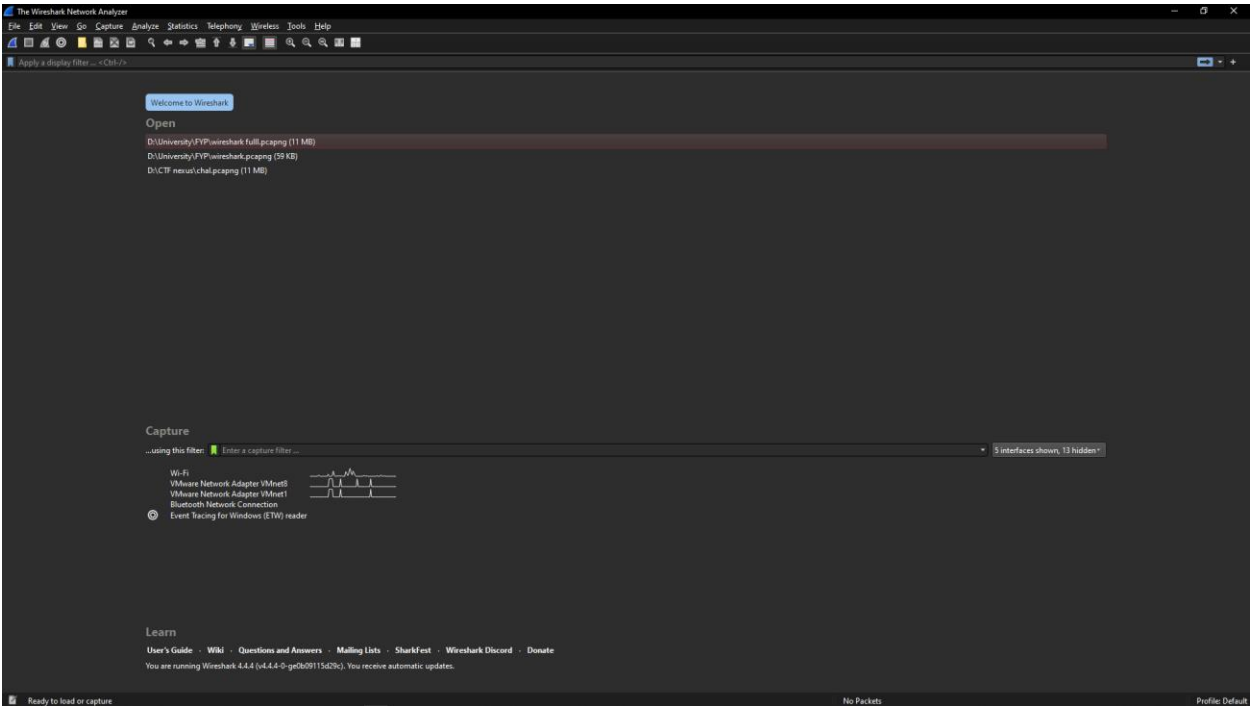


# Task 1: Analyze Network Traffic

Monitor and analyze network traffic using a tool like Wireshark.

Wireshark is opened



Lets start network traffic capture of Wi-Fi

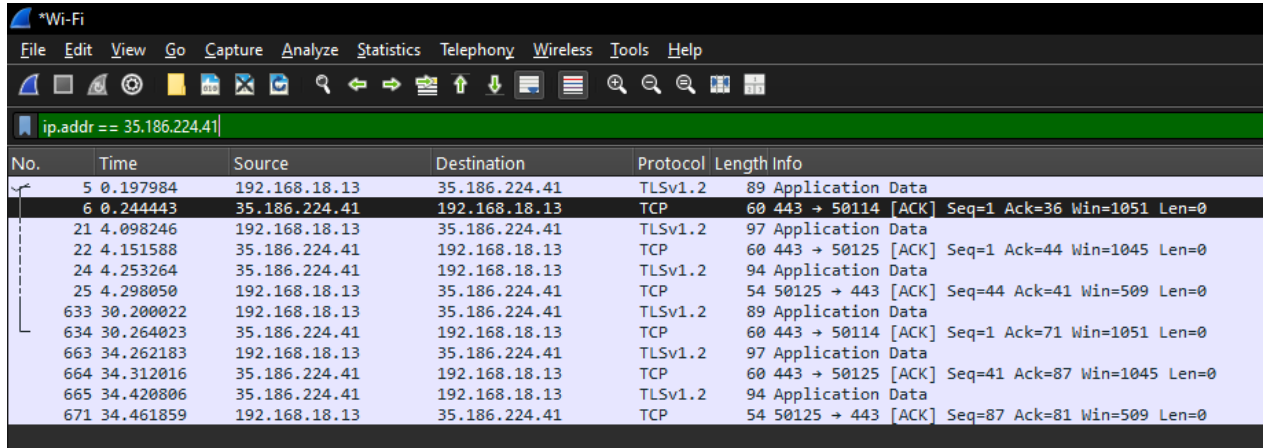
Captured 1061 packets

1049	51.161080	3e:52:a1:07:ef:72	Broadcast	AR
1050	51.161080	3e:52:a1:07:ef:72	Broadcast	AR
1051	51.161080	3e:52:a1:07:ef:72	Broadcast	AR
1052	51.161376	192.168.18.13	192.168.243.130	TC
1053	51.698524	52.34.114.27	192.168.18.13	TL
1054	51.698524	20.207.73.82	192.168.18.13	TL
1055	51.698524	20.207.73.82	192.168.18.13	TL
1056	51.698524	20.207.73.82	192.168.18.13	TC
1057	51.698524	204.79.197.222	192.168.18.13	TC
1058	51.698585	192.168.18.13	52.34.114.27	TC
1059	51.698641	192.168.18.13	20.207.73.82	TC
1060	51.698781	192.168.18.13	20.207.73.82	TC
1061	52.009608	192.168.18.13	20.207.73.82	TC

# Now lets analyze it by applying filters

## 1. Filter by IP Address:

ip.addr == 35.186.224.41



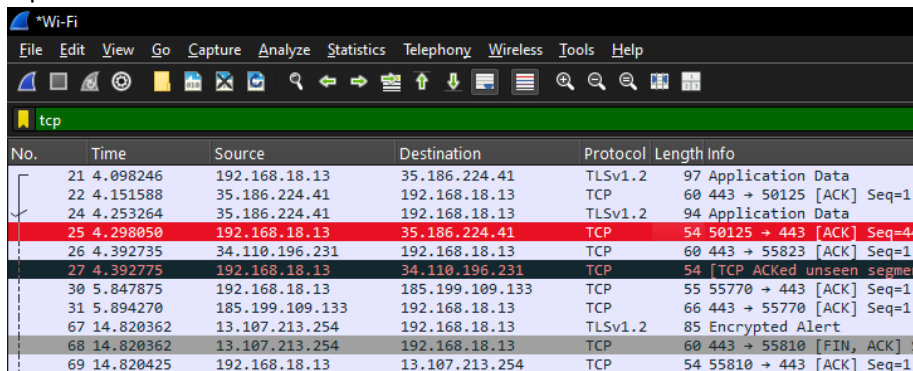
The screenshot shows the Wireshark interface with the filter 'ip.addr == 35.186.224.41' applied. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Length	Info
5	0.197984	192.168.18.13	35.186.224.41	TLSv1.2	89	Application Data
6	0.244443	35.186.224.41	192.168.18.13	TCP	60	443 → 50114 [ACK] Seq=1 Ack=36 Win=1051 Len=0
21	4.098246	192.168.18.13	35.186.224.41	TLSv1.2	97	Application Data
22	4.151588	35.186.224.41	192.168.18.13	TCP	60	443 → 50125 [ACK] Seq=1 Ack=44 Win=1045 Len=0
24	4.253264	35.186.224.41	192.168.18.13	TLSv1.2	94	Application Data
25	4.298050	192.168.18.13	35.186.224.41	TCP	54	50125 → 443 [ACK] Seq=44 Ack=41 Win=509 Len=0
633	30.200022	192.168.18.13	35.186.224.41	TLSv1.2	89	Application Data
634	30.264023	35.186.224.41	192.168.18.13	TCP	60	443 → 50114 [ACK] Seq=1 Ack=71 Win=1051 Len=0
663	34.262183	192.168.18.13	35.186.224.41	TLSv1.2	97	Application Data
664	34.312016	35.186.224.41	192.168.18.13	TCP	60	443 → 50125 [ACK] Seq=41 Ack=87 Win=1045 Len=0
665	34.420806	35.186.224.41	192.168.18.13	TLSv1.2	94	Application Data
671	34.461859	192.168.18.13	35.186.224.41	TCP	54	50125 → 443 [ACK] Seq=87 Ack=81 Win=509 Len=0

Using this filter we can see traffic of a specific ip address

## 2. Filter by Protocol:

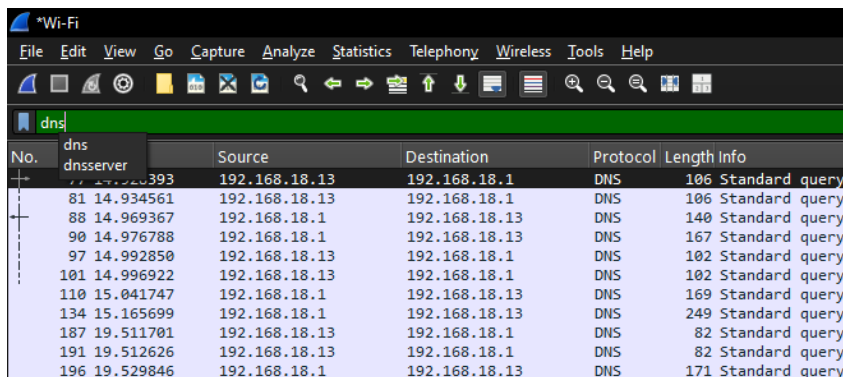
tcp



The screenshot shows the Wireshark interface with the filter 'tcp' applied. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Length	Info
21	4.098246	192.168.18.13	35.186.224.41	TLSv1.2	97	Application Data
22	4.151588	35.186.224.41	192.168.18.13	TCP	60	443 → 50125 [ACK] Seq=1 A
24	4.253264	35.186.224.41	192.168.18.13	TLSv1.2	94	Application Data
25	4.298050	192.168.18.13	35.186.224.41	TCP	54	50125 → 443 [ACK] Seq=44
26	4.392735	34.110.196.231	192.168.18.13	TCP	60	443 → 55823 [ACK] Seq=1 A
27	4.392775	192.168.18.13	34.110.196.231	TCP	54	[TCP ACKed unseen segment
30	5.847875	192.168.18.13	185.199.109.133	TCP	55	55770 → 443 [ACK] Seq=1 A
31	5.894270	185.199.109.133	192.168.18.13	TCP	66	443 → 55770 [ACK] Seq=1 A
67	14.820362	13.107.213.254	192.168.18.13	TLSv1.2	85	Encrypted Alert
68	14.820362	13.107.213.254	192.168.18.13	TCP	60	443 → 55810 [FIN, ACK] Se
69	14.820425	192.168.18.13	13.107.213.254	TCP	54	55810 → 443 [ACK] Seq=1 A

Dns



The screenshot shows the Wireshark interface with the filter 'dns' applied. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Length	Info
77	14.932393	192.168.18.13	192.168.18.1	DNS	106	Standard query
81	14.934561	192.168.18.13	192.168.18.1	DNS	106	Standard query
88	14.969367	192.168.18.1	192.168.18.13	DNS	140	Standard query
90	14.976788	192.168.18.1	192.168.18.13	DNS	167	Standard query
97	14.992850	192.168.18.13	192.168.18.1	DNS	102	Standard query
101	14.996922	192.168.18.13	192.168.18.1	DNS	102	Standard query
110	15.041747	192.168.18.1	192.168.18.13	DNS	169	Standard query
134	15.165699	192.168.18.1	192.168.18.13	DNS	249	Standard query
187	19.511701	192.168.18.13	192.168.18.1	DNS	82	Standard query
191	19.512626	192.168.18.13	192.168.18.1	DNS	82	Standard query
196	19.529846	192.168.18.1	192.168.18.13	DNS	171	Standard query

## 3. Filter by Port Number:

tcp.port == 53

tcp.port == 53							
No.	Time	Source	Destination	Protocol	Port No	Length	Info
1385	27.263584	192.168.18.13	192.168.18.1	TCP	53	66	56024 → 53 [SYN] Seq=0 Win=6424
1386	27.263787	192.168.18.13	192.168.18.1	TCP	53	66	56025 → 53 [SYN] Seq=0 Win=6424
1390	27.268217	192.168.18.1	192.168.18.13	TCP	56024	66	53 → 56024 [SYN, ACK] Seq=0 Ack
1391	27.268292	192.168.18.13	192.168.18.1	TCP	53	54	56024 → 53 [ACK] Seq=1 Ack=1 Wi
1392	27.268400	192.168.18.13	192.168.18.1	TCP	53	56	56024 → 53 [PSH, ACK] Seq=1 Ack
1393	27.268446	192.168.18.13	192.168.18.1	DNS	53	89	Standard query 0x1e54 A sync-v2
1394	27.268523	192.168.18.1	192.168.18.13	TCP	56025	66	53 → 56025 [SYN, ACK] Seq=0 Ack
1395	27.268579	192.168.18.13	192.168.18.1	TCP	53	54	56025 → 53 [ACK] Seq=1 Ack=1 Wi
1396	27.268681	192.168.18.13	192.168.18.1	TCP	53	56	56025 → 53 [PSH, ACK] Seq=1 Ack
1397	27.268715	192.168.18.13	192.168.18.1	DNS	53	89	Standard query 0x3961 HTTPS syn
1398	27.270177	192.168.18.1	192.168.18.13	TCP	56024	60	53 → 56024 [ACK] Seq=1 Ack=3 Wi
1399	27.270513	192.168.18.1	192.168.18.13	TCP	56024	60	53 → 56024 [ACK] Seq=1 Ack=38 Wi

#### 4. Filter by Source or Destination:

ip.src == 192.168.18.13

ip.src == 192.168.18.13							
No.	Time	Source	Destination	Protocol	Port No	Length	Info
6	0.001732	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] S
7	0.002253	192.168.18.13	150.171.72.254	TLSv1.2	443	92	Application Data
10	0.004957	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] S
11	0.006922	192.168.18.13	150.171.72.254	TLSv1.2	443	136	Application Data
19	0.125281	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] S
20	0.127692	192.168.18.13	204.79.197.222	TLSv1.2	443	723	Application Data
24	0.173066	192.168.18.13	204.79.197.222	TCP	443	54	55992 → 443 [ACK] S
26	1.851086	192.168.18.13	2.16.158.176	TCP	443	1438	55985 → 443 [ACK] S
27	1.851086	192.168.18.13	2.16.158.176	TLSv1.2	443	1414	Application Data
28	1.851276	192.168.18.13	2.16.158.176	TCP	443	1438	55985 → 443 [ACK] S
29	1.851276	192.168.18.13	2.16.158.176	TCP	443	1438	55985 → 443 [ACK] S
30	1.851276	192.168.18.13	2.16.158.176	TCP	443	1438	55985 → 443 [ACK] S

ip.dst == 192.168.18.13

ip.dst == 192.168.18.13							
No.	Time	Source	Destination	Protocol	Port No	Length	Info
1	0.000000	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] S
2	0.000000	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] S
3	0.000433	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] S
4	0.001676	150.171.72.254	192.168.18.13	TLSv1.2	56020	396	New Session Ticket
5	0.001676	150.171.72.254	192.168.18.13	TLSv1.2	56020	123	Application Data
8	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	603	Application Data
9	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	92	Application Data
15	0.114281	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] S
16	0.119168	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] S
17	0.125227	150.171.72.254	192.168.18.13	TLSv1.2	56020	253	Application Data

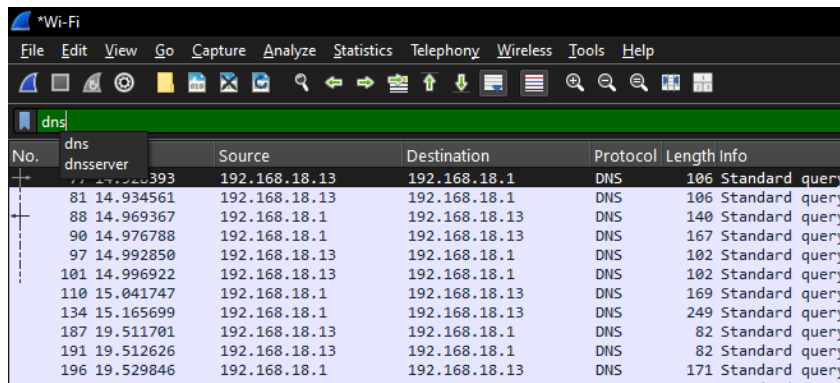
These are some common filters that are used in industry

# Identify common network protocols and traffic patterns.

## 1. DNS (Domain Name System) – Port 53

Function:

- DNS translates human-readable domain names (e.g., www.google.com) into IP addresses (e.g., 142.250.190.14).
- Without DNS, users would need to remember IP addresses instead of domain names.



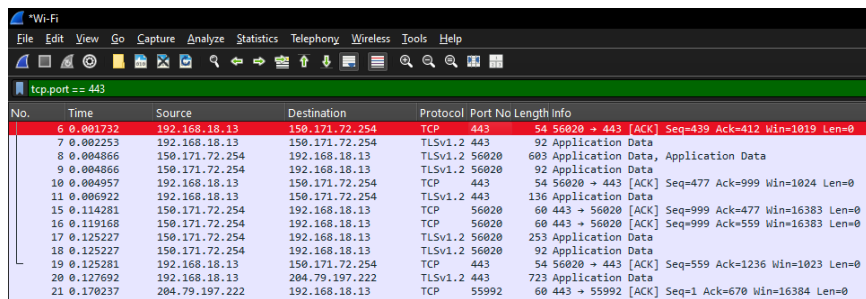
A screenshot of the Wireshark network protocol analyzer showing a capture of DNS traffic. The filter bar at the top is set to 'dns'. The packet list on the left shows several DNS queries. The selected packet (No. 81) is expanded, showing the details of a standard query from source 192.168.18.13 to destination 192.168.18.1.

No.	dns	Source	Destination	Protocol	Length	Info
81	14.934561	192.168.18.13	192.168.18.1	DNS	106	Standard query
88	14.969367	192.168.18.1	192.168.18.13	DNS	140	Standard query
90	14.976788	192.168.18.1	192.168.18.13	DNS	167	Standard query
97	14.992850	192.168.18.13	192.168.18.1	DNS	102	Standard query
101	14.996922	192.168.18.13	192.168.18.1	DNS	102	Standard query
110	15.041747	192.168.18.1	192.168.18.13	DNS	169	Standard query
134	15.165699	192.168.18.1	192.168.18.13	DNS	249	Standard query
187	19.511701	192.168.18.13	192.168.18.1	DNS	82	Standard query
191	19.512626	192.168.18.13	192.168.18.1	DNS	82	Standard query
196	19.529846	192.168.18.1	192.168.18.13	DNS	171	Standard query

## 2. HTTP & HTTPS – Ports 80 & 443

HTTP (Hypertext Transfer Protocol) – Port 80

- Used for communication between web browsers and servers.
- Sends and receives web pages, images, and other resources.
- **Not encrypted**, making it vulnerable to interception (e.g., MITM attacks).



A screenshot of the Wireshark network protocol analyzer showing a capture of HTTP traffic. The filter bar at the top is set to 'tcp.port == 443'. The packet list on the left shows several HTTP requests and responses. The selected packet (No. 6) is expanded, showing the details of a GET request from source 192.168.18.13 to destination 150.171.72.254.

No.	Time	Source	Destination	Protocol	Port No	Length	Info
6	0.001732	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] Seq=439 Ack=412 Win=1019 Len=0
7	0.002253	192.168.18.13	150.171.72.254	TLSv1.2	443	92	Application Data
8	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	603	Application Data, Application Data
9	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	92	Application Data
10	0.004957	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] Seq=477 Ack=999 Win=1024 Len=0
11	0.006922	192.168.18.13	150.171.72.254	TLSv1.2	443	136	Application Data
15	0.114281	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] Seq=999 Ack=477 Win=16383 Len=0
16	0.119168	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020 [ACK] Seq=999 Ack=559 Win=16383 Len=0
17	0.125227	150.171.72.254	192.168.18.13	TLSv1.2	56020	253	Application Data
18	0.125227	150.171.72.254	192.168.18.13	TLSv1.2	56020	92	Application Data
19	0.125281	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443 [ACK] Seq=559 Ack=1236 Win=1023 Len=0
20	0.127692	192.168.18.13	204.79.197.222	TLSv1.2	443	723	Application Data
21	0.170237	204.79.197.222	192.168.18.13	TCP	55992	60	443 → 55992 [ACK] Seq=1 Ack=670 Win=16384 Len=0

## 3. TCP & UDP – Transport Layer Protocols

TCP (Transmission Control Protocol) – Connection-Oriented

- Reliable data transfer with **error checking and retransmission**.
- Uses a **three-way handshake** before data is sent:
  1. SYN (Client → Server)
  2. SYN-ACK (Server → Client)
  3. ACK (Client → Server)
- Used by HTTP, HTTPS, FTP, SSH, and email protocols (SMTP, IMAP, POP3).

tcp									
No.	Time	Source	Destination	Protocol	Port No	Length	Info		
1	0.000000	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020	[ACK]	Seq=1 Ack=1 Win=16385 Len=0
2	0.000000	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020	[ACK]	Seq=1 Ack=88 Win=16385 Len=0
3	0.000433	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020	[ACK]	Seq=1 Ack=439 Win=16383 Len=0
4	0.001676	150.171.72.254	192.168.18.13	TLSv1.2	56020	396	New Session Ticket, Change Cipher Spec, Encrypted Handsh		
5	0.001676	150.171.72.254	192.168.18.13	TLSv1.2	56020	123	Application Data		
6	0.001732	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443	[ACK]	Seq=439 Ack=412 Win=1019 Len=0
7	0.002253	192.168.18.13	150.171.72.254	TLSv1.2	443	92	Application Data		
8	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	603	Application Data, Application Data		
9	0.004866	150.171.72.254	192.168.18.13	TLSv1.2	56020	92	Application Data		
10	0.004957	192.168.18.13	150.171.72.254	TCP	443	54	56020 → 443	[ACK]	Seq=477 Ack=999 Win=1024 Len=0
11	0.006922	192.168.18.13	150.171.72.254	TLSv1.2	443	136	Application Data		
15	0.114281	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020	[ACK]	Seq=999 Ack=477 Win=16383 Len=0
16	0.119168	150.171.72.254	192.168.18.13	TCP	56020	60	443 → 56020	[ACK]	Seq=999 Ack=559 Win=16383 Len=0
17	0.125227	150.171.72.254	192.168.18.13	TLSv1.2	56020	253	Application Data		
18	0.125227	150.171.72.254	192.168.18.13	TLSv1.2	56020	92	Application Data		

## 4. ARP (Address Resolution Protocol)

Function:

- Maps **IP addresses** to **MAC addresses** on a local network.
- Used in LAN communication when a device wants to send data but doesn't know the destination MAC address.

arp									
No.	Time	Source	Destination	Protocol	Port No	Length	Info		
12	0.040371	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.161? Tell 192.168.18.29		
13	0.040371	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.142? Tell 192.168.18.29		
14	0.041323	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.26? Tell 192.168.18.29		
25	1.063083	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.142? Tell 192.168.18.29		
45	2.087485	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.142? Tell 192.168.18.29		
51	3.008148	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.139? Tell 192.168.18.29		
52	3.009187	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.10? Tell 192.168.18.29		
53	3.111703	3e:52:a1:07:ef:72	Broadcast	ARP		60	Who has 192.168.18.22? (ARP Probe)		
54	4.033255	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.139? Tell 192.168.18.29		
55	4.033255	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.10? Tell 192.168.18.29		
56	5.057727	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.139? Tell 192.168.18.29		
57	5.057727	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.10? Tell 192.168.18.29		
58	5.059605	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.11? Tell 192.168.18.29		
59	5.059605	3e:52:a1:07:ef:72	Broadcast	ARP		60	Who has 192.168.18.1? Tell 192.168.18.22		
65	7.103955	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.3? Tell 192.168.18.29		
66	7.106655	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.6? Tell 192.168.18.29		
67	7.106655	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.5? Tell 192.168.18.29		
68	7.106655	3e:52:a1:07:ef:72	Broadcast	ARP		42	Who has 192.168.18.4? Tell 192.168.18.29		

## 5. ICMP (Internet Control Message Protocol) – Ping & Network Diagnostics

Function:

- Used for **network troubleshooting**, testing connectivity, and error reporting.
- Works at the **Network Layer (Layer 3 of the OSI Model)**.

*Wi-Fi									
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help									
icmp									
No.	Time	Source	Destination	Protocol	Port No	Length	Info		
icmp				icmpv6					

<i>Protocol</i>	<i>Port</i>	<i>Function</i>	<i>Security Risks</i>	<i>Wireshark Filter</i>
<i>DNS</i>	53	Resolves domain names to IP addresses	DNS spoofing, cache poisoning	dns
<i>HTTP</i>	80	Transfers web pages (unencrypted)	MITM attacks, data interception	http
<i>HTTPS</i>	443	Secure web browsing (encrypted)	SSL/TLS vulnerabilities	tcp.port == 443
<i>TCP</i>	Various	Reliable, connection-based transport	SYN floods, session hijacking	tcp
<i>UDP</i>	Various	Fast, connectionless transport	Packet loss, DDoS attacks	udp
<i>ARP</i>	No port (Layer 2)	Maps IP addresses to MAC addresses	ARP spoofing, MITM attacks	arp
<i>ICMP</i>	No port (Layer 3)	Network diagnostics, pings	Ping floods, ICMP tunneling	icmp