

TASK 2 REPORT: Packet Analysis using Wireshark

Objective

To capture and analyze network traffic using Wireshark and understand basic network protocols.

Tool Used

- Wireshark (Packet Capture and Analysis Tool)
- Network Interface
- Active Interface: Wi-Fi

Procedure

Opened Wireshark and selected the active network interface.

Started packet capture.

Generated network traffic by browsing websites.

Stopped the capture after sufficient packets were collected.

Applied protocol filters to analyze traffic:

DNS filter

TCP filter

ARP filter

UDP filter

HTTP filter Inspected

FILZA SAFDAR

DNS Traffic Analysis

DNS packets were observed resolving domain names into IP addresses.

DNS communication used UDP port 53.

Source and destination IP addresses were visible in the packet details.

This confirms domain name resolution before communication begins.

No.	Time	Source	Destination	Protocol	Length	Info
74	12.863441	192.168.100.37	192.168.100.1	DNS	74	Standard query 0xc95d HTTPS www.google.com
75	12.866995	192.168.100.37	192.168.100.1	DNS	74	Standard query 0x3e A www.google.com
76	12.873718	192.168.100.1	192.168.100.37	DNS	90	Standard query response 0x1ef A www.google.com A 142.250.187.36
77	12.875885	192.168.100.1	192.168.100.37	DNS	99	Standard query response 0x95d HTTPS www.google.com HTTPS
166	23.051099	fe80::71ef:6e30:549.. fe80::1		DNS	112	Standard query 0xd602 A storeedgefd.dsx.mp.microsoft.com
169	23.066843	fe80::1	fe80::71ef:6e30:549.. DNS		356	Standard query response 0x602 A storeedgefd.dsx.mp.microsoft.com CNAME storeedgefd.xbtservices.akadns.net CNAME storeedgefd...
201	23.847403	fe80::1	fe80::71ef:6e30:549.. fe80::1	DNS	96	Standard query 0x3e08 HTTPS web.whatsapp.com
202	23.847983	fe80::1	fe80::71ef:6e30:549.. fe80::1	DNS	96	Standard query 0x7e58 A web.whatsapp.com
203	23.849097	fe80::1	fe80::71ef:6e30:549.. fe80::1	DNS	109	Standard query 0x8d97 HTTPS _5222_https.web.whatsapp.com
208	23.849534	fe80::1	fe80::71ef:6e30:549.. fe80::1	DNS	96	Standard query 0x89c1 A web.whatsapp.com
205	23.854745	fe80::1	fe80::71ef:6e30:549.. DNS		202	Standard query response 0x6d8b HTTPS web.whatsapp.com CNAME mmx-ds.cdn.whatsapp.net SOA a.ns.whatsapp.net
208	23.859485	fe80::1	fe80::71ef:6e30:549.. DNS		149	Standard query response 0x7e50 A web.whatsapp.com CNAME mmx-ds.cdn.whatsapp.net A 57.144.149.32
209	23.860729	fe80::1	fe80::71ef:6e30:549.. DNS		190	Standard query response 0x897 No such name HTTPS _5222_https.web.whatsapp.com SOA a.ns.whatsapp.net
327	27.853169	192.168.100.37	192.168.100.1	DNS	79	Standard query 0x7efc HTTPS clients4.google.com
328	27.853644	192.168.100.37	192.168.100.1	DNS	79	Standard query 0x5a18 A clients4.google.com
332	27.863772	192.168.100.1	192.168.100.37	DNS	163	Standard query response 0x5a18 HTTPS clients4.google.com CNAME clients.l.google.com SOA ns1.google.com
333	27.863772	192.168.100.1	192.168.100.37	DNS	129	Standard query response 0xa18 A clients4.google.com CNAME clients.l.google.com A 142.250.203.14

> Frame 74: Packet, 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface Device\NPF_{2} [0000] 34:00:a3:af:3c:42 -> 00:00:c0:38:00:00 [0010] 00:00:c0:38:00:00 [01] 31:01:c0:a8:64:25 c0:a8:64:25:00:00 [0020] 00:00:c0:38:00:00 [02] 00:00:c0:38:00:00 [03] 00:00:c0:38:00:00 [04] 00:00:c0:38:00:00 [05] 00:00:c0:38:00:00 [06] 00:00:c0:38:00:00 [07] 00:00:c0:38:00:00 [08] 00:00:c0:38:00:00 [09] 00:00:c0:38:00:00 [0a] 00:00:c0:38:00:00 [0b] 00:00:c0:38:00:00 [0c] 00:00:c0:38:00:00 [0d] 00:00:c0:38:00:00 [0e] 00:00:c0:38:00:00 [0f]

TCP Traffic Analysis

TCP packets were observed ensuring reliable communication.

TCP uses a three-way handshake (SYN, SYN-ACK, ACK).

Source and destination ports were visible.

TCP ensures ordered and error-free data transmission

No.	Time	Source	Destination	Protocol	Length	Info
36	6.454789	192.168.100.37	57.144.149.32	TLSv1.2	123	Application Data
37	6.480241	57.144.149.32	192.168.100.37	TCP	60	443 → 50002 [ACK] Seq=1 Ack=70 Win=300 Len=0
38	6.659426	57.144.149.32	192.168.100.37	TLSv1.2	125	Application Data
39	6.784281	192.168.100.37	57.144.149.32	TCP	54	50002 → 443 [ACK] Seq=70 Ack=72 Win=514 Len=0
60	10.198091	192.168.100.37	162.159.133.234	TLSv1.2	100	Application Data
61	10.203730	162.159.133.234	192.168.100.37	TCP	64	443 → 50006 [ACK] Seq=1 Ack=47 Win=16 Len=0
62	10.446261	162.159.133.234	192.168.100.37	TLSv1.2	104	Application Data
63	10.498064	192.168.100.37	162.159.133.234	TCP	54	50006 → 443 [ACK] Seq=47 Ack=47 Win=511 Len=0
145	19.604601	192.168.100.37	4.213.25.242	TLSv1.2	97	Application Data
146	19.664666	4.213.25.242	192.168.100.37	TLSv1.2	228	Application Data
147	19.710182	192.168.100.37	4.213.25.242	TCP	54	49714 → 443 [ACK] Seq=44 Ack=175 Win=509 Len=0
176	23.078880	192.168.100.37	184.51.98.166	TCP	66	50012 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
177	23.248861	184.51.98.166	192.168.100.37	TCP	70	443 → 50012 [SYN, ACK] Seq=1 Ack=1 Win=64240 Len=0 MSS=1412 SACK_PERM WS=128
172	23.249017	192.168.100.37	184.51.98.166	TCP	54	50012 → 443 [ACK] Seq=1 Ack=1 Win=131072 Len=0
173	23.252794	192.168.100.37	184.51.98.166	TLSv1.2	253	Client Hello (SNI=storeedgefd.dsx.mp.microsoft.com)
179	23.450615	184.51.98.166	192.168.100.37	TCP	64	443 → 50012 [ACK] Seq=1 Ack=208 Win=64128 Len=0
180	23.450944	184.51.98.166	192.168.100.37	TLSv1.2	1470	Server Hello
181	23.451320	184.51.98.166	192.168.100.37	TCP	1470	443 → 50012 [ACK] Seq=1413 Ack=200 Win=64128 Len=1412 [TCP PDU reassembled in 184]

TASK:02

FILZA SAFDAR



ARP Traffic Analysis

ARP packets were captured showing requests and replies.

ARP Request packets asked “Who has this IP address?”.

ARP Reply packets returned the corresponding MAC address.

This process enables devices to communicate within the same local network.

No.	Time	Source	Destination	Protocol	Length	Info
9806	98.681600	Intel_ae:76:71	HuaweiTechno_af:3c:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
10087	127.177349	HuaweiTechno_af:3c:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.1
10888	127.177418	Intel_ae:76:71	HuaweiTechno_af:3c:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
10134	131.478196	DLinkInterna_10:51:..	Intel_ae:76:71	ARP	42	Who has 192.168.100.37 Tell 192.168.100.4
10135	131.478261	Intel_ae:76:71	DLinkInterna_10:51:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
10275	145.404269	DLinkInterna_10:51:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.4
10276	145.404329	Intel_ae:76:71	DLinkInterna_10:51:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
11070	150.319080	HuaweiTechno_af:3c:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.1
11071	150.319080	Intel_ae:76:71	HuaweiTechno_af:3c:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
17574	173.442674	HuaweiTechno_af:3c:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.1
17575	173.442700	Intel_ae:76:71	HuaweiTechno_af:3c:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
20380	180.683467	Intel_ae:76:71	DLinkInterna_10:51:..	ARP	42	Who has 192.168.100.4 Tell 192.168.100.37
20381	180.687601	DLinkInterna_10:51:..	Intel_ae:76:71	ARP	42	192.168.100.4 is at 1:c5:f2:b1:10:51:8c
21669	201.824678	DLinkInterna_10:51:..	Intel_ae:76:71	ARP	42	Who has 192.168.100.37 Tell 192.168.100.4
21670	201.824711	Intel_ae:76:71	DLinkInterna_10:51:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
22855	205.821240	DLinkInterna_10:51:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.4
22857	205.821510	Intel_ae:76:71	DLinkInterna_10:51:..	ARP	42	192.168.100.37 is at 00:c6:ae:76:71
2398	221.075944	HuaweiTechno_af:3c:..	Intel_ae:76:71	ARP	60	Who has 192.168.100.37 Tell 192.168.100.1

> Frame 11071: Packet. 42 bytes on wire (336 bits). 42 bytes captured (336 bits) on interface \Device\NPF_{0000...} 34 00 a3 af 3c 42 00 c2 c6 ae 76 71 08 06 00 01 4 ... <B... - va...

UDP Traffic Analysis

UDP packets were observed carrying lightweight traffic.

UDP does not establish a connection like TCP.

Source and destination ports were visible in packet details.

UDP is commonly used by DNS, streaming, and real-time applications.

No.	Time	Source	Destination	Protocol	Length	Info
11010	147.938837	57.144.149.32	192.168.100.37	QUIC	634	Protected Payload (K90)
11011	147.943828	57.144.149.32	192.168.100.37	QUIC	1274	Protected Payload (K90)
11012	147.944635	57.144.149.32	192.168.100.37	QUIC	80	Protected Payload (K90), DCID=ba00d84e1151a5d5
11013	147.945565	57.144.149.32	192.168.100.37	QUIC	1274	Protected Payload (K90)
11014	147.946452	57.144.149.32	192.168.100.37	QUIC	314	Protected Payload (K90)
11015	147.949949	57.144.149.32	192.168.100.37	QUIC	1210	Protected Payload (K90)
11016	147.952243	192.168.100.37	57.144.149.32	QUIC	77	Protected Payload (K90), DCID=ba00d84e1151a5d5
11017	147.989881	57.144.149.32	192.168.100.37	QUIC	90	Protected Payload (K90)
11062	148.417984	192.168.100.4	239.255.255.250	SSDP	175	M-SEARCH * HTTP/1.1
11077	150.523639	192.168.100.4	239.255.255.250	SSDP	143	M-SEARCH * HTTP/1.1
11092	151.155114	192.168.100.37	57.144.149.32	QUIC	120	Protected Payload (K90), DCID=ba00d84e1151a5d5
11095	151.178899	57.144.149.32	192.168.100.37	QUIC	90	Protected Payload (K90)
11096	151.178999	57.144.149.32	192.168.100.37	QUIC	186	Protected Payload (K90)
11097	151.179450	57.144.149.32	192.168.100.37	QUIC	90	Protected Payload (K90)
11099	151.181438	192.168.100.37	192.168.100.1	DNS	80	Standard query 0x2dd0 HTTPS beacons.gcp.gvt2.com
11100	151.181880	192.168.100.37	192.168.100.1	DNS	80	Standard query 0x156d A beacons.gcp.gvt2.com
11101	151.190382	192.168.100.1	192.168.100.37	DNS	178	Standard query response 0x2dd0 HTTPS beacons.gcp.gvt2.com CNAME beacons-handoff.gcp.gvt2.com
11102	151.190560	192.168.100.1	192.168.100.37	DNS	126	Standard query response 0x156d A beacons.gcp.gvt2.com CNAME beacons-handoff.gcp.gvt2.com A 142.250.202.99

> Frame 11062: Packet. 175 bytes on wire (1400 bits). 175 bytes captured (1400 bits) on interface \Device\NPF_{0000...} 00 00 c2 6e ae 76 71 1c 5f 2b 10 51 8c 08 00 45 00 ... vq... +Q - E...

> Ethernet II, Src: DLinkInterna_10:51:8c (1:c5:f2:b1:10:51:8c), Dst: Intel_ae:76:71 (00:c6:ae:76:71) | 0010 00 a1 00 00 40 00 02 11 63 a5 c9 a8 64 04 ef ff ... @... c... d... 0020 ff fa h5 a7 f6 00 8d fc 69 d4 2d 53 d5 a1 52 ... l... 1IM-SFAR

TASK:02

FILZA SAFDAR

Packet Detail Inspection

Individual packets were inspected to observe:

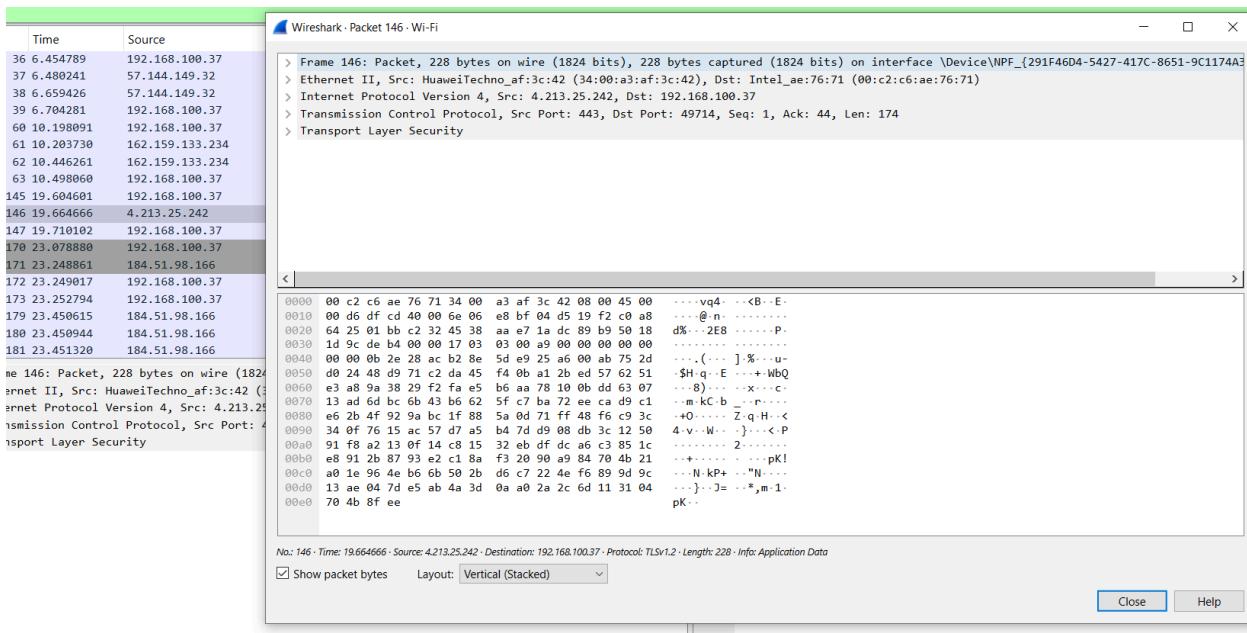
Source IP address

Destination IP address

Source and destination ports

Protocol type (TCP)

This helps understand how data flows between device



The screenshot shows a Wireshark capture window titled "Wireshark - Packet 146 - Wi-Fi". The packet list pane shows several network frames, with frame 146 highlighted. The details pane displays the following information for frame 146:

```

> Frame 146: Packet, 228 bytes on wire (1824 bits), 228 bytes captured (1824 bits) on interface \Device\NPF_{291F46D4-5427-417C-8651-9C1174A3
> Ethernet II, Src: HuaweiTechno_af:3c:42 (34:00:a3:af:3c:42), Dst: Intel_ae:76:71 (00:c2:c6:ae:76:71)
> Internet Protocol Version 4, Src: 4.213.25.242, Dst: 192.168.100.37
> Transmission Control Protocol, Src Port: 443, Dst Port: 49714, Seq: 1, Ack: 44, Len: 174
> Transport Layer Security

```

The bytes pane shows the raw hex and ASCII data for the selected TLS handshake message. The status bar at the bottom indicates: "No. 146 · Time: 19.664666 · Source: 4.213.25.242 · Destination: 192.168.100.37 · Protocol: TLSv1.2 · Length: 228 · Info: Application Data".

HTTPS Traffic Analysis

HTTPS traffic was observed during packet capture; however, the data payload was encrypted. Due to encryption, application-layer content was not readable. This demonstrates how HTTPS provides secure communication by protecting data confidentiality.

TASK:02

FILZA SAFDAR

PROTOCOLS

The following protocols were identified during packet capture:

Protocol

Purpose

DNS

Resolves domain names to IP addresses

ARP

Maps IP addresses to MAC addresses

TCP

Provides reliable, connection-oriented communication

UDP

Provides fast, connectionless communication

HTTPS

Ensures secure encrypted data transfer

TCP Three-Way Handshake

The TCP three-way handshake establishes a reliable connection between two devices.

Steps:

SYN – Client sends a synchronization request.

SYN-ACK – Server acknowledges and sends its own synchronization.

ACK – Client acknowledges, and the connection is established.

This process ensures both devices are ready to transmit data reliably.

Conclusion

The packet analysis demonstrated the functioning of multiple network protocols. ARP handled IP-to-MAC resolution within the local network, UDP enabled fast data transmission, DNS resolved domain names, and TCP ensured reliable communication. Wireshark proved effective for analyzing real network traffic.

GITHUB:https://github.com/filzasafdar8-netizen/CODEINTERN_TASKS

TASK:02