OSI Model,TCP/IP Model and Wireshark Packet Analysis

1. OSI Layer Explanations

Layer 1 – Physical Layer

This layer is the quiet workhorse of the network world. It deals with the raw signals—electrical pulses, light flashes, or radio waves—that travel through cables and air. No understanding of data happens here; it's just bits marching in a steady rhythm. Ethernet cables, fiber optics, Wi-Fi radio signals, voltage levels, connectors, and physical topologies belong here. You can imagine it like the actual road on which vehicles travel: without the asphalt or the rails, nothing else can move.

Layer 2 – Data Link Layer

Here the network gains awareness of neighbors. The Data Link Layer shapes raw bits into frames, attaches MAC addresses, and checks for errors using CRC. Switches, NICs, VLANs, ARP, and protocols like Ethernet and PPP live here. It's the traffic manager inside a city block: it ensures local deliveries reach the right door, even if the rest of the world remains a mystery. Collisions are handled, and corrupted frames are quietly filtered out so higher layers don't panic

Layer 3 – Network Layer

This is the layer where a device learns how to travel beyond its local area. The Network Layer handles logical addressing and routing—IP addresses, subnets, routers, and routing protocols like OSPF, RIP, and BGP. Packets may hop through multiple networks, each router choosing the next turn based on routing tables. It's like the postal service's long-distance system: envelopes move from city to city through a maze of interconnected routes until they find the right region.

Layer 4 – Transport Layer

Here the network learns to care about reliability and order. This layer decides whether data should arrive carefully (TCP) or quickly (UDP). It forms segments, tracks port numbers, handles re-transmissions, flow control, and ensures that scattered pieces of data get reassembled like puzzle tiles. It resembles a courier service that either delivers fragile items with great caution or drops off parcels rapidly with no signatures— depending on what the sender wants.

Layer 5 – Session Layer

The Session Layer is like the host of a long conversation. It creates, manages, and gracefully ends communication sessions between devices. It keeps track of who is speaking, when, and ensures that if a connection drops, it can be resumed. Protocols such as RPC and NetBIOS sit here. Think of it as a moderator in a meeting room who ensures that the discussion keeps flowing without two people talking over each other.

Layer 6 – Presentation Layer

This layer acts as the translator and stylist of the network. It converts data into formats that applications can understand—handling encryption, compression, and encoding. SSL/TLS (for encryption), JPEG, MP3, GIF, and text encoding schemes like ASCII and UTF-8 show up here. Picture a multilingual editor who reformats and translates a document so the final reader receives it in the exact style and language they expect.

Layer 7 – Application Layer

At the top sits the layer users actually interact with—even if they don't realize it. It provides interfaces for email, browsing, file transfers, and other network services. Protocols like HTTP, HTTPS, FTP, SMTP, DNS, and DHCP operate here. It's like the front counter of a service desk: people walk up, make requests, and receive what they need, while the deeper layers do the hidden heavy lifting behind the wall.

2. OSI Mnemonic

"Please Don't Nap Too Soon, People Are Watching."

Word	OSI Layer
Please	**Physical**

Don’t	Data Link
Nap	Network
Too	Transport
Soon	Session
People	Presentation
Are Watching	Application

3. OSI vs TCP/IP Model Comparison

The OSI model is a seven-layer staircase, each step carefully separating responsibilities—from raw signals at the bottom to user-facing applications at the top. It’s a detailed blueprint, almost theoretical in places, offering a clean way to study how networks behave. The TCP/IP model, meanwhile, is the practical four-layer framework the internet actually runs on. Instead of slicing tasks into fine layers, it folds related functions together— much like packing fewer but sturdier bags for a long trip.

Where OSI gives a highly structured, “textbook-perfect” perspective, TCP/IP reflects real-world protocol stacks. The upper OSI layers (Application, Presentation, Session) merge into a single Application layer in TCP/IP; the lower layers (Data Link and Physical) condense into a Network Access layer. Both models describe the same journey of data, just with different levels of granularity.

OSI Layer	TCP/IP Layer	Explanation
Application (L7)	**Application Layer**	TCP/IP combines user-facing services here.

Presentation (L6)	**Application Layer**	Formatting, encryption, compression are also handled by the TCP/IP Application layer.
Session (L5)	**Application Layer**	Session management is included within TCP/IP Application protocols.
Transport (L4)	**Transport Layer**	TCP/UDP control reliability and ports.
Network (L3)	**Internet Layer**	IP, routing, addressing.
Data Link (L2)	**Network Access Layer**	Frames, MAC addresses, NICs.
Physical (L1)	**Network Access Layer**	Cables, signals, physical transmission.

4. Protocol Data Units (PDUs)

OSI Layer	**PDU Name**	**Notes**
Layer 4 – Transport	**Segment (TCP) / Datagram (UDP)**	TCP behaves like a careful courier (segments), while UDP tosses lightweight datagrams without ceremony.

Layer 3 – Network	**Packet**	Carries IP addresses and travels across multiple networks.
Layer 2 – Data Link	**Frame**	Wrapped with MAC addresses; perfect for local delivery.
Layer 1 – Physical	**Bits**	Raw 1s and 0s racing through cables or airwaves.

5. Addressing Concepts

1. MAC Address – used at Layer 2 (Data Link)

A MAC address is the hardware identifier burned into a network interface card. It's like a permanent name tag that a device carries on a local network. Layer 2 uses this address to deliver frames from one device to another within the same LAN. Switches rely on MAC addresses to decide which port should receive a frame, keeping the local traffic flowing neatly.

2. IP Address – used at Layer 3 (Network)

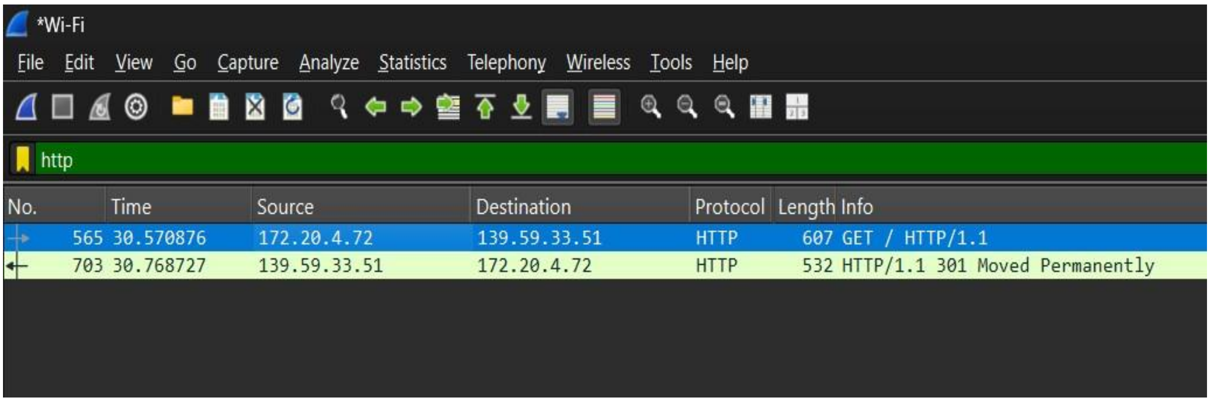
An IP address is a logical address assigned to a device so it can be located across different networks. Routers use IP addresses to move packets from one network to another, choosing paths and forwarding packets hop by hop. While MAC handles "who is next door," the IP address handles "where in the world should this go?"

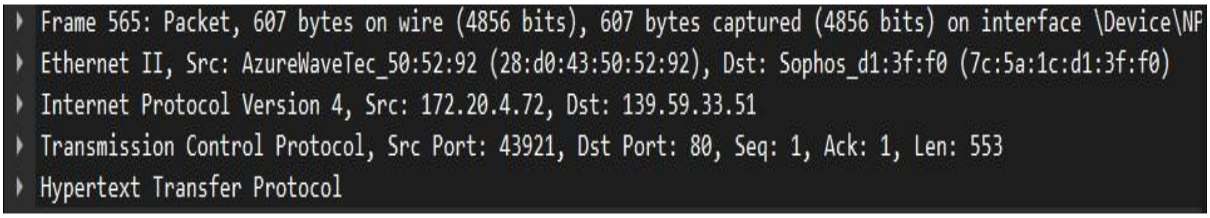
3. Port Number – used at Layer 4 (Transport)

A port number identifies a specific application or service running on a device. The Transport layer uses ports to make sure data meant for a web browser doesn't accidentally land in a video player. TCP and UDP both use port numbers like tiny door labels, helping the device deliver segments or datagrams to the right process.

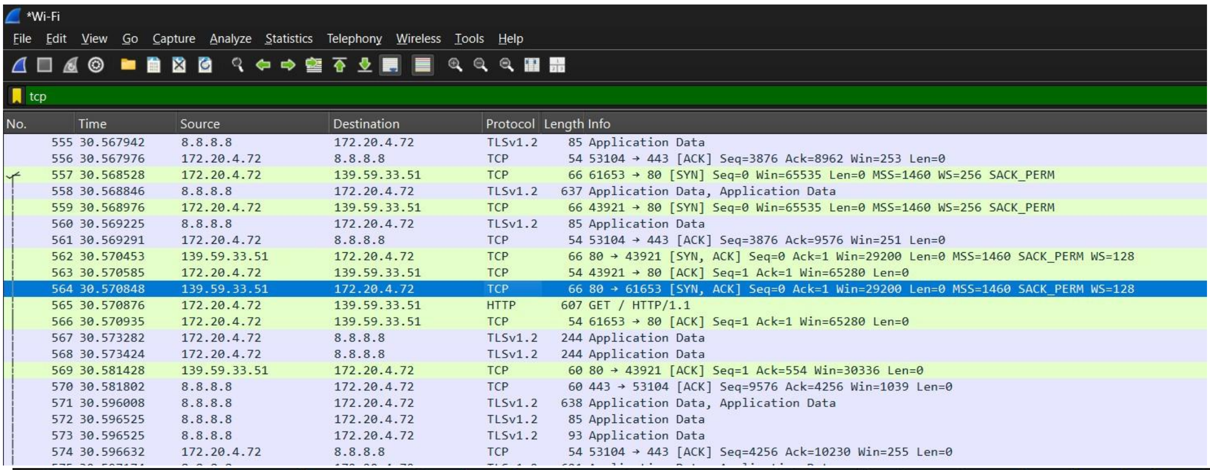
Part B – Wireshark Practical

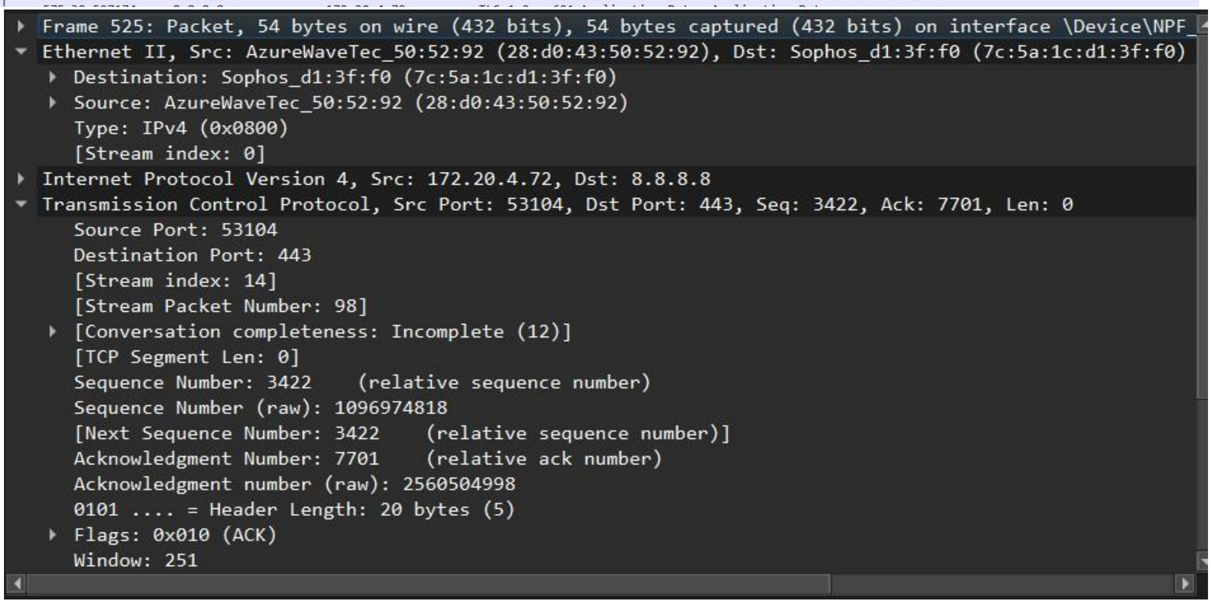
1. HTTP Traffic



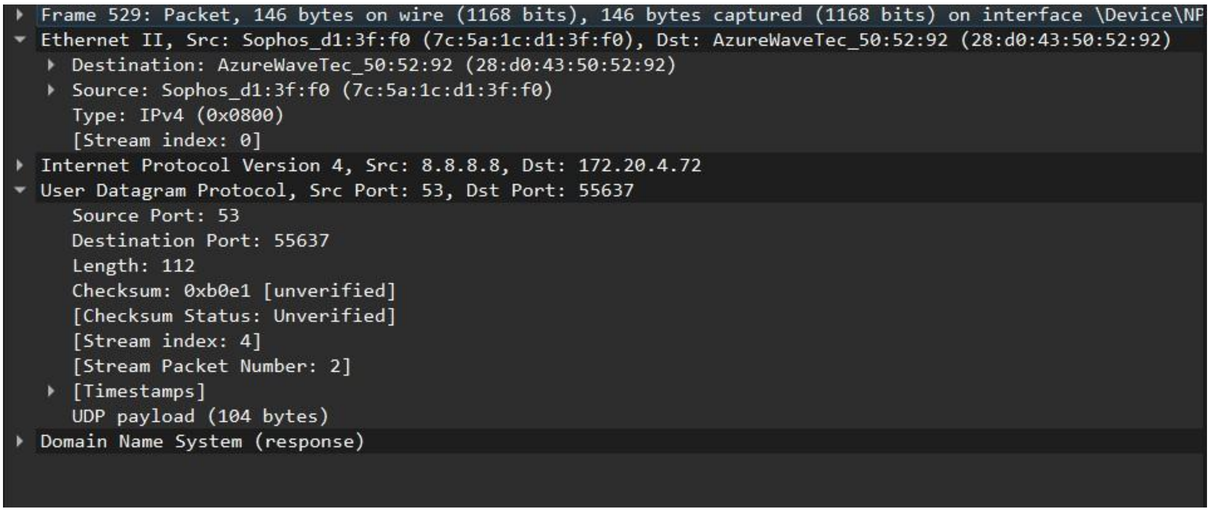


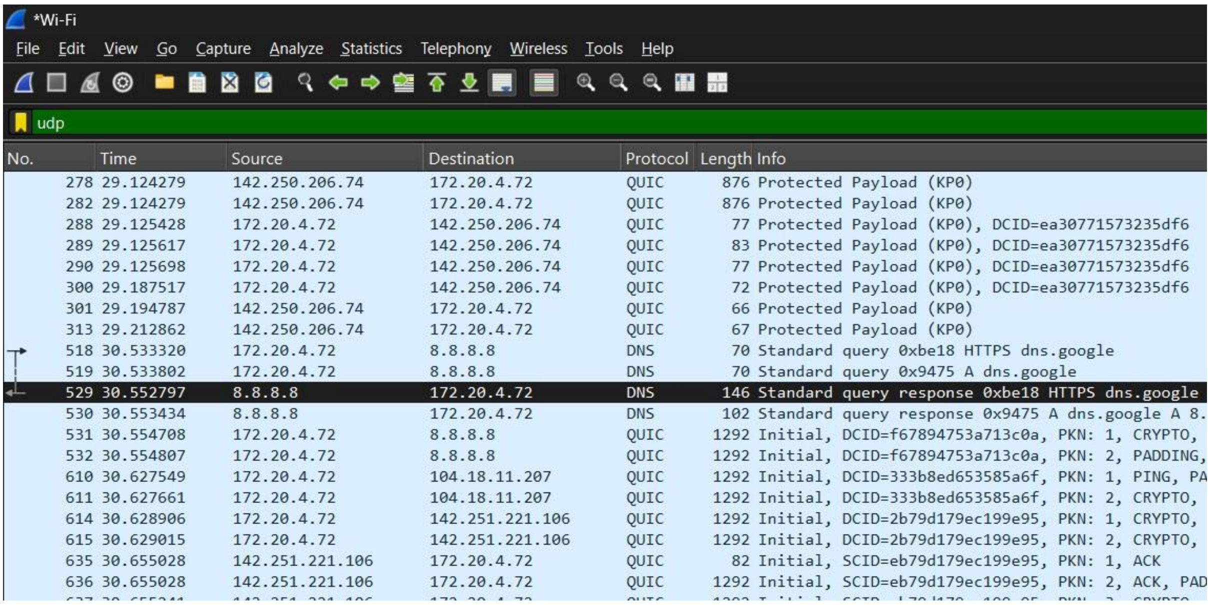
2. TCP Packets



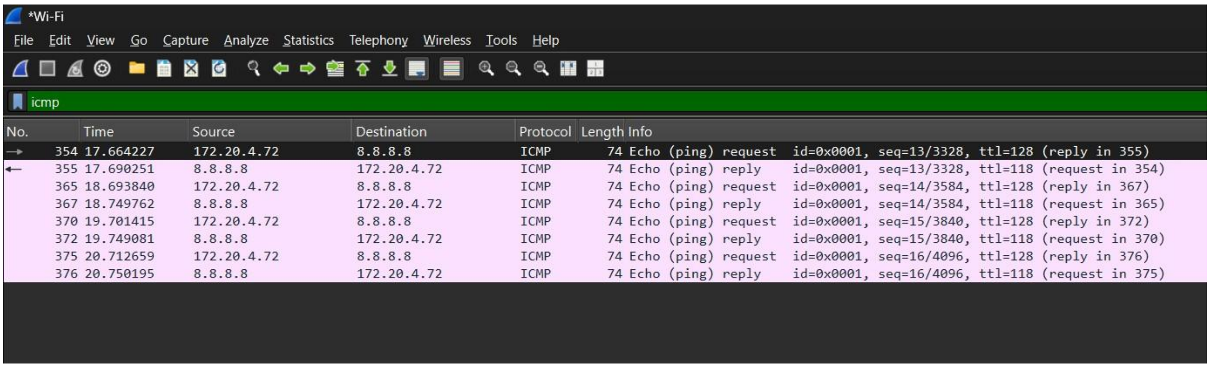


3. UDP Packets





4. ICMP Packets



```
Frame 354: Packet, 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{3
Ethernet II, Src: AzureWaveTec_50:52:92 (28:d0:43:50:52:92), Dst: Sophos_d1:3f:f0 (7c:5a:1c:d1:3f:f0)
    Destination: Sophos_d1:3f:f0 (7c:5a:1c:d1:3f:f0)
    Source: AzureWaveTec_50:52:92 (28:d0:43:50:52:92)
    Type: IPv4 (0x0800)
    [Stream index: 3]
Internet Protocol Version 4, Src: 172.20.4.72, Dst: 8.8.8.8
Internet Control Message Protocol
    Type: Echo (ping) request (8)
    Code: 0
    Checksum: 0x4d4e [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 13 (0x000d)
    Sequence Number (LE): 3328 (0x0d00)
    [Response frame: 355]
    Data (32 bytes)
```

```
Frame 355: Packet, 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{3
Ethernet II, Src: Sophos_d1:3f:f0 (7c:5a:1c:d1:3f:f0), Dst: AzureWaveTec_50:52:92 (28:d0:43:50:52:92)
    Destination: AzureWaveTec_50:52:92 (28:d0:43:50:52:92)
    Source: Sophos_d1:3f:f0 (7c:5a:1c:d1:3f:f0)
    Type: IPv4 (0x0800)
    [Stream index: 3]
Internet Protocol Version 4, Src: 8.8.8.8, Dst: 172.20.4.72
Internet Control Message Protocol
    Type: Echo (ping) reply (0)
    Code: 0
    Checksum: 0x554e [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 13 (0x000d)
    Sequence Number (LE): 3328 (0x0d00)
    [Request frame: 354]
    [Response time: 26.024 ms]
    Data (32 bytes)
```

5. ARP Frames

*Wi-Fi

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

arp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	32:83:d3:1a:20:ce	Broadcast	ARP	60	ARP Announcement for 172.20.9.23
2	1.804419	c2:0e:43:8f:a8:d3	Broadcast	ARP	60	Gratuitous ARP for 172.20.2.65 (Reply)
3	1.906544	4a:25:92:71:f1:8e	Broadcast	ARP	60	Gratuitous ARP for 172.20.3.219 (Reply)
4	2.008997	32:83:d3:1a:20:ce	Broadcast	ARP	60	ARP Announcement for 172.20.9.23
285	10.042591	26:97:96:98:7c:cf	Broadcast	ARP	60	Gratuitous ARP for 172.18.4.199 (Reply)
291	12.454192	1e:da:0e:9b:8f:81	Broadcast	ARP	60	Gratuitous ARP for 172.20.4.135 (Request)
322	14.194878	0a:ce:e2:81:b4:1f	Broadcast	ARP	60	Gratuitous ARP for 172.20.5.28 (Reply)
369	18.803412	c2:0e:43:8f:a8:d3	Broadcast	ARP	60	Gratuitous ARP for 172.20.2.65 (Reply)
371	19.724593	36:70:bc:9a:e6:d4	Broadcast	ARP	60	ARP Announcement for 172.20.2.205
377	21.158001	36:70:bc:9a:e6:d4	Broadcast	ARP	60	ARP Announcement for 172.20.2.205
412	28.531207	AzureWaveTec_86:6a:…	Broadcast	ARP	60	ARP Announcement for 172.20.9.60
413	29.658339	ba:0a:ac:5b:71:24	Broadcast	ARP	60	Gratuitous ARP for 172.20.6.234 (Request)
414	29.658339	ba:0a:ac:5b:71:24	Broadcast	ARP	60	Gratuitous ARP for 172.20.6.234 (Request)
415	29.658339	ba:0a:ac:5b:71:24	Broadcast	ARP	60	Gratuitous ARP for 172.20.6.234 (Request)
416	30.170150	32:83:d3:1a:20:ce	Broadcast	ARP	60	ARP Announcement for 172.20.9.23

```
Frame 291: Packet, 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{3
Ethernet II, Src: 1e:da:0e:9b:8f:81 (1e:da:0e:9b:8f:81), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: 1e:da:0e:9b:8f:81 (1e:da:0e:9b:8f:81)
    Type: ARP (0x0806)
    [Stream index: 5]
    Padding: 000000000000000000000000000000000000
Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    [Is gratuitous: True]
    Sender MAC address: 1e:da:0e:9b:8f:81 (1e:da:0e:9b:8f:81)
    Sender IP address: 172.20.4.135
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 172.20.4.135
```

```
Frame 322: Packet, 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{3
Ethernet II, Src: 0a:ce:e2:81:b4:1f (0a:ce:e2:81:b4:1f), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: 0a:ce:e2:81:b4:1f (0a:ce:e2:81:b4:1f)
    Type: ARP (0x0806)
    [Stream index: 6]
    Padding: 000000000000000000000000000000000000
Address Resolution Protocol (reply/gratuitous ARP)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    [Is gratuitous: True]
    Sender MAC address: 0a:ce:e2:81:b4:1f (0a:ce:e2:81:b4:1f)
    Sender IP address: 172.20.5.28
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 172.20.5.28
```