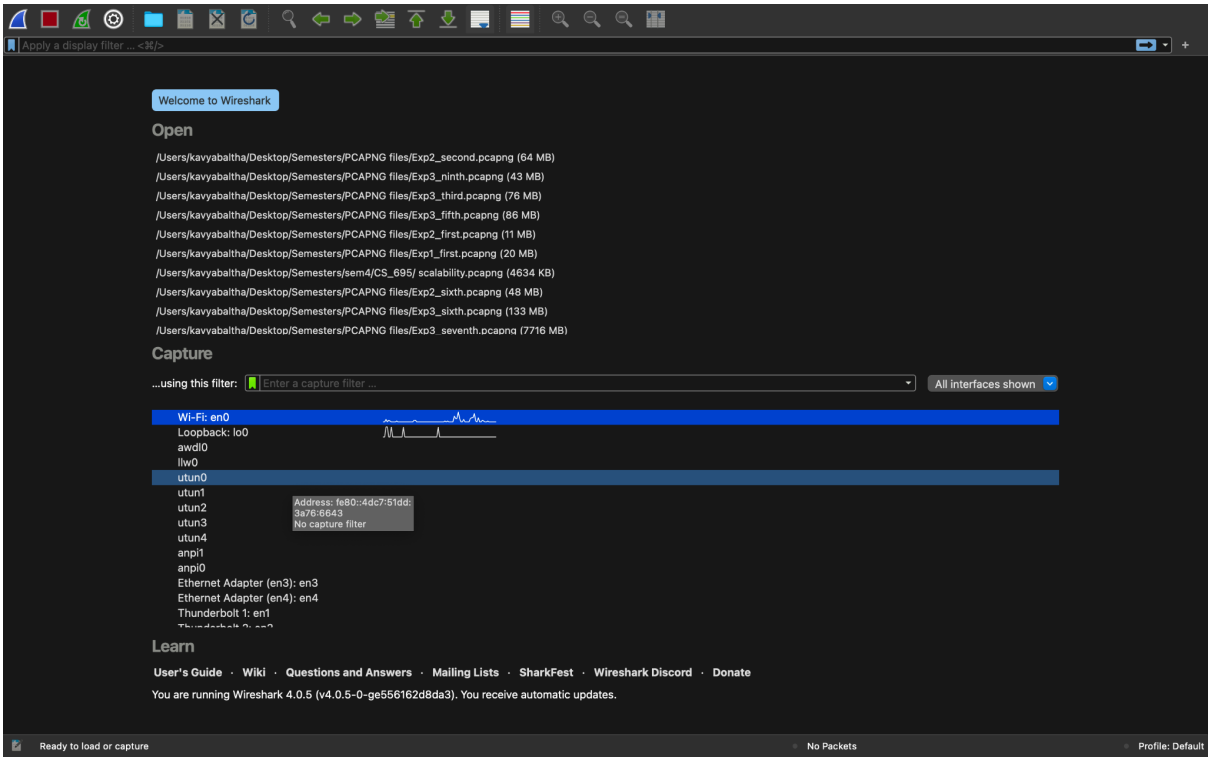
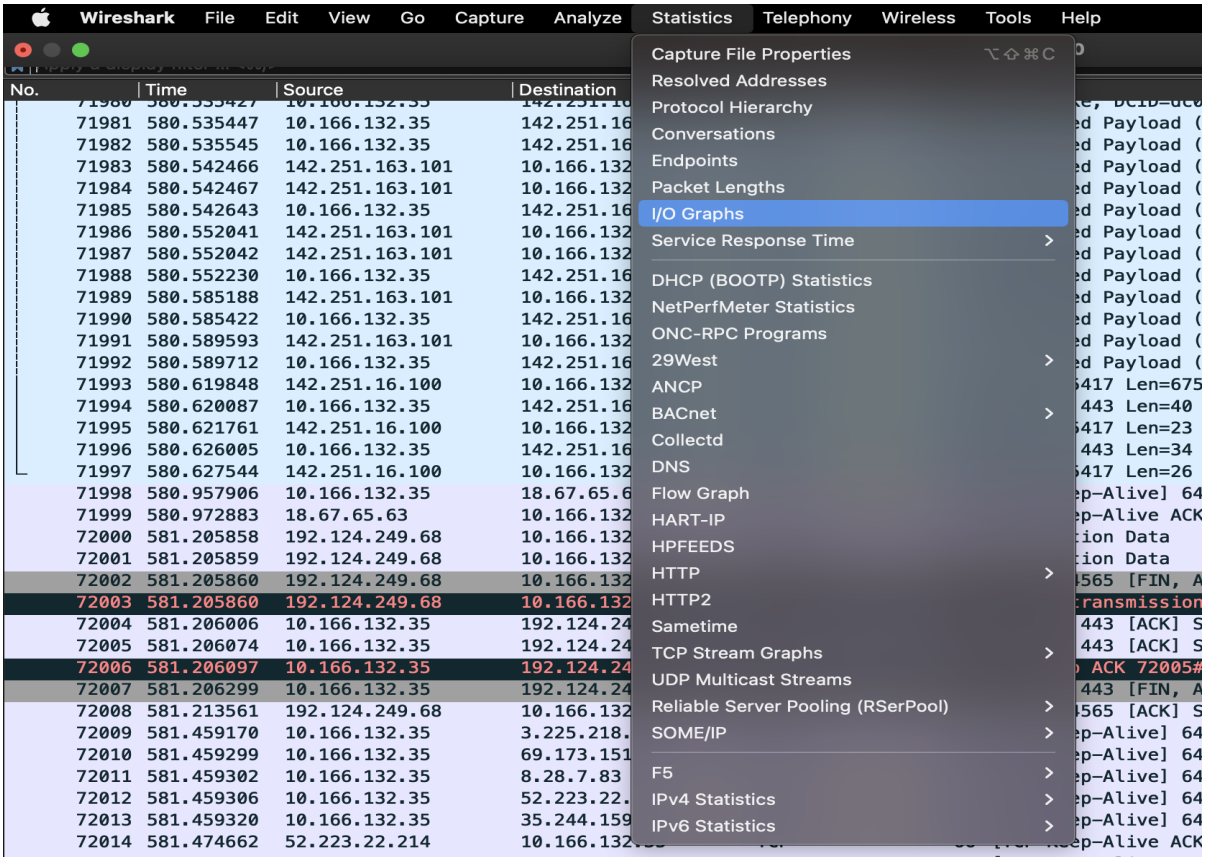
Network Measurement

Wireshark:
Wireshark is a popular network protocol analyzer that is used for troubleshooting, analysis, software and communication protocol development, and education. It allows you to capture and examine individual packets of data transmitted over a network. With Wireshark, you can capture network traffic in real-time or analyse saved packet capture files.

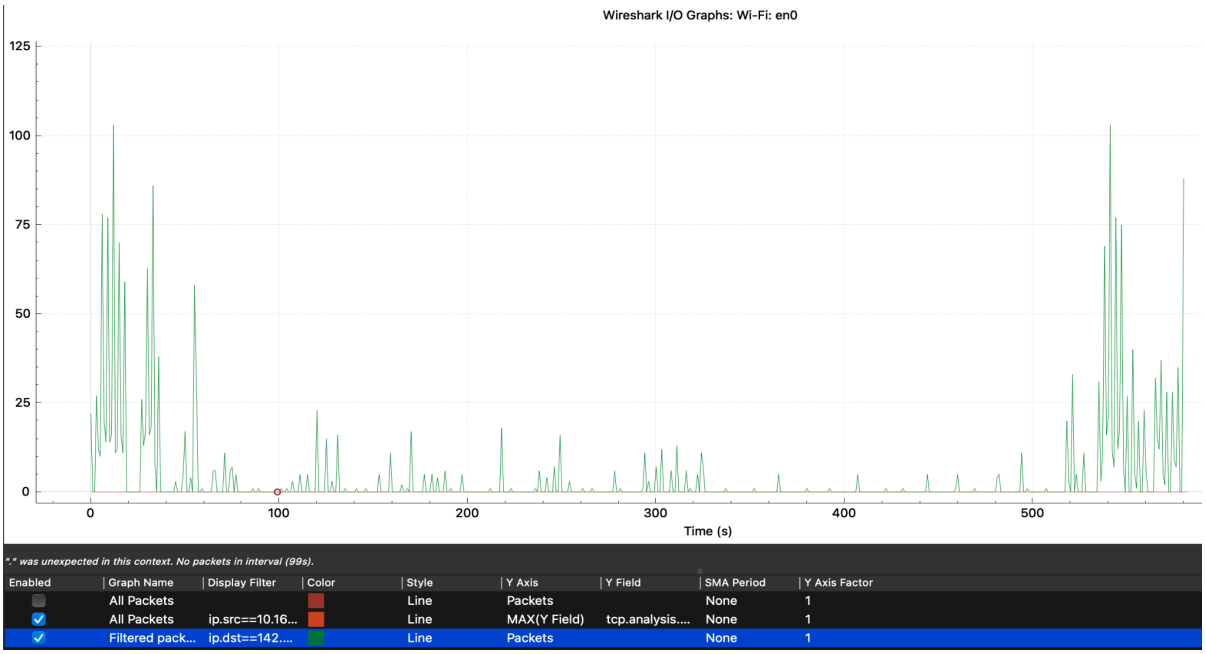
Step 1: To capture the network data select the wireless interface for which the data needs to be captured is shown below in the capture section and double click on it to start capturing and click the stop icon to stop the capture.



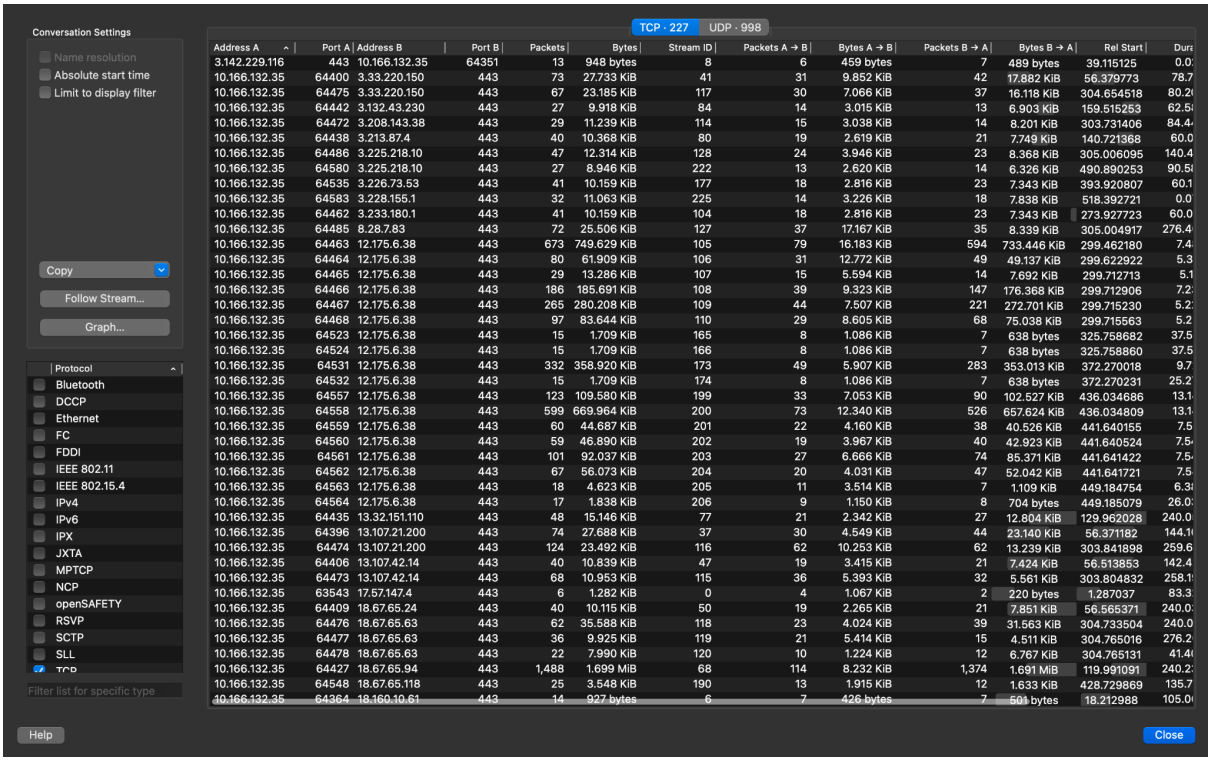
Step 2: The I/O Graph in Wireshark is a feature that provides a graphical representation of network traffic between two hosts. Once you have selected the conversation, you can create an I/O Graph by navigating to the Statistics menu and selecting I/O Graph. This will open a new window that displays the I/O Graph.



Step 3: The I/O Graph in Wireshark provides a range of customization options that allow you to select the data to be graphed, the time frame, and the type of graph to be used. This makes it easy to customise the graph to your specific needs. Adjust the source and destination ip address which needs to be analysed in the Display filter tab.

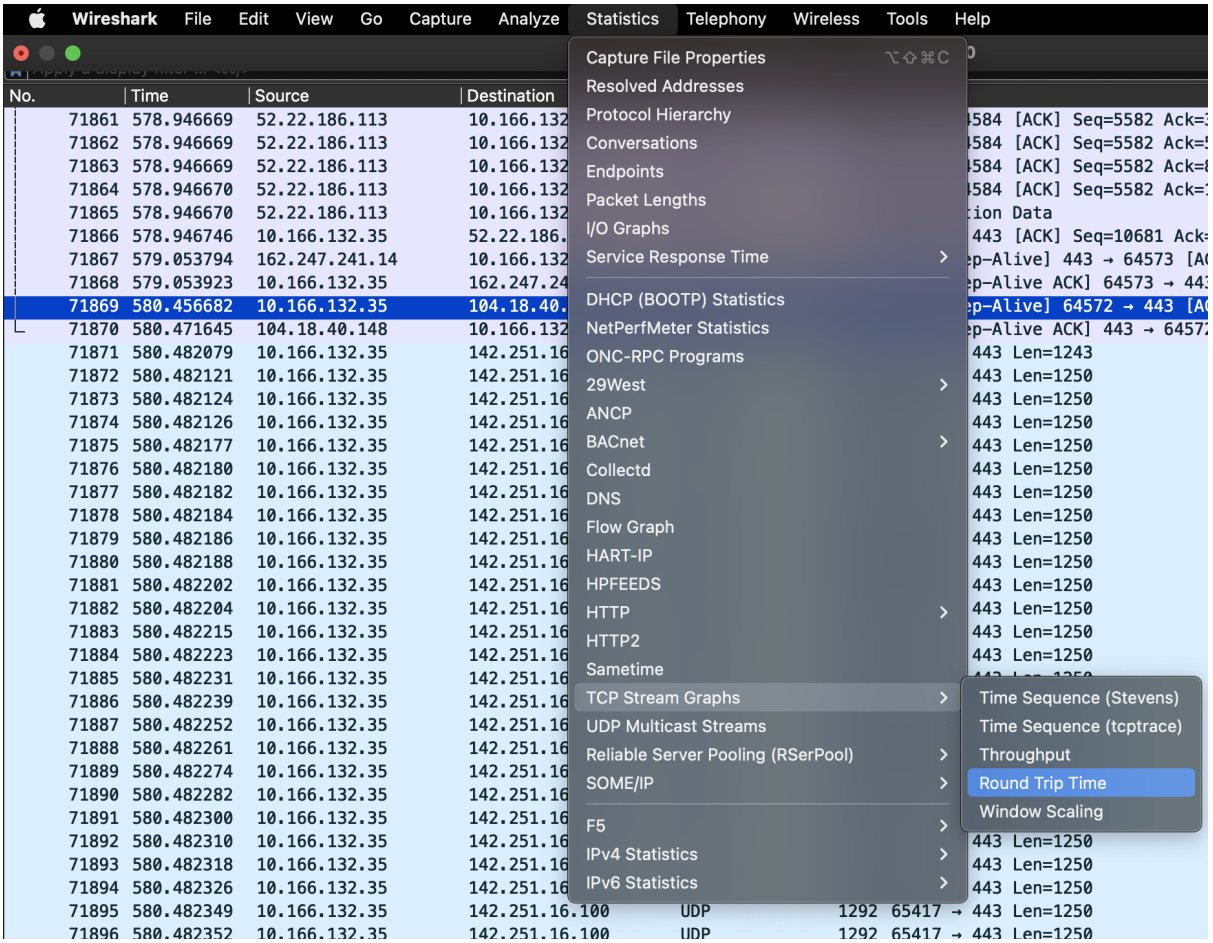


Step 4: The Conversation tab in Wireshark is a feature that displays a summary of the network traffic between different hosts in a captured network traffic trace. It provides a way to quickly analyze the communication patterns between hosts, including the number of packets and bytes sent, the duration of the conversation, and the protocols used. To access the Conversation tab, open a network traffic trace in Wireshark and click on the Statistics menu. From there, select Conversations. The Conversations window provides several customization options, such as filtering conversations based on protocol or host, sorting conversations by various criteria, and exporting the data to a CSV file for further analysis.

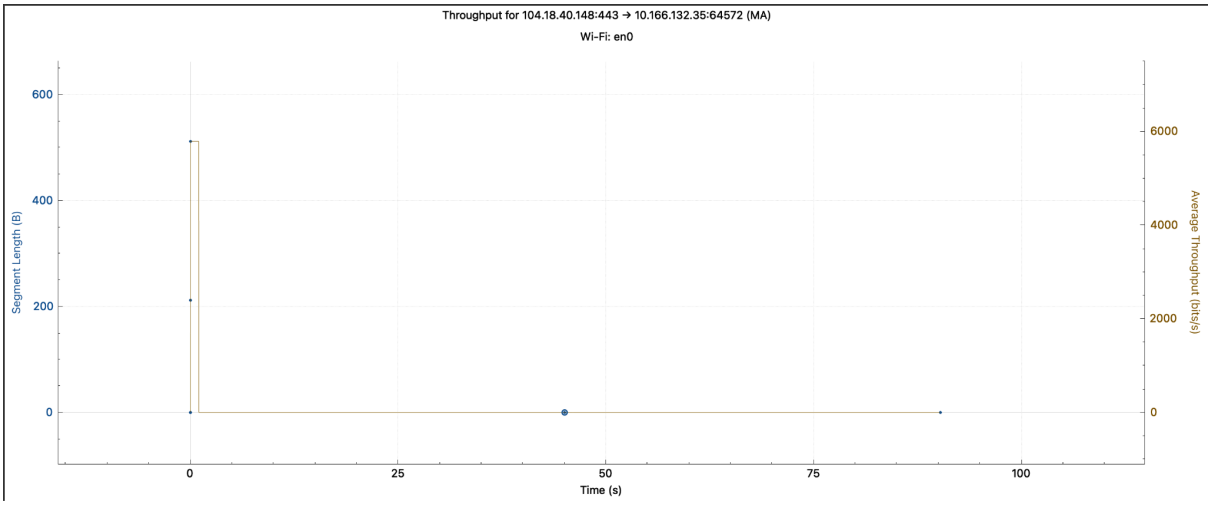


Step 5: The TCP Graphs feature in Wireshark provides a way to visualize the Round Trip Time (RTT) and throughput of a TCP stream over time. To access the RTT, throughput graphs, click statistics -> TCP stream graphs -> RTT/throughput graphs. Throughput refers to the amount of data that can be transmitted over a network or communication channel in a given amount of time. It is typically measured in bits per second (bps) or bytes per second (Bps). Throughput is an important measure of the performance of a network or communication channel.

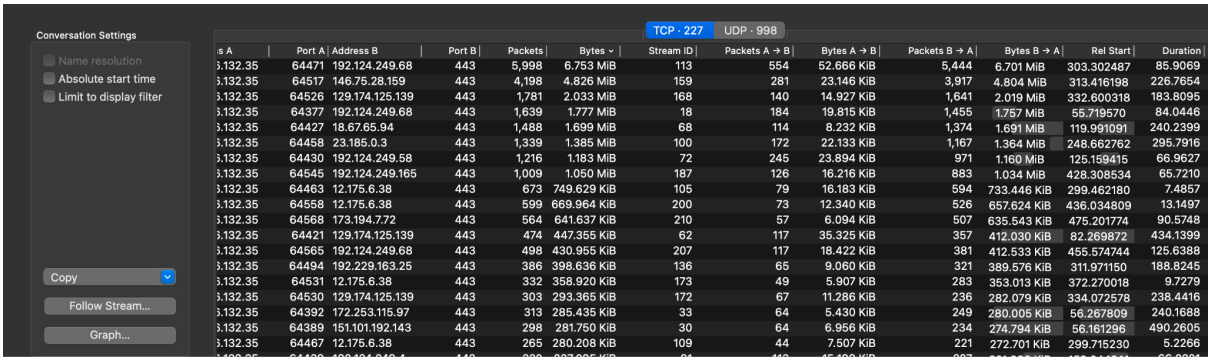
RTT stands for Round Trip Time, which is the amount of time it takes for a data packet to travel from the sender to the receiver and back again. RTT is typically measured in milliseconds (ms) or microseconds (μs), and it is an important metric for measuring network latency and determining network performance.



Step 6: The throughput graph displays the amount of data transferred over time and can help identify periods of high or low network utilisation. The throughput graph displays the data rate for each packet in the stream, as well as the average and maximum data rates. To interpret the graphs, look for patterns or trends over time. For example, a sudden drop in throughput may indicate network congestion.

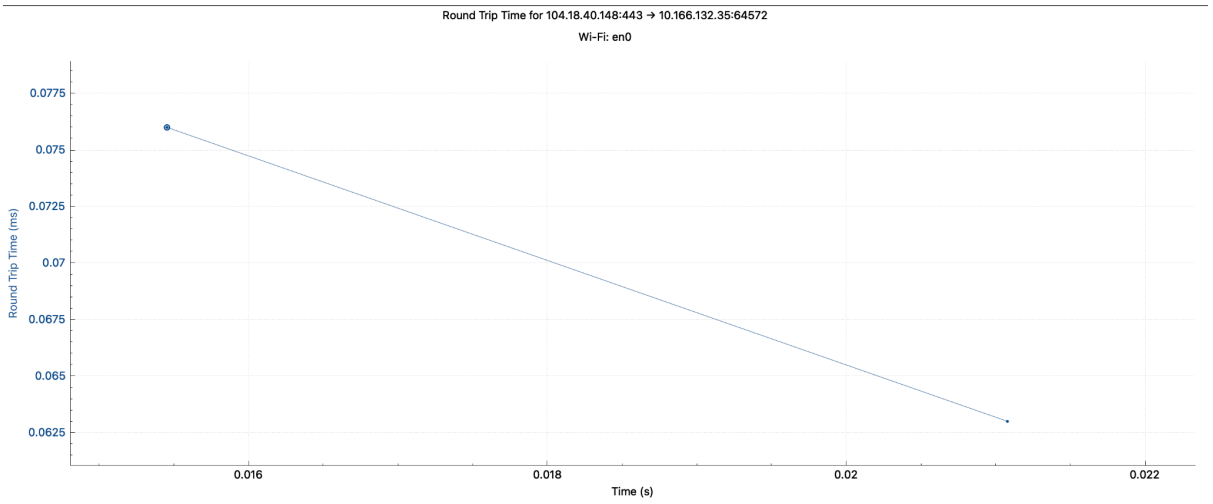


Steps to Calculate throughput : Open the conversations tab as said previously and select a conversation to be analysed. If you select the first conversation from the below. Throughput is calculated to be No. of bytes/duration i.e., 6.753 MB/ 85.9069



s A	Port A	Address B	Port B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
.132.35	64471	192.124.249.68	443	5,998	6.753 MiB	113	554	52.666 KiB	5,444	6.701 MiB	303.302487	85.9069
.132.35	64517	146.75.28.159	443	4,198	4.826 MiB	159	281	23.146 KiB	3,917	4.804 MiB	313.416198	226.7654
.132.35	64526	129.174.125.139	443	1,781	2.033 MiB	168	140	14.927 KiB	1,641	2.019 MiB	332.600318	183.8095
.132.35	64377	192.124.249.68	443	1,639	1.777 MiB	18	184	19.815 KiB	1,455	1.757 MiB	55.719570	84.0446
.132.35	64427	18.67.65.94	443	1,488	1.699 MiB	68	114	8.232 KiB	1,374	1.691 MiB	119.991091	240.2399
.132.35	64458	23.185.0.3	443	1,339	1.385 MiB	100	172	22.133 KiB	1,167	1.364 MiB	248.662762	295.7916
.132.35	64430	192.124.249.58	443	1,216	1.183 MiB	72	245	23.894 KiB	971	1.160 MiB	125.159415	66.9627
.132.35	64545	192.124.249.165	443	1,009	1.050 MiB	187	126	16.216 KiB	883	1.034 MiB	428.308534	65.7210
.132.35	64463	12.175.6.38	443	673	749.629 KiB	105	79	16.183 KiB	594	733.446 KiB	299.462180	7.4857
.132.35	64558	12.175.6.38	443	599	669.964 KiB	200	73	12.340 KiB	526	657.624 KiB	436.034809	13.1497
.132.35	64568	173.194.7.72	443	564	641.637 KiB	210	57	6.094 KiB	507	635.543 KiB	475.201774	90.5748
.132.35	64421	129.174.125.139	443	474	447.355 KiB	62	117	35.325 KiB	357	412.030 KiB	82.269872	434.1399
.132.35	64565	192.124.249.68	443	498	430.955 KiB	207	117	18.422 KiB	381	412.533 KiB	455.574744	125.6388
.132.35	64494	192.229.163.25	443	386	398.636 KiB	136	65	9.060 KiB	321	389.576 KiB	311.971150	188.8245
.132.35	64531	12.175.6.38	443	332	358.920 KiB	173	49	5.907 KiB	283	353.013 KiB	372.270018	9.7279
.132.35	64530	129.174.125.139	443	303	293.365 KiB	172	67	11.286 KiB	236	282.079 KiB	334.072578	238.4416
.132.35	64392	172.253.115.97	443	313	285.435 KiB	33	64	5.430 KiB	249	280.005 KiB	56.267809	240.1688
.132.35	64389	151.101.192.143	443	298	281.750 KiB	30	64	6.956 KiB	234	274.794 KiB	56.161296	490.2605
.132.35	64467	12.175.6.38	443	265	280.208 KiB	109	44	7.507 KiB	221	272.701 KiB	299.715230	5.2266

Step 7: The RTT graph displays the time it takes for a packet to travel from the sender to the receiver and back again. It shows the variation in the RTT over time, which can indicate network latency or congestion. The RTT graph displays the minimum, maximum, and average RTT values for each packet in the stream, as well as the overall RTT for the stream.



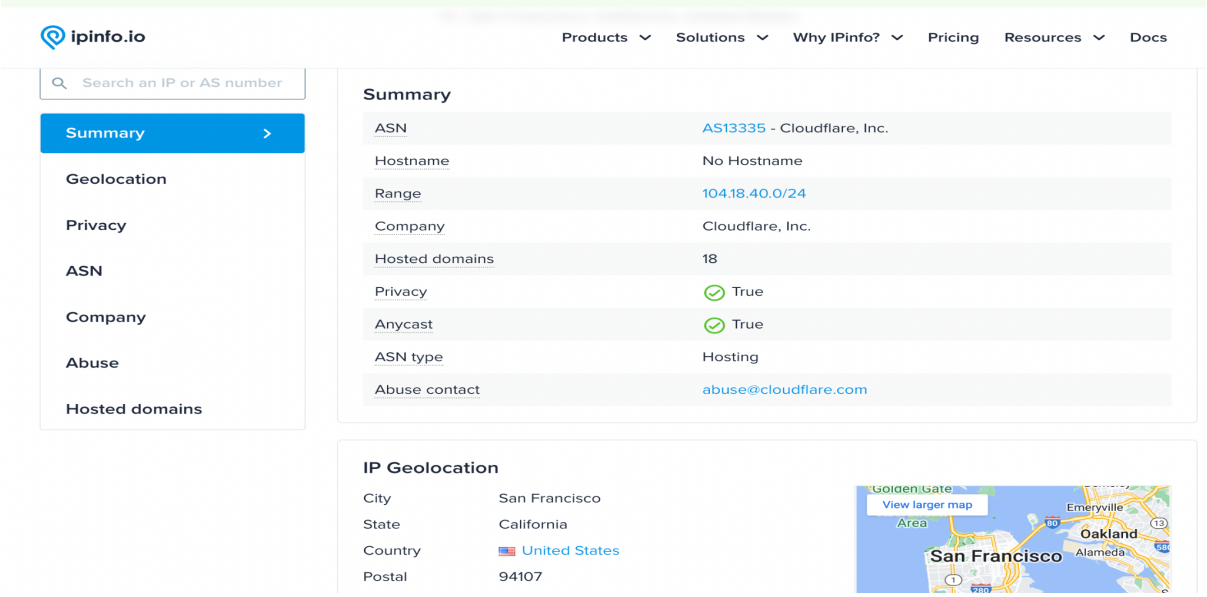
Wireshark's ability to display packet details such as packet size, packet timing, and any packet loss or retransmission can provide a deeper understanding of the network traffic. This feature helps in identifying the root cause of network issues. Wireshark also has the ability to decode and display a wide range of protocols, making it a valuable tool for analysing network traffic in various network environments. Regenerate response

References:

https://www.wireshark.org/docs/

ipinfo.io :
ipinfo.io is a simple and free web API that provides a variety of information about an IP address, including its geolocation, Autonomous System Number (ASN), and whether anycast routing is used for the server hosting that IP address. The response from the API will be a JSON object containing various information about the IP address, including its location, ASN, and anycast status.

Select an ip address to check its details. Example ip address: 104.18.40.148.
Go to https://ipinfo.io/104.18.40.148



Reference: https://ipinfo.io/

Ping:

When you run the command "ping 104.18.40.148" in the command prompt or terminal, it sends an ICMP echo request packet to the specified IP address (in this case, 104.18.40.148), and waits for a response from the host.

The output of the ping command provides information about the RTT (Round Trip Time) between the sender and the receiver, as well as the packet loss rate, which is the percentage of packets that are lost during the transmission.

The output typically includes the following information:
The IP address of the host being pinged (104.18.40.148)
The size of the packet being sent
The number of packets sent and received

The minimum, maximum, and average RTT in milliseconds
The percentage of packet loss

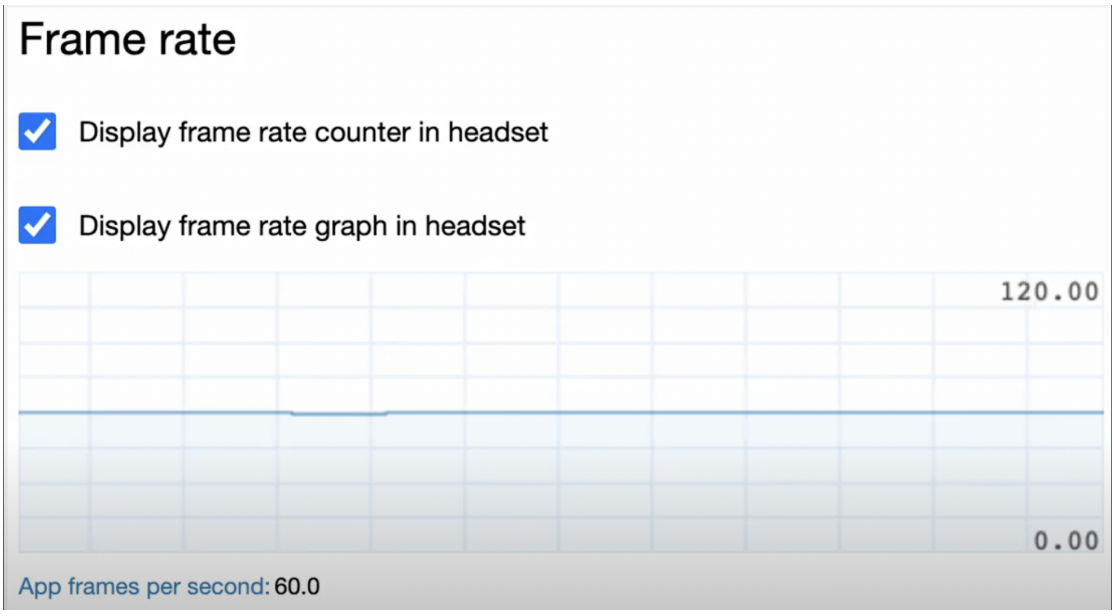
Output:

Pinging 104.18.40.148 with 32 bytes of data:
Reply from 104.18.40.148: bytes=32 time=7ms TTL=57
Reply from 104.18.40.148: bytes=32 time=8ms TTL=57
Reply from 104.18.40.148: bytes=32 time=6ms TTL=57
Reply from 104.18.40.148: bytes=32 time=7ms TTL=57

Ping statistics for 104.18.40.148:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 8ms, Average = 7ms

Frame rate:

Frame rate refers to the number of frames per second (fps) that are rendered and displayed on the user's device. A higher frame rate provides more immersive and realistic experience, whereas a lower frame rate causes lag, and poor user satisfaction. One can identify the Frame rate while using hololens or when connected to Windows Device Portal.

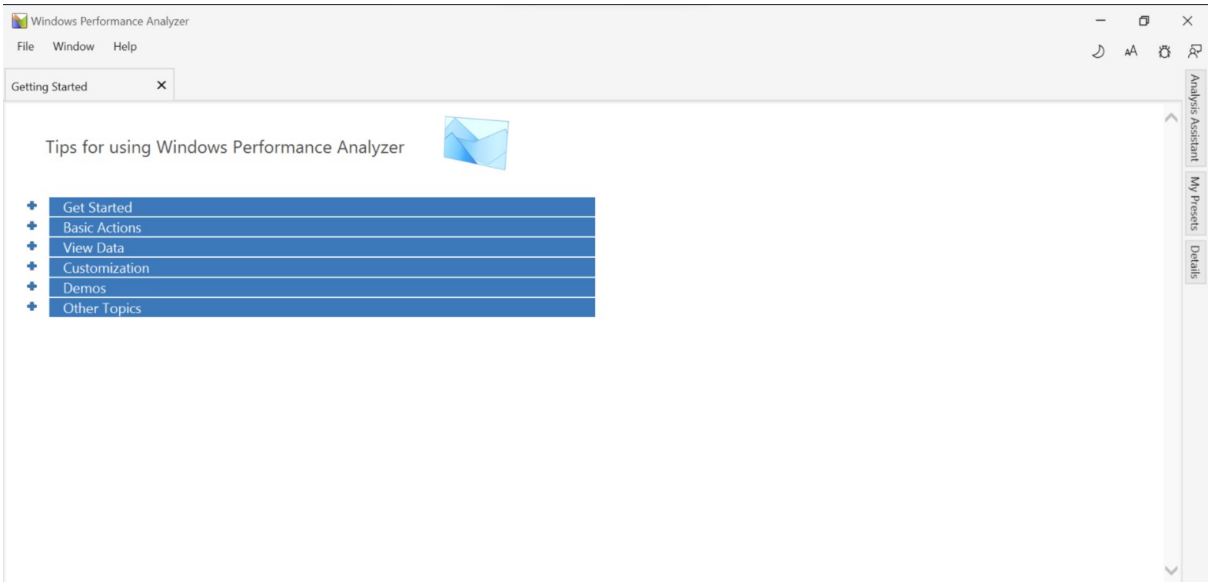


WPA:

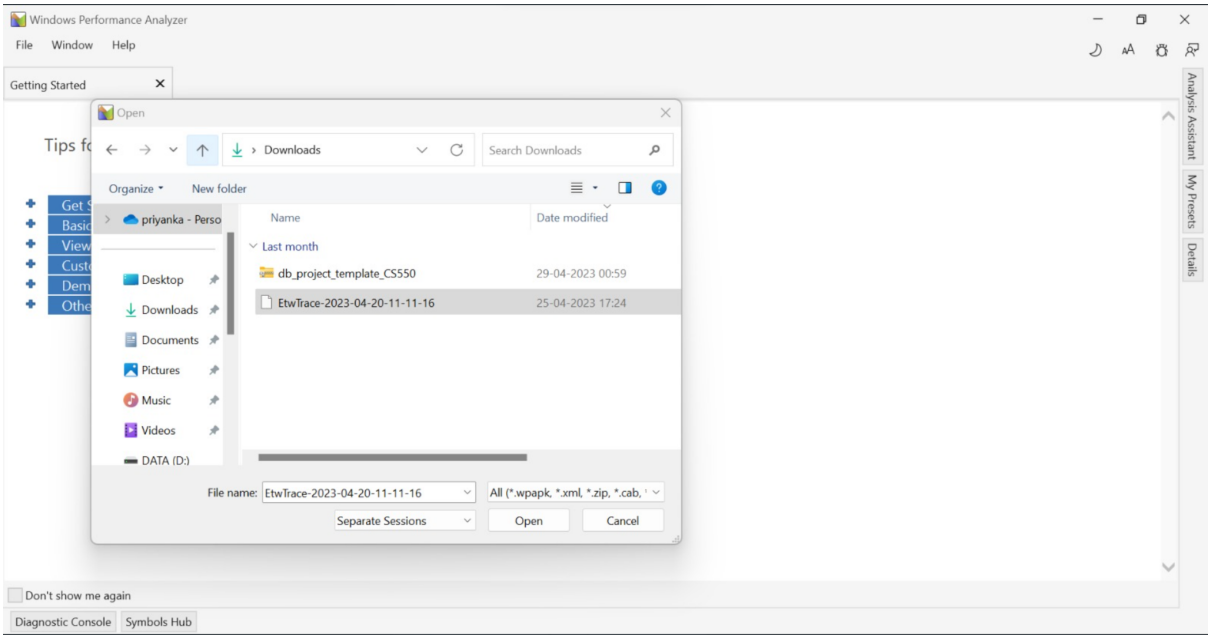
Capturing network traffic from the Hololens device using the Windows Device Portal and analysing it with a WPA tool can provide valuable insights into the network behaviour of the device

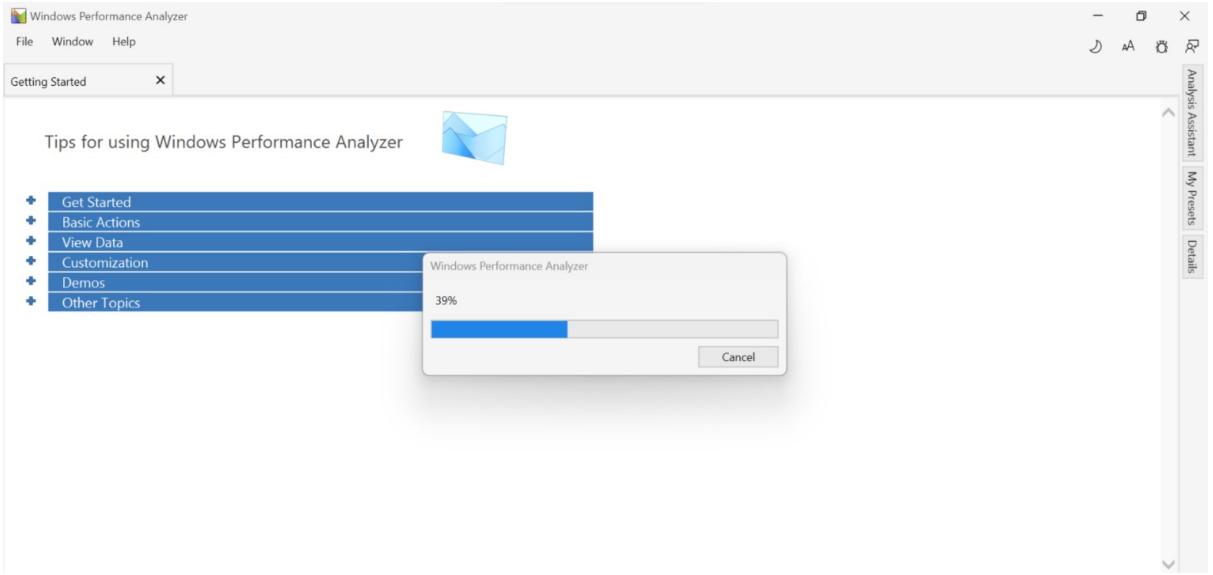
Analysing CPU utilisation: One of the insights that you can gain from a WPA analysis of Hololens network traffic is the amount of CPU capacity being used by the device. By analysing the captured packets and generating graphs of CPU utilisation over time, you can identify periods when the device is using a lot of CPU capacity, and determine the causes of this usage. For example, you might discover that the Hololens is using a lot of CPU capacity during certain types of network activity, or that a particular app or process is causing high CPU usage.

Step 1: Open the WPA tool. It looks like the below.

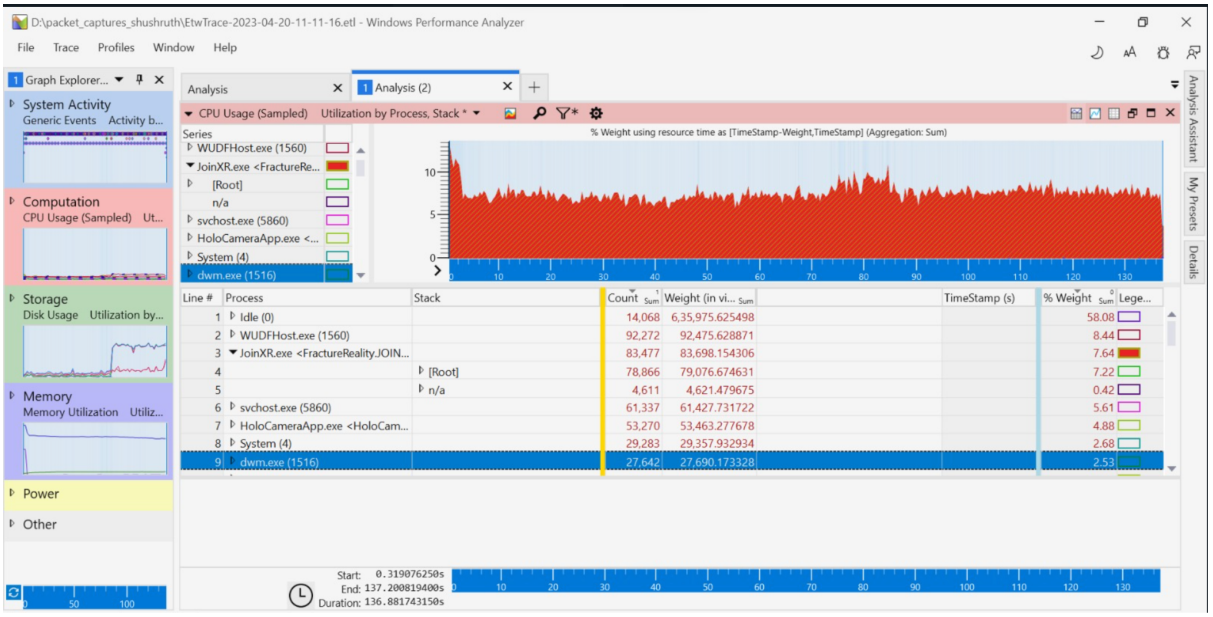


Step 2: Open the trace file, which we have captured in the Windows device portal. Usually the trace file is a huge file.





Step 3: Here we are analysing the CPU utilisation. Under the Computation tab in WPA, you can see the CPU utilisation graph for a specific application. You can select the JoinXR application and the CPU utilisation graph will be displayed for that application. Under the Processes section, you can see the percentage of the CPU utilisation for the JoinXR application. By analysing the CPU utilisation graph, you can identify if the JoinXR application is utilising the CPU efficiently or if there are any performance issues that need to be addressed. This information can be helpful in troubleshooting and optimising the performance of the application.



In addition to CPU utilisation, it's also important to monitor the memory and GPU utilisation of the Hololens tool to ensure optimal performance. To analyse memory utilisation in WPA, you can go to the Memory tab and select the process for which you want to view the memory usage. This will show you a detailed breakdown of the memory usage for the selected process, including the working set, private bytes, and virtual bytes.

To analyse GPU utilisation, you can use tools like Microsoft's Windows Performance Toolkit (WPT) or NVIDIA's Nsight Graphics to capture and analyse GPU performance metrics. These tools allow you to view GPU usage, frame rates, and other performance metrics to identify any issues or bottlenecks that may be impacting the performance of the Hololens tool.

Reference:

https://learn.microsoft.com/en-us/windows-hardware/test/wpt/windows-performance-analyzer