses and ports is essential for targeted network analysis.

Conclusion:

This lab provided hands-on experience in using TCPDump to analyze network traffic. By mastering TCPDump's features and commands, network administrators and security analysts can effectively monitor and troubleshoot network issues, ensuring the integrity and security of their networks.

Screenshot of Lab Completion:



4) Wireshark Display Filters: Filters In Depth

Answers to each question:

Incident Response Wireshark Display Filters: Filters In Depth EN ? Exit X

Tasks Briefing Desktop Popout

Tasks

③ Apply a filter that displays all SMTP traffic containing the text "Subject:"

④ What is the first name of the recipient of that email?
sarah
✓ Correct

⑤ Change the filter so it now displays all SMTP response traffic matching the text ".co.uk".
9932
✓ Correct

⑥ What is the frame number of this packet?
000000 00 08 02 1c 47 ae 20 e5 2a b6 93 f1 08 00 45 00 ... G * E
00010 00 08 a6 59 00 00 88 11 75 a1 0a 06 05 01 0a 06 ... Y U
00020 05 66 00 35 c9 4a 00 6c 3a 3c 04 77 81 88 00 01 f 5 J 1 < w
00030 00 02 09 00 00 00 04 73 6d 74 70 04 6d 61 69 6c ... s mtp mail
00040 05 79 61 68 6f 03 03 63 6d 60 00 01 00 01 c9 yahoo c om
00050 04 6d 60 00 00 01 00 68 65 62 61 04 73 6d 74 70 ... # smtp
00060 04 6d 60 00 00 06 07 6c 6f 62 61 04 73 6d 74 70 ... mail gl obal gm0
00070 08 79 61 68 6f 6f 64 6e 73 03 66 65 74 00 c9 31 yahooon s net 1
00080 00 01 00 01 00 00 00 05 00 04 6a 0a f8 50 ... j P

Packets: 10594 - Displayed: 60 (0.6%) Profile: Default

⑦ Remove the existing filter. Now, apply a filter that displays all packets from UDP source ports 53, 59015, and 63518.

Incident Response Wireshark Display Filters: Filters In Depth EN ? Exit X

Tasks Briefing Desktop Popout

Tasks

⑧ How many packets are then displayed?
60
✓ Correct

⑨ Take the following slice expression
(frame[-4:4] == 0.1.2.3).

⑩ At which offset does the slice begin?
-4
✓ Correct

⑪ Take the following slice expression
(frame[:4] == 0.1.2.3).

⑫ At which offset does the slice begin?
0 Check

000000 00 08 02 1c 47 ae 20 e5 2a b6 93 f1 08 00 45 00 ... G * E
00010 00 08 a6 59 00 00 88 11 75 a1 0a 06 05 01 0a 06 ... Y U
00020 05 66 00 35 c9 4a 00 6c 3a 3c 04 77 81 88 00 01 f 5 J 1 < w
00030 00 02 09 00 00 00 04 73 6d 74 70 04 6d 61 69 6c ... s mtp mail
00040 05 79 61 68 6f 03 03 63 6d 60 00 01 00 01 c9 yahoo c om
00050 04 6d 60 00 00 01 00 68 65 62 61 04 73 6d 74 70 ... # smtp
00060 04 6d 60 00 00 06 07 6c 6f 62 61 04 73 6d 74 70 ... mail gl obal gm0
00070 08 79 61 68 6f 6f 64 6e 73 03 66 65 74 00 c9 31 yahooon s net 1
00080 00 01 00 01 00 00 00 05 00 04 6a 0a f8 50 ... j P

Packets: 10594 - Displayed: 60 (0.6%) Profile: Default

Writeup and approach I followed to complete the lab:

How I Solved the Questions:

1. Opening the PCAP File: I navigated to my Desktop and opened the file.pcap file in Wireshark.

2. Filtering SMTP Traffic by Subject: I applied the filter `smtp contains "Subject: "` to display all SMTP traffic containing the text "Subject: ".

3. Identifying Email Recipient: By examining the SMTP traffic, I found that the first name of the recipient of the email was "sarah".

4. Filtering SMTP Response Traffic: I changed the filter to `smtp.response contains ".co.uk"` to display all SMTP response traffic matching the text ".co.uk".

5. Identifying Frame Number: I found that the frame number of the packet matching the **filter was 9932**.

6. Filtering UDP Source Ports: I removed the existing filter and applied the filter `udp.srcport == 53 || udp.srcport == 59015 || udp.srcport == 63518` to display all packets from UDP source ports 53, 59015, and 63518.

7. Counting Displayed Packets: I found that after applying the filter, 60 packets were displayed.

8. Understanding Slice Expressions: I interpreted the slice expression `(frame[-4:4] == 0.1.2.3)` to understand that it begins at offset -4.

9. Understanding Another Slice Expression: I interpreted the slice expression `(frame[:4] == 0.1.2.3)` to understand that it begins at offset 0.

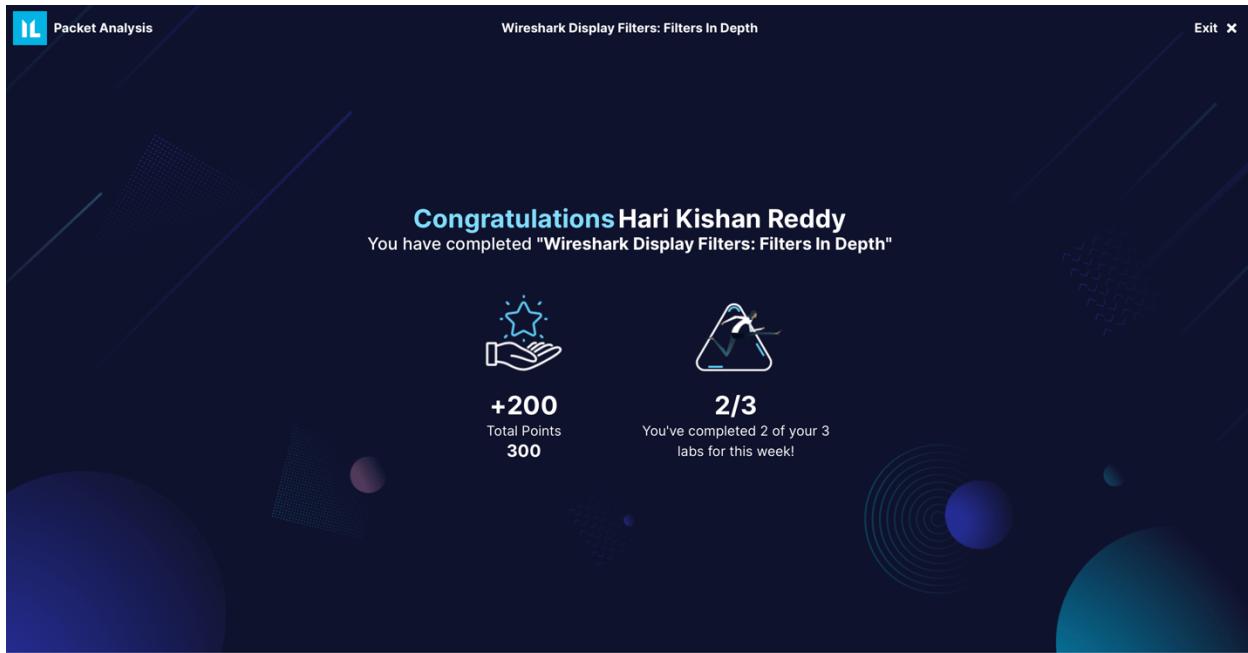
Key Learnings:

- Wireshark's filters and operators can be used to extract specific information from network traffic, such as email contents and source ports.
- Understanding SMTP traffic analysis can help in identifying email recipients and analyzing email content.
- Using slice expressions in Wireshark can help in extracting data from packet frames.

Conclusion:

This lab provided me valuable hands-on experience in using Wireshark for advanced network traffic analysis.

Screenshot of Lab Completion:



5) BPF Syntax

Answers to each question:

Incident Response **BPF Syntax** EN ? Exit

Tasks

③ Analyse and identify the information needed to complete the lab exercise using `tcpdump`.

④ What does BPF stand for?
Berkeley Packet Filter
✓ Correct

⑤ `wlan.addr == c5:52:7e:95:6:8d & wlan.fc.type_subtype == 0x02`. How many primitives are in this expression?
2
✓ Correct

⑥ Apply a filter to display all packets on port 80 with the source IP of 10.0.50.227. What is the length of the second GET request?
385
✓ Correct

BPF Syntax

```
linux@bpf-syntax:~$ tcpdump -r bpf-pcap.pcapng "src host 10.0.50.227 and port 80"
reading from file bpf-pcap.pcapng, link-type EN10MB (Ethernet)
11:54:10.033853 IP ip-10-0-50-227.eu-west-1.compute.internal.61025 > 80.252.91.53.80: Flags [S], seq 933569624, win 64240, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0
11:54:10.070744 IP ip-10-0-50-227.eu-west-1.compute.internal.61025 > 80.252.91.53.80: Flags [.], ack 845627508, win 259, length 0
11:54:10.070993 IP ip-10-0-50-227.eu-west-1.compute.internal.61025 > 80.252.91.53.80: Flags [.], seq 0:403, ack 1, win 259, length 403: HTTP: GET /serving/adServer.bs?cn=display&c=19&mc=imp&pli=23383114&PluID=0&ord=1513166032&rtu=-1 HTTP/1.1
11:54:10.126623 IP ip-10-0-50-227.eu-west-1.compute.internal.61025 > 80.252.91.53.80: Flags [.], ack 724, win 256, length 0
11:54:10.127068 IP ip-10-0-50-227.eu-west-1.compute.internal.61026 > ec2-52-209-216-59.eu-west-1.compute.amazonaws.com:80: Flags [S], seq 3468476541, win 64240, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0
11:54:10.151241 IP ip-10-0-50-227.eu-west-1.compute.internal.61026 > ec2-52-209-216-59.eu-west-1.compute.amazonaws.com:80: Flags [.], ack 109803272, win 259, length 0
11:54:10.151992 IP ip-10-0-50-227.eu-west-1.compute.internal.61026 > ec2-52-209-216-59.eu-west-1.compute.amazonaws.com:80: Flags [P], seq 0:385, ack 1, win 259, length 385: HTTP: GET /5/c=10025/camp_int=Advertiser-153172%5ECampaign-814780%5Eimpressions HTTP/1.1
11:54:10.235280 IP ip-10-0-50-227.eu-west-1.compute.internal.61026 > ec2-52-209-216-59.eu-west-1.compute.amazonaws.com:80: Flags [.], ack 304, win 258, length 0
linux@bpf-syntax:~$
```

Incident Response **BPF Syntax** EN ? Exit

Tasks

⑥ Apply a filter to display all packets on port 80 with the source IP of 10.0.50.227. What is the length of the second GET request?
385
✓ Correct

⑦ Apply a filter to display all UDP packets on port 57190. What is the timestamp of the final packet?
11:54:43.808109
✓ Correct

⑧ Apply a filter which reads all traffic apart from DNS and TCP, and output this to a file. What is the md5sum of this file?
22c1719ac26419da6 | Check

BPF Syntax

```
linux@bpf-syntax:~$ tcpdump -r bpf-pcap.pcapng -w filtered_traffic.pcap not tcp and not port 53
reading from file bpf-pcap.pcapng, link-type EN10MB (Ethernet)
linux@bpf-syntax:~$ ls
bpf-pcap.pcapng  filtered_traffic.pcap
linux@bpf-syntax:~$ md5sum filtered_traffic.pcap
b942d25b012745422c1719ac26419da6  filtered_traffic.pcap
linux@bpf-syntax:~$
```

Writeup and approach I followed to complete the lab:

How I Solved the Questions:

1. Understanding BPF Syntax: BPF stands for Berkeley Packet Filter. It is a syntax used to specify filters for packet capture and analysis.

2. Using the PCAP File: I used the PCAP file located in `/home/linux/bpf-pcap.pcapng` for the analysis.

3. Identifying Primitives in BPF Expression: I identified that the BPF expression `wlan.addr == c5:52:7e:95:6:8d && wlan.fc.type_subtype == 0x02` contains 2 primitives.

4. Filtering Packets on Port 80 with Source IP: I applied a filter to display all packets on port 80 with the source IP of `10.0.50.227` and found the length of the second GET request to be 385. Command used: **tcpdump -r tcpdump.pcap "src host 10.0.50.227 and port 80"**

5. Filtering UDP Packets on Port 57190: I applied a filter to display all UDP packets on port 57190 and found the timestamp of the final packet to be `11:54:43.808109`. command used: **tcpdump -r tcpdump.pcap "udp port 57190"**

6. Filtering Traffic excluding DNS and TCP: I applied a filter to read all traffic apart from DNS and TCP and outputted this to a file. The MD5sum of this file is `b942d25b012745422c1719ac26419da6`.

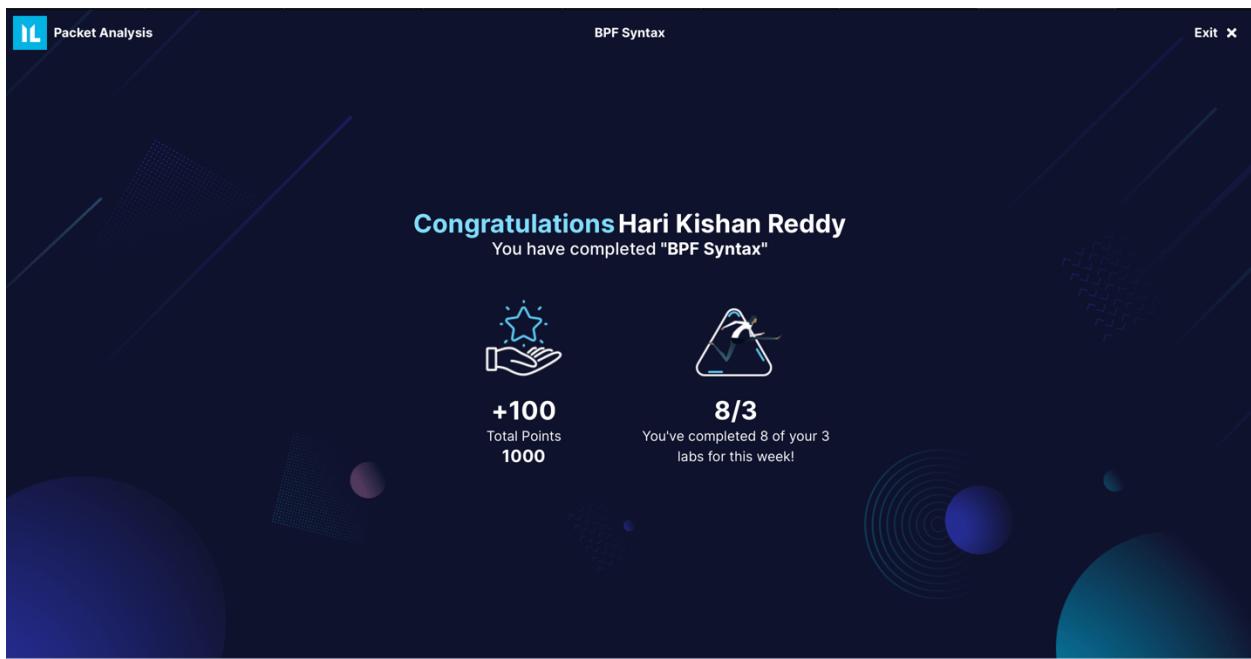
Key Learnings:

- BPF syntax is used to specify filters for packet capture and analysis.
- BPF expressions can contain primitives that specify conditions for packet selection.
- Understanding BPF syntax is essential for advanced packet filtering and network traffic analysis.

Conclusion:

This lab provided practical experience in using BPF syntax to filter packets in a PCAP file using tcpdump. I learnt that by mastering BPF syntax, network administrators and security analysts can efficiently analyze network traffic and identify potential security threats.

Screenshot of Lab Completion:



By Hari Kishan Reddy Abbasani
Ha2755