# Evidence on Summary of Learning of the 3 core activities and lesson review

Activity 1:

So in summary we learnt about the different layers used in the TCP/IP model and how a data is dent and received between two users.

then we drew a diagram to illustrate visually how the data is sent and recived between toe users using all those layers.

Activity 2:

We used the developer tools in chrome to analyse the messages sent and received in a webpage. we analysed what happened behind the scenes using Network analyse feature name of the file/request, Method, Status, Protocol, Scheme, domain, Type, size, time, connection Id (TCP), Waterfall (Timing sequence of the messages), total response time, byte transferred, and number of requests shown at the bottom of the panel.

Activity 3:

We analzed the same webpage using a tool called wireshark. in this we had a more insights in analyzing the packets. We discussed about the protocols, the time it takes to send and receive packets, How long did it take to receive the HTTP OK reply from the HTTP GET message was sent, what is the ip address of source and destination and many more.

At last we compared using wireshark and developer tools in capturing the packets. we have produced evidence for all the captures as screeshots of the tools we used.

This was the place where I cleared my doubts about the application layer in the TCP/IP model. Of course I learnt a lot in the group activity but as extra reference I got some idea from this website as well, I've provided the link below.


1. *Process software: TCP/IP defined - TCP/IP Protocols*. (n.d.). https://www.process.com/resources/tcpip/library_tcpip3_chap6.html#:~:text=Transport%20layer%20%E2%80%94%20Provides%20the%20reliable%20data,transfer%2C%20network%20file%20access%2C%20and%20electronic%20mail.


I've provided the notes I took related to this topic as evidence as well below.

# Evidence:  Module Exercises

You need to provide evidence of successful completion of **all module exercises.**

Exercise 1:

✓ What are the two additional layers that we have in IOS/OSI reference model compared to TCP/IP model? *10/10

- ○ Presentation and Network
- ○ Session and Transport
- ○ Network and Transport
- ◉ Session and Presentation ✓

Exercise 2:

✓ Which of the following statement is correct? * 10/10

- ○ Network layer protocol encapsulates data-link layer messages.
- ○ Transport layer protocol encapsulates network layer messages.
- ○ Application layer protocols encapsulate transport layer messages.
- ◉ Transport layer protocol encapsulates application layer messages. ✓

Exercise 3:

✓ Wireshark Packet sniffer * 10/10

- ○ captures all application layer messages and cannot used to analyse transport layer data.
- ◉ helps to analyse data link, network, transport and application layer protocols. ✓
- ○ captures all message, but cannot use to analyse application layer protocols.

Exercise 4:

✓ Network layer protocol encapsulates data-link layer segment and create a network-layer datagram *10/10

- ○ True
- ◉ False ✓

Exercise 5:

✓ Suppose users share a 1 Gbps (Gigabits per second) link and each user transmits continuously at 100 Mbps (Megabits per second) when transmitting, but each user transmits only 50% percent at the time. Assume that the network use packet switching. Which of the following statements is correct? *10/10

- ○ If 10 users transmit simultaneously, then there will be a significant queuing delay before the link.
- ○ Since we use packet switching, we can have any number of users and they can transmit packets without a delay.
- ◉ If 22 users transmit simultaneously, then there will be a queuing delay just before the link. ✓

Exercise 6:

✓ In TCP/IP model, each layer can perform all the tasks separately and is not relying on the services provided by the layer below. *10/10

- ○ True
- ◉ False ✓

Exercise 7:

✓ Suppose multiple users share a 40 Mbps link and each user transmits continuously at 10 Mbps. However, when they are transmitting, each user transmits only 40 % of the time. *10/10

When circuit switching is used, how many users can be supported?

- ○ 1
- ○ 6
- ○ 8
- ○ 2
- ◉ 4 ✓

Exercise 8:

✓ How long does it take to transmit 1500 Bytes in 100 kbps (kilobits per second link)? *10/10

- ○ 0.015 S
- ○ 0.15 S
- ○ 1.2 S
- ○ 0. 2 S
- ◉ 0.12 S ✓

Exercise 9:

✓ When a packet is transmitted in a network link (optical/wireless/ copper), it travels through that transmission medium to reach its destination. The time taken for a single bit to propagate from the link to its destination is called "propagation delay" which depends on the distance between the sender and the receiver and the link's propagation speed. *10/10

Propagation Delay = Distance between the sender and receiver/ link speed.

Let's say we are sending packets from Melbourne to Sydney (900 km) via an optical fibre. The propagation speed of the optical fibre is 3×108 m/s. What would be the propagation delay that those packets experienced?

- (●) 3 ms ✓
- ( ) 3 s
- ( ) 300 ms
- ( ) 0.03 s

Exercise 10:

✓ Which layer does not belong to the TCP/IP model? * 10/10

- (●) Presentation Layer ✓
- ( ) Transport Layer
- ( ) Data-Link Layer
- ( ) Network Layer
- ( ) Application Layer

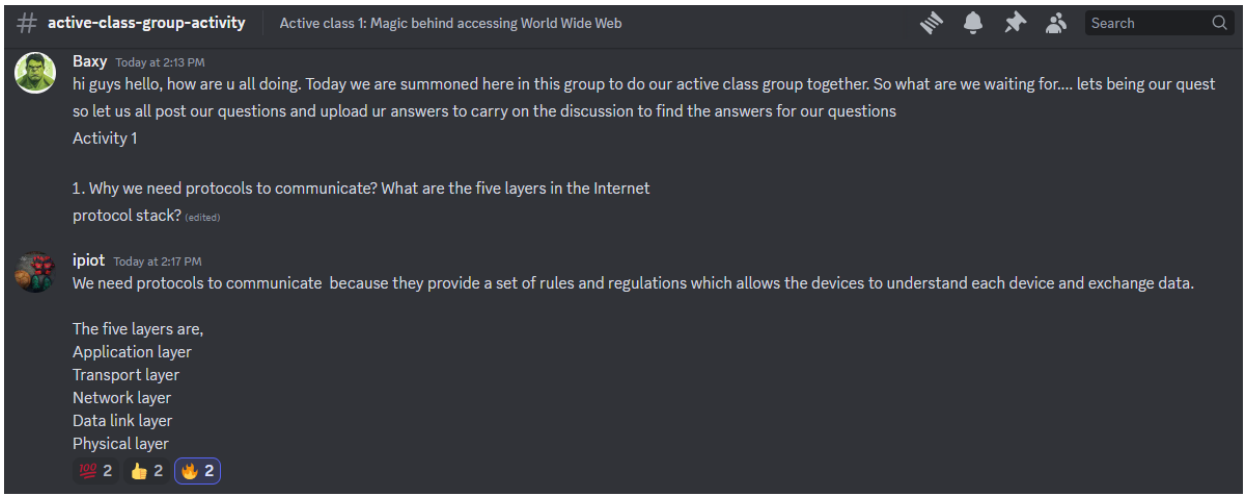
Exercise 11:

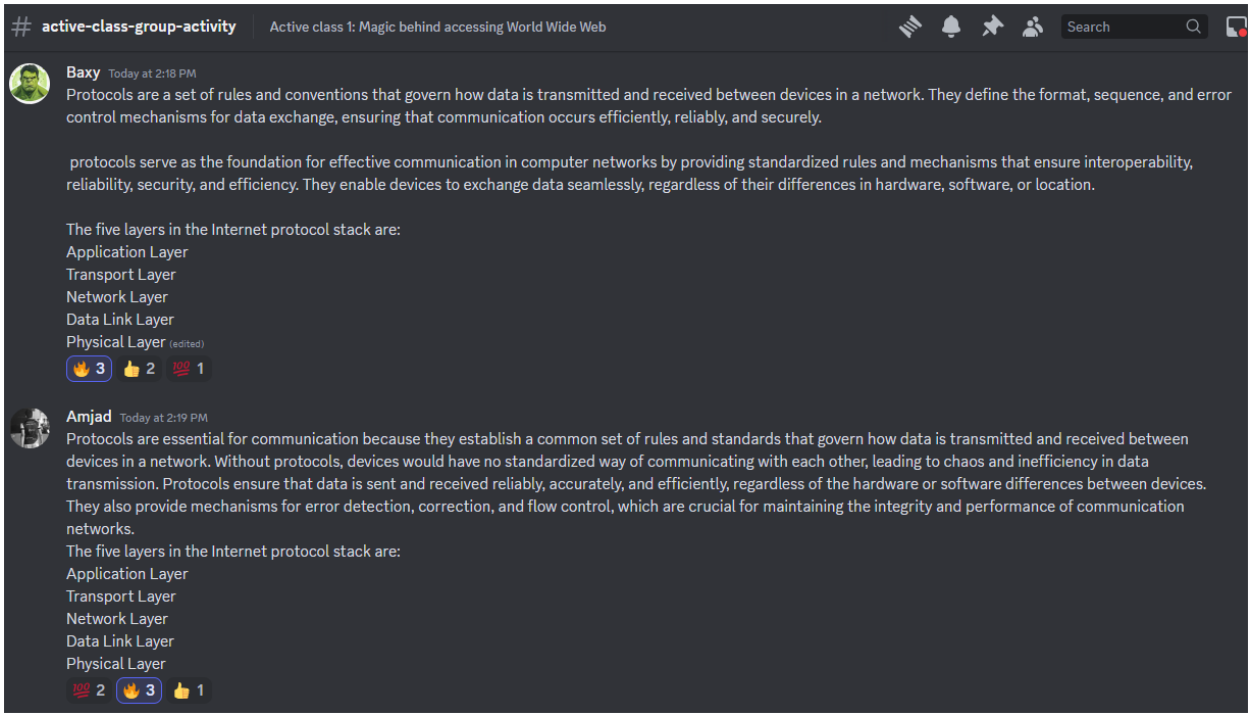
✓ Which of the following protocol is not supported by the application layer of TCP/IP module? *10/10

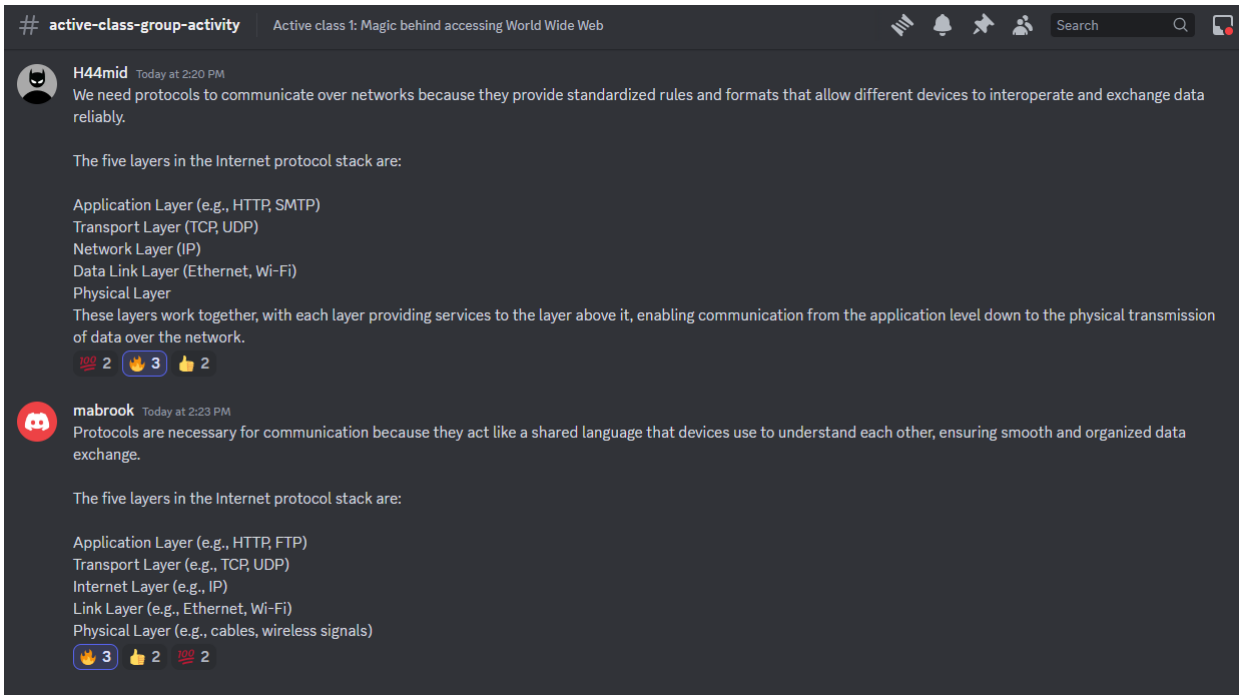
- ○ HTTP
- ○ DNS
- ○ SMTP
- ◉ TCP ✓

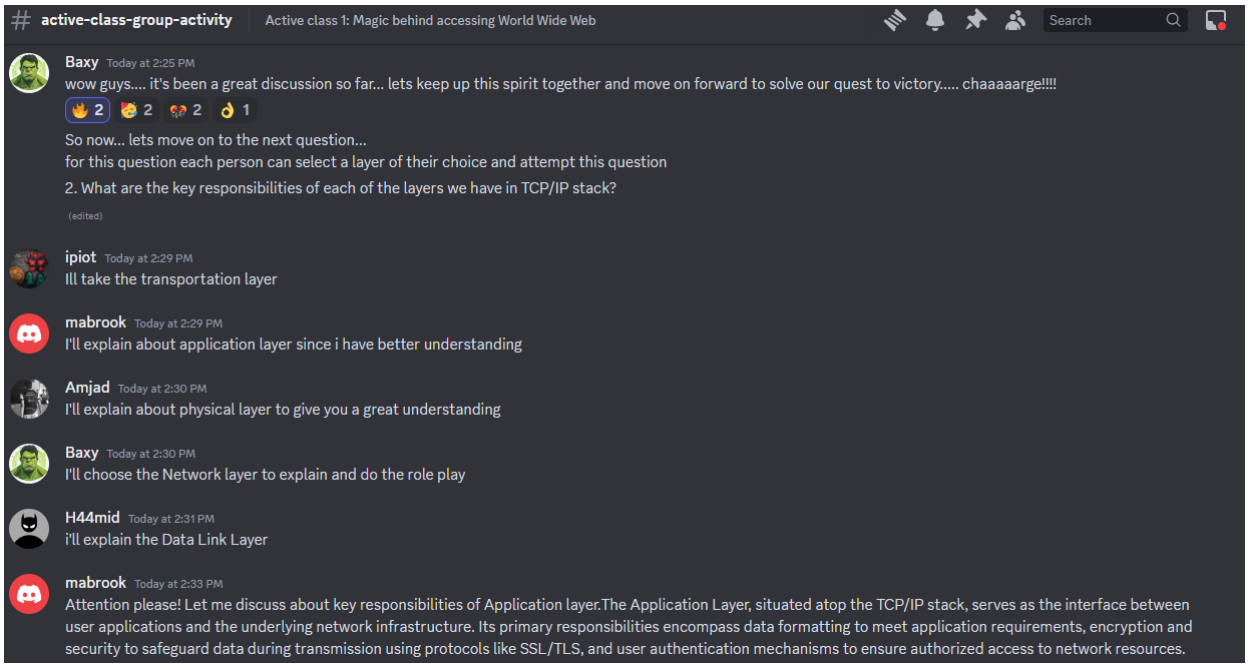
## Activity 1

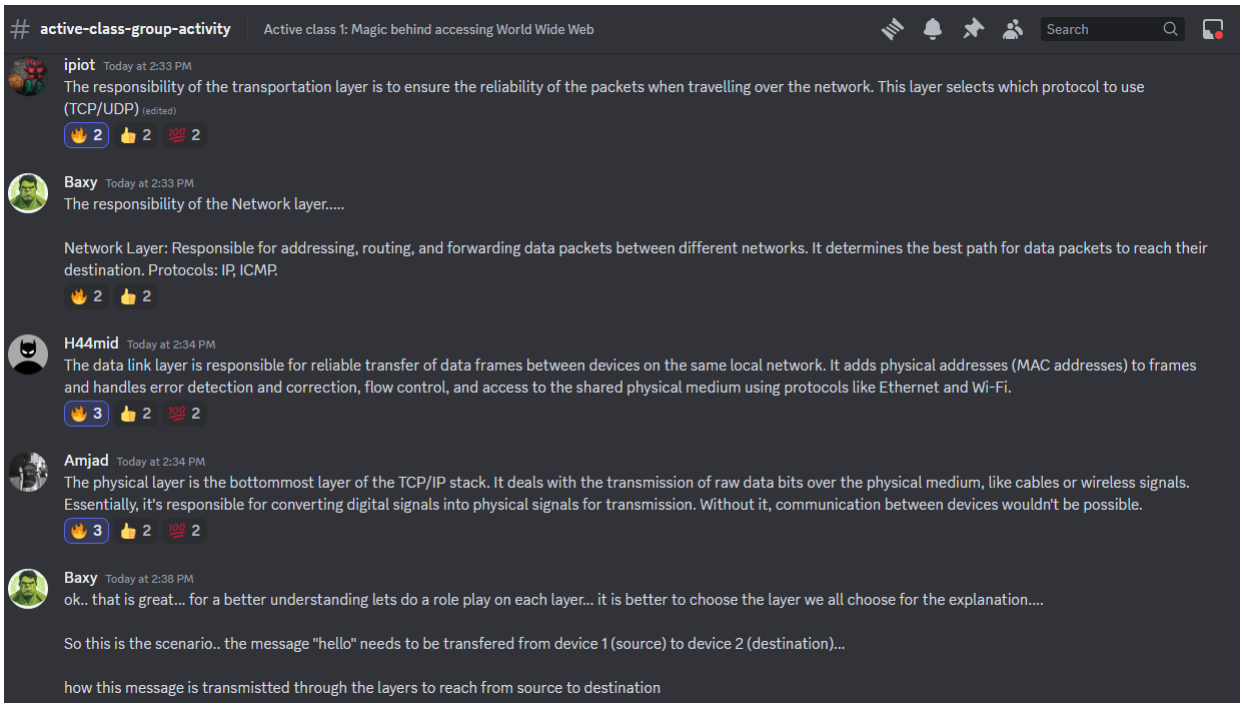
**We made a group of 5 members and using discord we discussed the questions and activities and provided diagrams too to express our understanding of this module of the specific requirements.**




active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
Baxy Today at 2:18 PM
Protocols are a set of rules and conventions that govern how data is transmitted and received between devices in a network. They define the format, sequence, and error control mechanisms for data exchange, ensuring that communication occurs efficiently, reliably, and securely.
protocols serve as the foundation for effective communication in computer networks by providing standardized rules and mechanisms that ensure interoperability, reliability, security, and efficiency. They enable devices to exchange data seamlessly, regardless of their differences in hardware, software, or location.
The five layers in the Internet protocol stack are:
Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer (edited)
3 2 1
Amjad Today at 2:19 PM
Protocols are essential for communication because they establish a common set of rules and standards that govern how data is transmitted and received between devices in a network. Without protocols, devices would have no standardized way of communicating with each other, leading to chaos and inefficiency in data transmission. Protocols ensure that data is sent and received reliably, accurately, and efficiently, regardless of the hardware or software differences between devices. They also provide mechanisms for error detection, correction, and flow control, which are crucial for maintaining the integrity and performance of communication networks.
The five layers in the Internet protocol stack are:
Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer
2 3 1


active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
H44mid Today at 2:20 PM
We need protocols to communicate over networks because they provide standardized rules and formats that allow different devices to interoperate and exchange data reliably.
The five layers in the Internet protocol stack are:
Application Layer (e.g., HTTP, SMTP)
Transport Layer (TCP, UDP)
Network Layer (IP)
Data Link Layer (Ethernet, Wi-Fi)
Physical Layer
These layers work together, with each layer providing services to the layer above it, enabling communication from the application level down to the physical transmission of data over the network.
2 3 2
mabrook Today at 2:23 PM
Protocols are necessary for communication because they act like a shared language that devices use to understand each other, ensuring smooth and organized data exchange.
The five layers in the Internet protocol stack are:
Application Layer (e.g., HTTP, FTP)
Transport Layer (e.g., TCP, UDP)
Internet Layer (e.g., IP)
Link Layer (e.g., Ethernet, Wi-Fi)
Physical Layer (e.g., cables, wireless signals)
3 2 2


active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
Baxy Today at 2:25 PM
wow guys.... it's been a great discussion so far... lets keep up this spirit together and move on forward to solve our quest to victory..... chaaaaarge!!!!
2 2 2 1
So now... lets move on to the next question...
for this question each person can select a layer of their choice and attempt this question
2. What are the key responsibilities of each of the layers we have in TCP/IP stack?
(edited)
ipiot Today at 2:29 PM
Ill take the transportation layer
mabrook Today at 2:29 PM
I'll explain about application layer since i have better understanding
Amjad Today at 2:30 PM
I'll explain about physical layer to give you a great understanding
Baxy Today at 2:30 PM
I'll choose the Network layer to explain and do the role play
H44mid Today at 2:31 PM
i'll explain the Data Link Layer
mabrook Today at 2:33 PM
Attention please! Let me discuss about key responsibilities of Application layer.The Application Layer, situated atop the TCP/IP stack, serves as the interface between user applications and the underlying network infrastructure. Its primary responsibilities encompass data formatting to meet application requirements, encryption and security to safeguard data during transmission using protocols like SSL/TLS, and user authentication mechanisms to ensure authorized access to network resources.

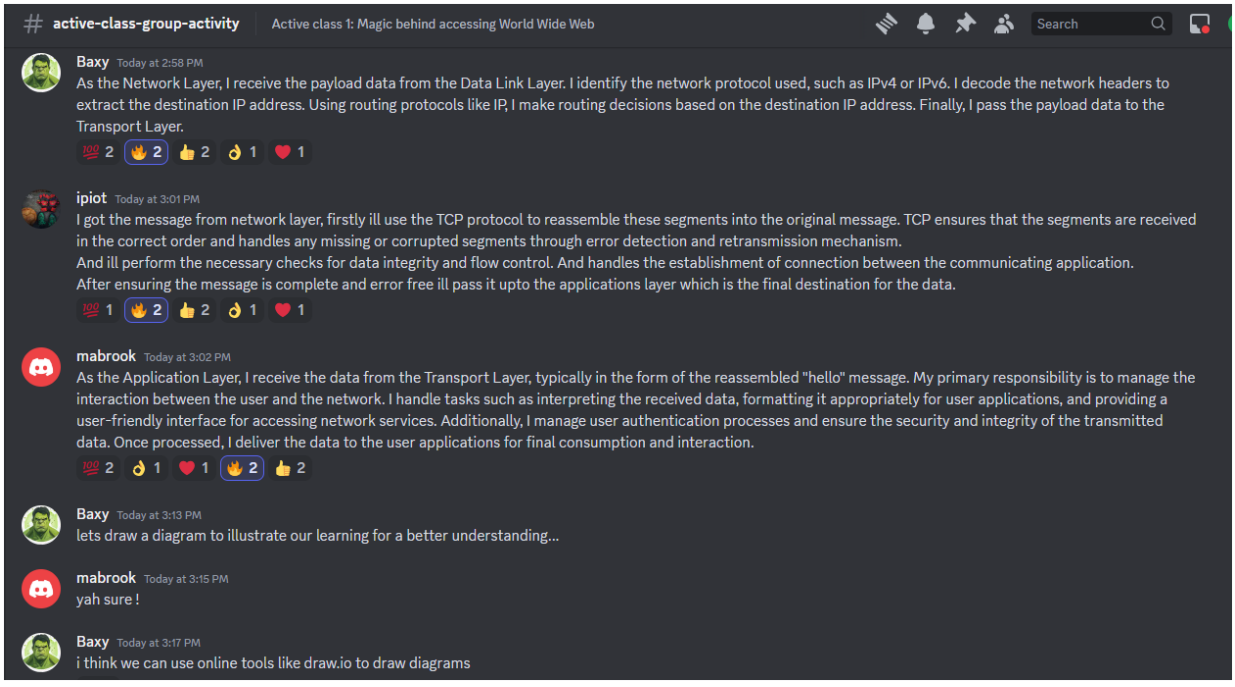

active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
ipiot Today at 2:33 PM
The responsibility of the transportation layer is to ensure the reliability of the packets when travelling over the network. This layer selects which protocol to use (TCP/UDP) (edited)
2 2 2
Baxy Today at 2:33 PM
The responsibility of the Network layer.....
Network Layer: Responsible for addressing, routing, and forwarding data packets between different networks. It determines the best path for data packets to reach their destination. Protocols: IP, ICMP.
2 2
H44mid Today at 2:34 PM
The data link layer is responsible for reliable transfer of data frames between devices on the same local network. It adds physical addresses (MAC addresses) to frames and handles error detection and correction, flow control, and access to the shared physical medium using protocols like Ethernet and Wi-Fi.
3 2 2
Amjad Today at 2:34 PM
The physical layer is the bottommost layer of the TCP/IP stack. It deals with the transmission of raw data bits over the physical medium, like cables or wireless signals. Essentially, it's responsible for converting digital signals into physical signals for transmission. Without it, communication between devices wouldn't be possible.
3 2 2
Baxy Today at 2:38 PM
ok.. that is great... for a better understanding lets do a role play on each layer... it is better to choose the layer we all choose for the explanation....
So this is the scenario.. the message "hello" needs to be transfered from device 1 (source) to device 2 (destination)...
how this message is transmistted through the layers to reach from source to destination

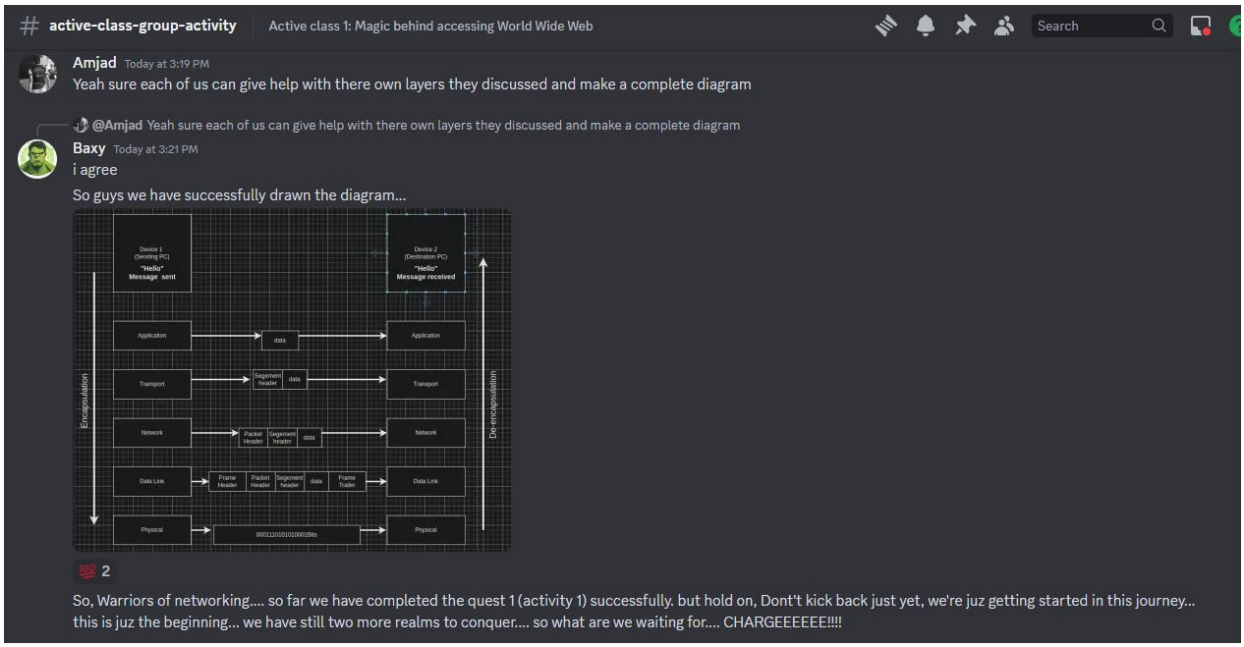
**Amjad** Today at 2:50 PM
The message "hello" is converted into digital format.
I physical layer transmit the data over copper cables, fiber optics, or wirelessly.
Data is modulated onto carrier signals for transmission.
I then send modulated signals over the medium.
At Destination, I receive and demodulate the signals.
I may detect and correct transmission errors as well.
Now I have passed the hello message to device 2 (edited)
🔥 2 💯 3

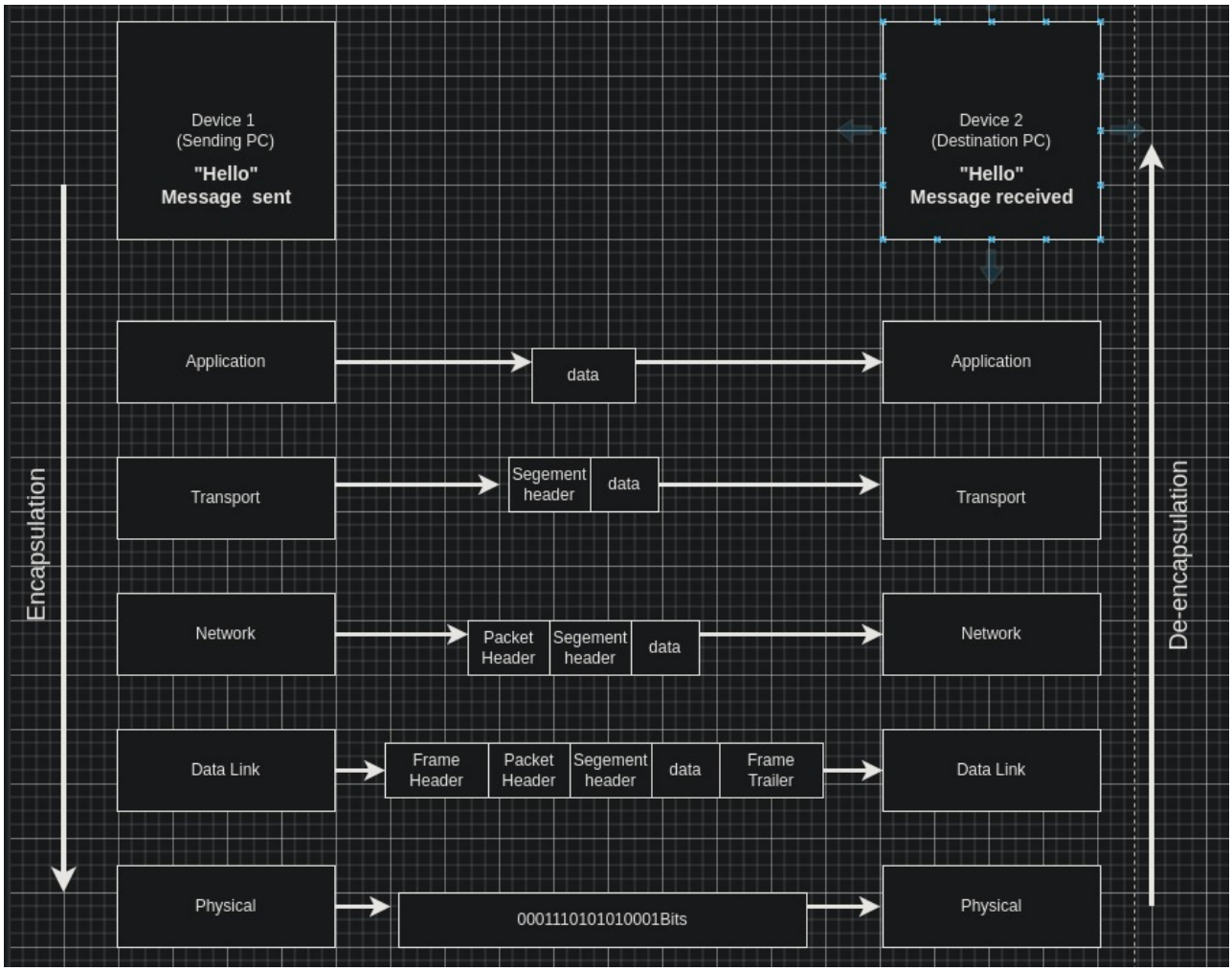
**Amjad** Today at 2:58 PM
In the De Encapsulation process
I receive the transmitted signals.
Signals are demodulated to extract digital data.
I detect the errors, if any.
Extracted data is passed to higher layers for processing. (edited)
🔥 2 💯 2 👌 1 👍 1 ❤️ 1

**H44mid** Today at 2:58 PM
As the data link layer, I receive the decapsulated data from the physical layer. This data could be the word "hello" that was originally encapsulated and transmitted. My role is to further process and prepare the data for the network layer.

I first verify the integrity of the received data using error detection mechanisms like checksums or CRC. If the data is valid, I strip off any remaining data link layer headers or trailers. The resulting payload data, which is the original "hello" message, is then encapsulated in a network layer protocol data unit (PDU), such as an IP packet. This encapsulated data is then passed up to the network layer for routing and further processing.
💯 2 👍 3 🔥 2

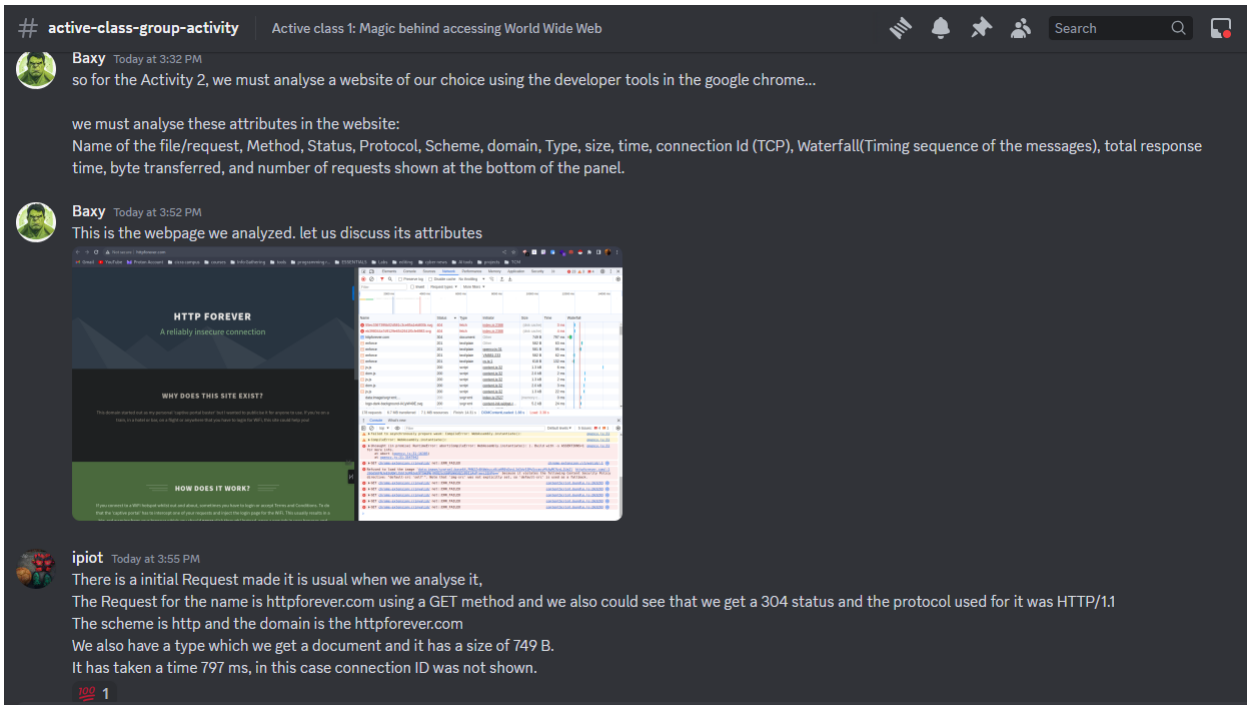


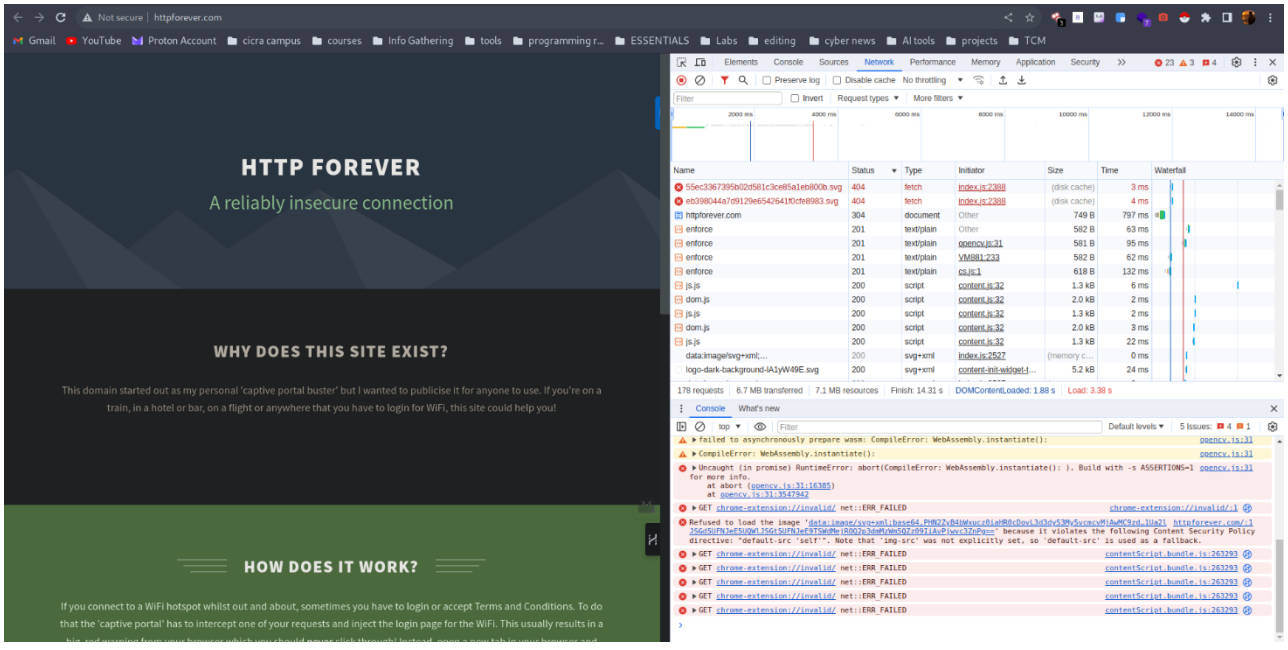

active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
Amjad Today at 3:19 PM
Yeah sure each of us can give help with there own layers they discussed and make a complete diagram
@Amjad Yeah sure each of us can give help with there own layers they discussed and make a complete diagram
Baxy Today at 3:21 PM
i agree
So guys we have successfully drawn the diagram...
Device 1 (Sending PC) "Hello" Message sent
Device 2 (Destination PC) "Hello" Message received
Application
data
Transport
Segement header
Network
Packet Header
Data Link
Frame Header
Frame Trailer
Physical
0001110101010001Bits
Encapsulation
De-encapsulation
2
So, Warriors of networking.... so far we have completed the quest 1 (activity 1) successfully. but hold on, Dont't kick back just yet, we're juz getting started in this journey... this is juz the beginning... we have still two more realms to conquer.... so what are we waiting for.... CHARGEEEEEE!!!!

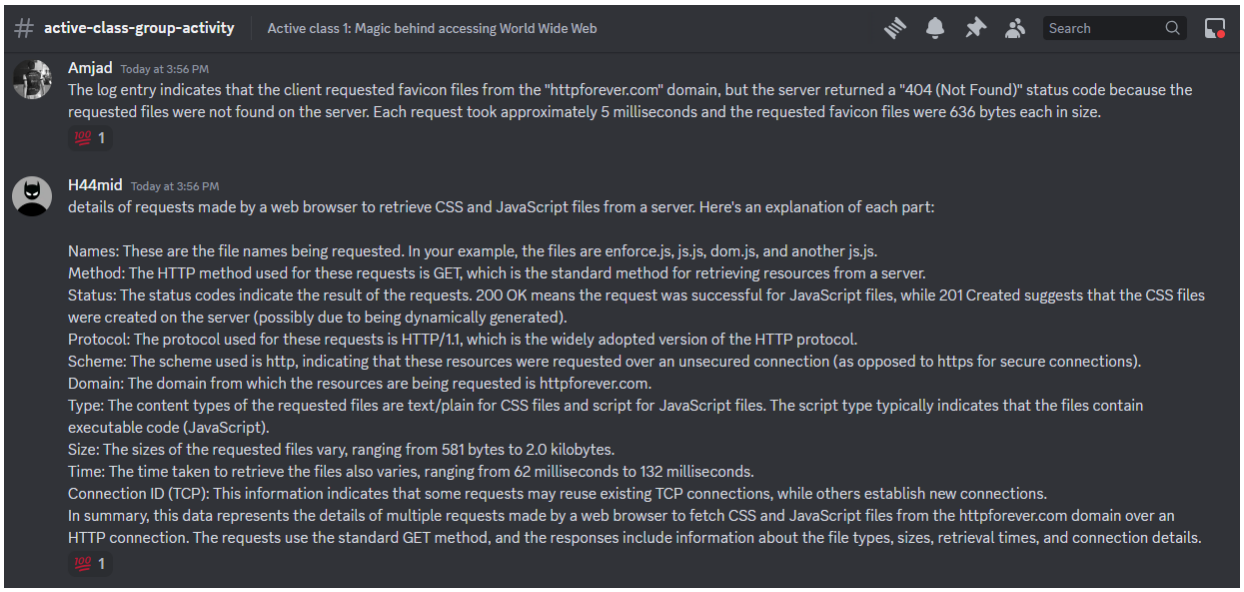

Device 1
(Sending PC)
"Hello"
Message sent
Device 2
(Destination PC)
"Hello"
Message received
Application
data
Application
Transport
Segement
header
data
Transport
Network
Packet
Header
Segement
header
data
Network
Data Link
Frame
Header
Packet
Header
Segement
header
data
Frame
Trailer
Data Link
Physical
0001110101010001Bits
Physical
Encapsulation
De-encapsulation

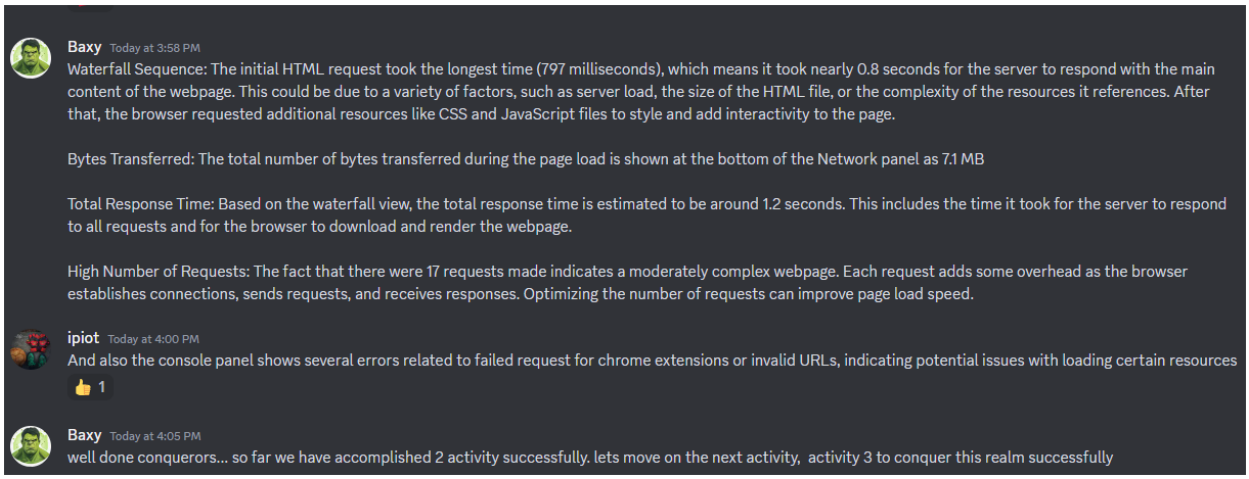
## Activity 2

**We used Wireshark to analyze the packet transfers occurring in a device when browsing the internet. We used a webpage's developers mode and analyzed the features of the module. (below are the evidences).**



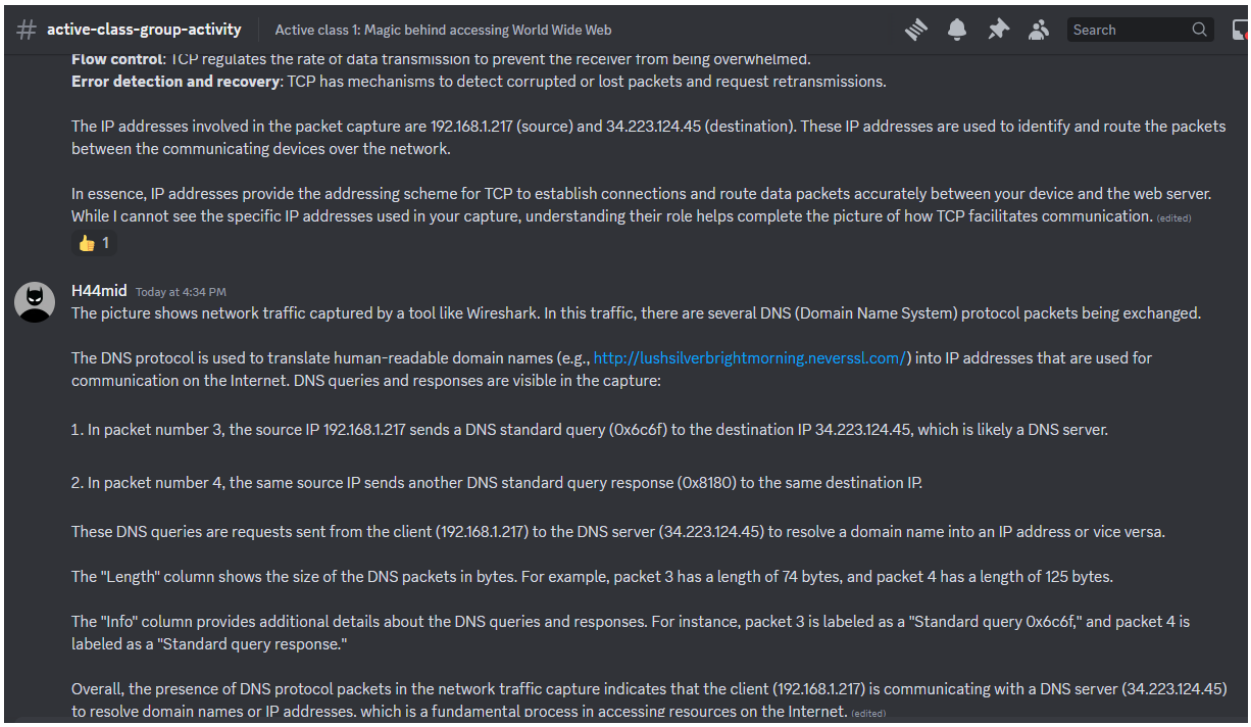




active-class-group-activity
Active class 1: Magic behind accessing World Wide Web
Search
Amjad Today at 3:56 PM
The log entry indicates that the client requested favicon files from the "httpforever.com" domain, but the server returned a "404 (Not Found)" status code because the requested files were not found on the server. Each request took approximately 5 milliseconds and the requested favicon files were 636 bytes each in size.
💯 1
H44mid Today at 3:56 PM
details of requests made by a web browser to retrieve CSS and JavaScript files from a server. Here's an explanation of each part:
Names: These are the file names being requested. In your example, the files are enforce.js, js.js, dom.js, and another js.js.
Method: The HTTP method used for these requests is GET, which is the standard method for retrieving resources from a server.
Status: The status codes indicate the result of the requests. 200 OK means the request was successful for JavaScript files, while 201 Created suggests that the CSS files were created on the server (possibly due to being dynamically generated).
Protocol: The protocol used for these requests is HTTP/1.1, which is the widely adopted version of the HTTP protocol.
Scheme: The scheme used is http, indicating that these resources were requested over an unsecured connection (as opposed to https for secure connections).
Domain: The domain from which the resources are being requested is httpforever.com.
Type: The content types of the requested files are text/plain for CSS files and script for JavaScript files. The script type typically indicates that the files contain executable code (JavaScript).
Size: The sizes of the requested files vary, ranging from 581 bytes to 2.0 kilobytes.
Time: The time taken to retrieve the files also varies, ranging from 62 milliseconds to 132 milliseconds.
Connection ID (TCP): This information indicates that some requests may reuse existing TCP connections, while others establish new connections.
In summary, this data represents the details of multiple requests made by a web browser to fetch CSS and JavaScript files from the httpforever.com domain over an HTTP connection. The requests use the standard GET method, and the responses include information about the file types, sizes, retrieval times, and connection details.
💯 1


Baxy Today at 3:58 PM
Waterfall Sequence: The initial HTML request took the longest time (797 milliseconds), which means it took nearly 0.8 seconds for the server to respond with the main content of the webpage. This could be due to a variety of factors, such as server load, the size of the HTML file, or the complexity of the resources it references. After that, the browser requested additional resources like CSS and JavaScript files to style and add interactivity to the page.
Bytes Transferred: The total number of bytes transferred during the page load is shown at the bottom of the Network panel as 7.1 MB
Total Response Time: Based on the waterfall view, the total response time is estimated to be around 1.2 seconds. This includes the time it took for the server to respond to all requests and for the browser to download and render the webpage.
High Number of Requests: The fact that there were 17 requests made indicates a moderately complex webpage. Each request adds some overhead as the browser establishes connections, sends requests, and receives responses. Optimizing the number of requests can improve page load speed.
ipiot Today at 4:00 PM
And also the console panel shows several errors related to failed request for chrome extensions or invalid URLs, indicating potential issues with loading certain resources
👍 1
Baxy Today at 4:05 PM
well done conquerors... so far we have accomplished 2 activity successfully. lets move on the next activity, activity 3 to conquer this realm successfully

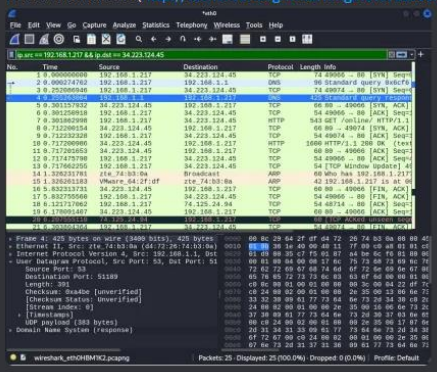
## Activity 3

**In the third activity we accessed a web page and accessed four main protocols and explained in depth using the Wireshark. We analyzed the packet transfers and analyzed the protocols and explained how it works separately. (below are the evidences for activity 3).**



**Baxy** Today at 4:30 PM

This is the website (http://lushsilverbrightmorning.neverssl.com/) we analyzed. let begin our discussion



There are 4 protocols in this section. I'll explain about the TCP protocol

The Transmission Control Protocol (TCP) is a fundamental protocol observed in the captured network traffic you provided. TCP is a connection-oriented protocol that establishes a reliable connection between two devices on a network before exchanging data. This ensures that the data is delivered accurately, in order, and without errors.

Some key features of TCP:

**Connection establishment**: Before data transfer, TCP establishes a connection through a three-way handshake process.
**Reliable data transfer**: TCP uses sequence numbers and acknowledgments to ensure reliable data delivery.
**Flow control**: TCP regulates the rate of data transmission to prevent the receiver from being overwhelmed.

**Amjad** Today at 4:39 PM

In the network packet we captured we can see 4 different types of protocols

Specifically for the HTTP protocol, we can see the following:

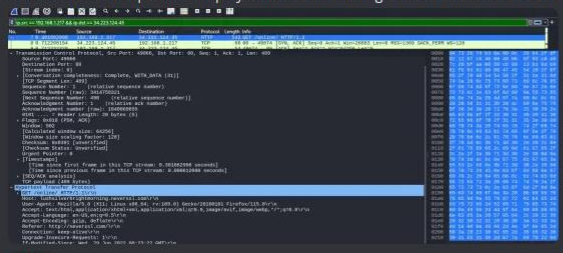
Entry 7: An HTTP GET request for "/online/" on port 80 (the default HTTP port) from the source IP 192.168.1.217 to the destination IP 34.223.124.45.
Entry 8: Another HTTP GET request for "/story/" on port 80 from the same source and destination IPs.
These entries indicate that the client (192.168.1.217) is making HTTP GET requests to retrieve content from a web server (34.223.124.45). The GET requests are for specific resources "/online/" and "/story/", which could be web pages or other web content hosted on that server.

HTTP is a protocol used for communication between web browsers (clients) and web servers, primarily for retrieving web pages, images, videos, and other resources that make up websites. In this case, the client is sending HTTP GET requests to the server to fetch the requested web content.

👏 2

**ipiot** Today at 4:41 PM

Here at this point the source ip (PC ip address) is 192.168.1.217 and the website server ip address is 34.223.124.45, and in this image we have a GET request from the source to the destination to get the HTTP confirmation, which has a destination port of 80 and the source port is 49066 there is a tcp segment send with the length of 489. The time stamp is also displayed in the above image as the first frame has taken about 0.301862998 and also it displays the previous frame.



💯 2

# active-class-group-activity | Active class 1: Magic behind accessing World Wide Web

Search

**mabrook** Today at 4:42 PM

In the above shown diagram there are 4 types of protocol. let me explain about the udp protocol, types and how it works.

UDP is a connectionless and unreliable transport layer protocol used for transmitting data packets over the internet. In the capture, we can see UDP being used for DNS (Domain Name System) queries and responses.

Frame 3 shows a DNS standard query sent over UDP from the client (192.168.1.217) to the DNS server (34.223.124.45). Frame 4 shows the corresponding DNS standard query response sent back over UDP from the server to the client.

The key characteristics of UDP as seen in this capture are:

Connectionless: UDP is connectionless, meaning there is no dedicated end-to-end connection established between the client and server. Each UDP packet is treated independently.
Unreliable: UDP provides no guarantees for delivery, ordering, or duplicate protection of packets. The DNS query and response are sent in individual UDP datagrams without any reliability mechanisms.
Low Overhead: UDP has a simple header structure and minimal overhead compared to TCP, making it suitable for applications like DNS where speed is important, and some packet loss is acceptable.
Best Effort Delivery: UDP packets are sent without any delivery guarantees. If a DNS response is lost or corrupted, the client may need to retransmit the query.

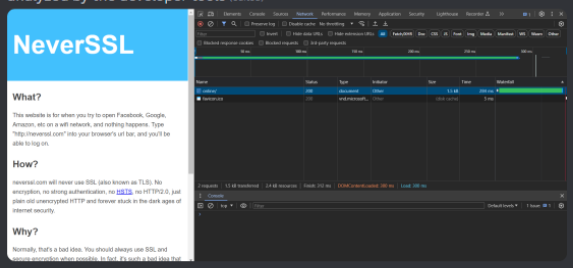
Therefore in summary, UDP is used in this capture for DNS queries and responses because it provides a fast, low-overhead method for sending and receiving small amounts of data without the need for a dedicated connection or reliability mechanisms. The trade-off is that UDP offers no guarantees for delivery, ordering, or duplicate protection of packets, which is acceptable for DNS as long as queries can be retransmitted if needed.

💯 2

# active-class-group-activity | Active class 1: Magic behind accessing World Wide Web

Search

**Baxy** Today at 4:51 PM
now lets discuss how the developer tools and the wireshark differ and have similarities in analyzing the packets. Below is the screenshot of the same website but analyzed by the developer tools (edited)




**ipiot** Today at 4:53 PM
In the visualisation we could see that there is a document and it has 200 as its status, the size of it is displayed as 1.5kb and it show the time it took to load was 284ms (edited)

**Baxy** Today at 4:56 PM
in this screenshot, we can see that the waterfall timeline offers a visual representation of the sequence and timing of network requests. It helps identify which resources are taking the longest to load and potential performance bottlenecks. We can analyze the time spent on different stages of the request-response cycle, such as DNS lookup, initial connection, SSL negotiation, server response time, and content download.

# active-class-group-activity | Active class 1: Magic behind accessing World Wide Web

Search

**H44mid** Today at 4:59 PM
In this case, the website being loaded is "NeverSSL," which intentionally uses unencrypted HTTP instead of secure HTTPS. The Network tab displays two network requests made by this website:

The first request is for "online/", which is likely the main HTML document. This request has a status code of 200 (OK), and its type is listed as "document".
The second request is for "favicon.ico", which is the website's favicon (the small icon displayed in the browser tab). This request also has a status code of 200 (OK), and its type is listed as "vnd.microsoft.icoOther".
The "Size" column shows the size of the resources transferred, and the "Time" column displays the time taken to load each resource.

At the bottom, there is a summary of the network activity, showing that there were 2 requests made, resulting in 15 KB of data transferred, and the total load time was 312 ms.

The Developer Tools also provide additional information and tools for debugging web applications, such as the Console panel visible in the image.

Overall, this image demonstrates the Network panel in the Developer Tools, which is a useful tool for web developers to analyze and optimize network requests and responses made by their web applications.

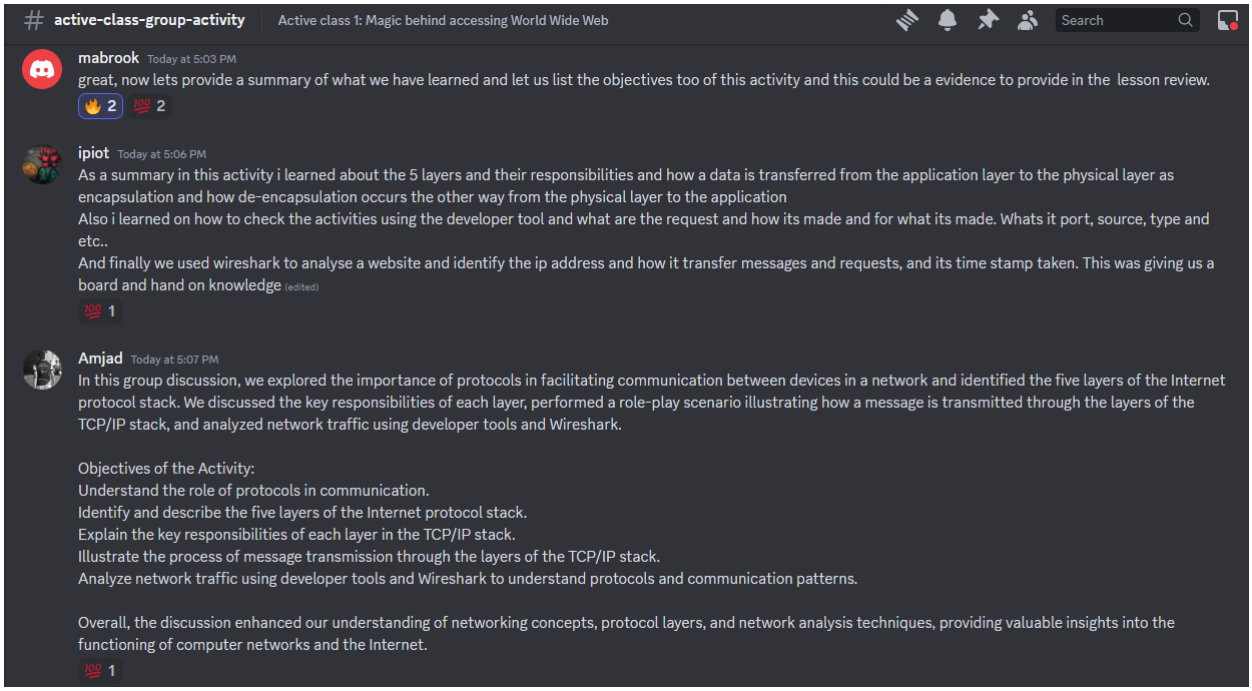
**Baxy** Today at 5:00 PM
there is another similarity

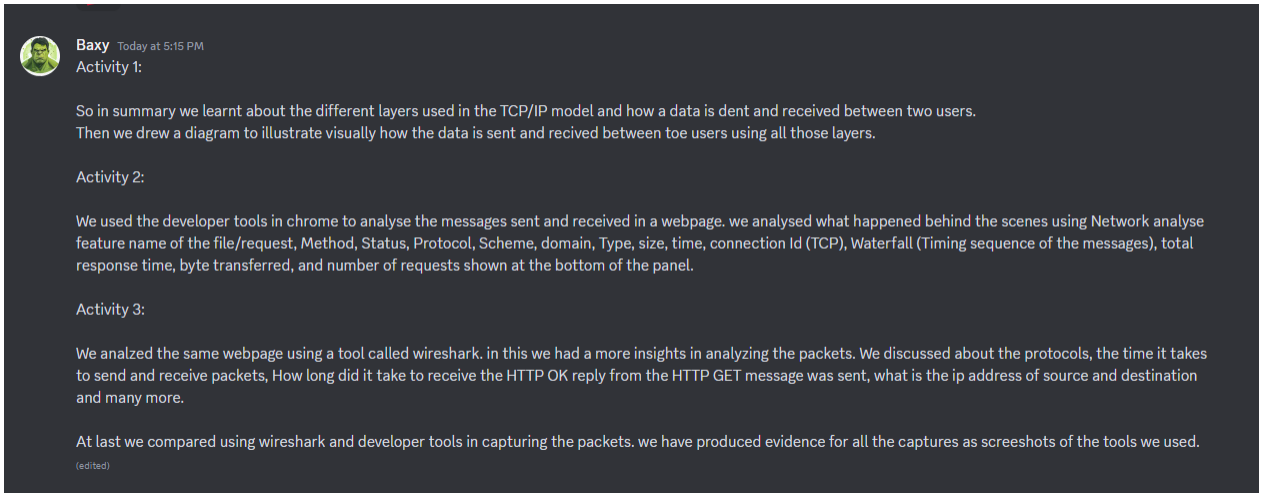
it is the Filtering and Sorting, Both Wireshark and Chrome's Developer Tools offer filtering and sorting capabilities to help you focus on specific network activity. In Wireshark, you can apply display filters to show only certain types of packets or protocols, while in Chrome's Developer Tools, you can filter resources based on criteria like type, domain, or text patterns in the URL or response body.

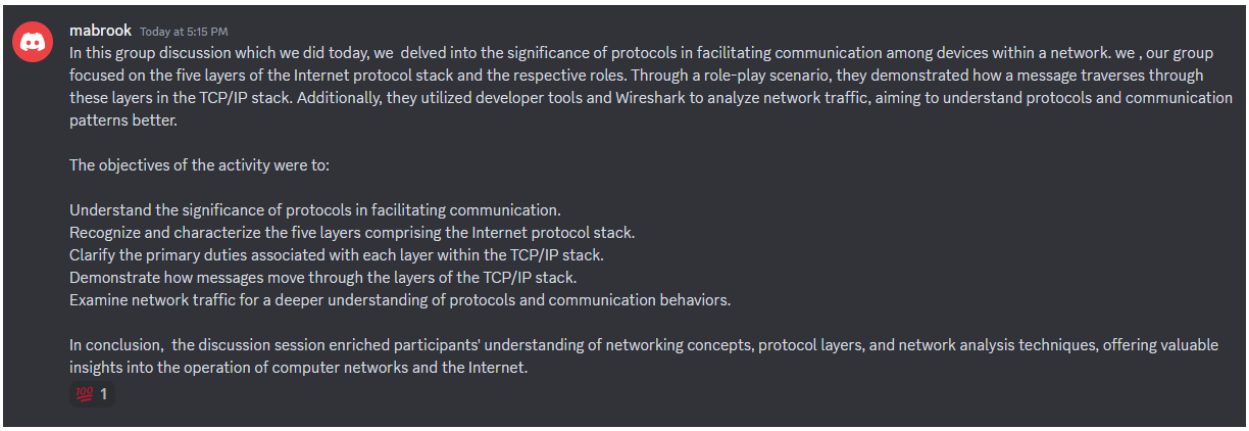
👍 2

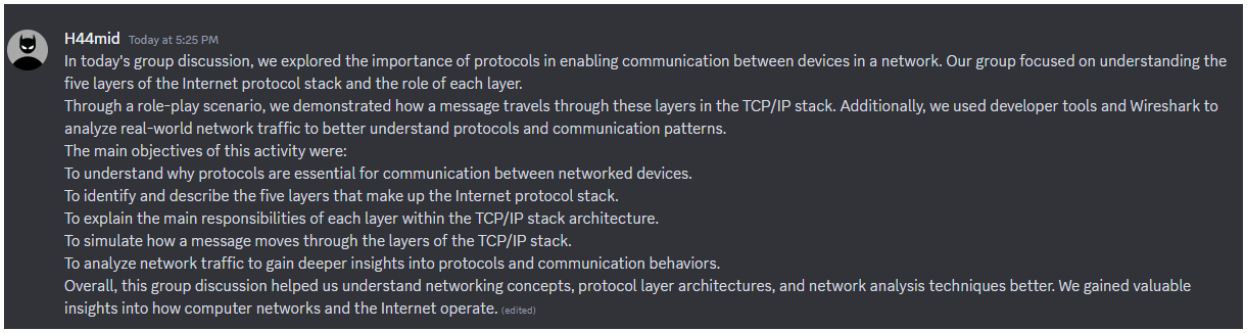
## Summary

**We, members in the group, have provided the summary related to what we have learned in module 1 and provided the objectives listed in this activity. ( evidence of submitting the summary and lesson review).**






**Overall, we as a group of 5 discussed the 3 activities using discord and analyzed a specific website by capturing the packets using Wireshark too and explained each sequence and protocols in depth and how these protocols work.**

## Members of the group.

- **Nirosh**
- **Iflal**
- **Amjad**
- **Haamid**
- **Mabrook.**

**Notes Evidence**

Network and communication

IPV4 – 32 bits long
IPv6 – 128 bits long

MAC – 48 bits long.

NAT translates Private IP addresses into Public IP address.

URL ⇄ IP

DNS helps to translate each other.

In the Internet, we Identify a device or a network by their IP address.

In a network, we identify a device with their MAC address.

7 layers in OSI model:

Application layer – Transformational decision making based on "Thing" Application and data.

Presentation – Custom Applications built using "Thing" data.

Session – Reporting, Mining, Machine learning.

TransPort – Bigdata, Horrerb & storage of "Thing" data.

Network – Cloud Infrastructure (Public, private, hybrid, managed)

Data link – Communications, protocol, Network, WiFi, Telecom.

Physical – Devices, Sensors, Controllers, etc.

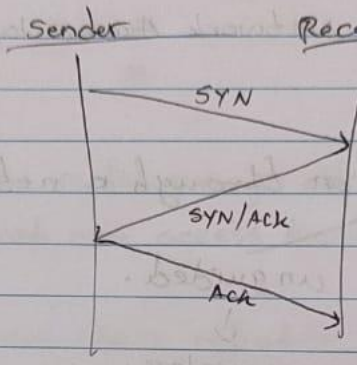
| | |
|---|---|
| Application | data. |
| ↓ | ↓ |
| Transport | data L4 header (segment). |
| ↓ | ↓ |
| Network | data L4 header, L3 header (Packet). |
| ↓ | ↓ |
| Data link | data L4, L3, L2 ~~header~~ trailer (Frame. |
| ↓ | |
| Physical | |

Protocol stack.

Layer name, Protocol, Protocol data unit, addressing.

| TCP | UDP |
| --- | --- |
| Reliable | Faster. |

## TCP

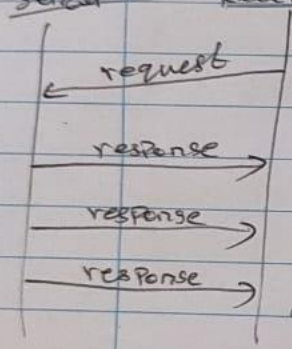



Slower but more reliable transfer

Typical application:
- File transfer Protocol (FTP)
- web browsing
- Email

unicast

## UDP




Faster but not guaranteed transfers.

Typical application
- Live streaming
- Online games
- VoIP.

unicast
Multicast
broadcast.

443 - HTTPS
80 - HTTP.

TCP/IP layers - 5 layers

| | | |
|---|---|---|
| ~~Application~~ | 4 | Application |
| | 3 | Transport |
| Application | 5 | Network |
| Transport | 4 | Data link |
| Network | 2 | physical. |
| ~~network access layer~~ | 1 | |

Application - It is the group of applications that let user access the network.

Transport - It provides a reliable data connection between two communicating devices.

~~Internet layer - It controls~~

Internet/network layer - controls the flow and routing of traffic to ensure data is sent speedily and accurately.

Network access layer/data link layer - handles the physical infrastructure that lets computer communicate with one another over the internet.

# OSI vs TCP/IP

All People seem to Need Data Processing

Application
Presentation
Session
→ Application

Transport → Transport

Network → Network

Data link
Physical
→ Network Access layer

Presentation and session are the ones that differentiate between OSI and TCP/IP.

## Data communication through a network

- guided
  - wired media
    - (Twisted) Pairs
    - (Coaxial) cables
    - (Fiber optic)
- unguided.
  - wireless media.
    - (wireless communication)
    - (satellite communication)
    - (Infrared communication)

Packet switching
Circuit switching.

## Packet vs Circuit

| | |
|---|---|
| Resource sharing | Dedicated resources |
| No call set up required. | Quality of service guaranteed. |
| Protocols needed for reliable data transfer, congestion control. | Not scalable at all. |
| can achieve circuit switching behaviours | with the cost employing special mechanism |

Rangana