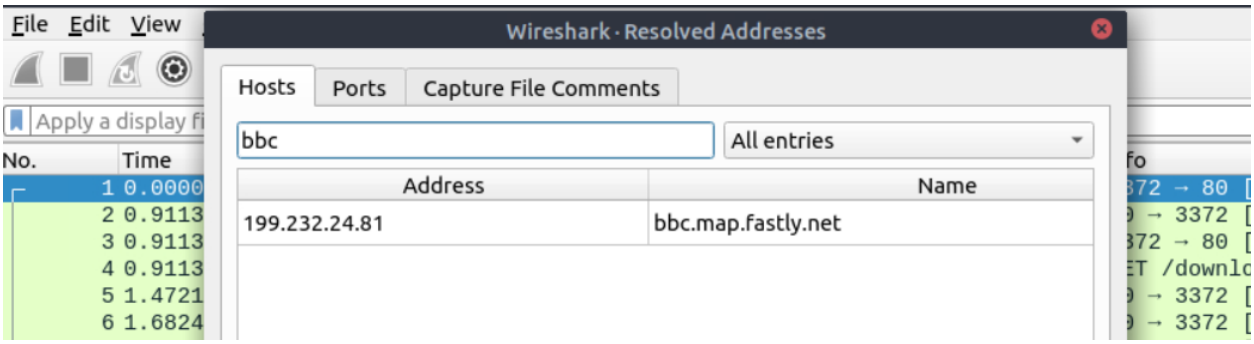
Wireshark: Packet Operations 2026/01/15

Statistics | Summary

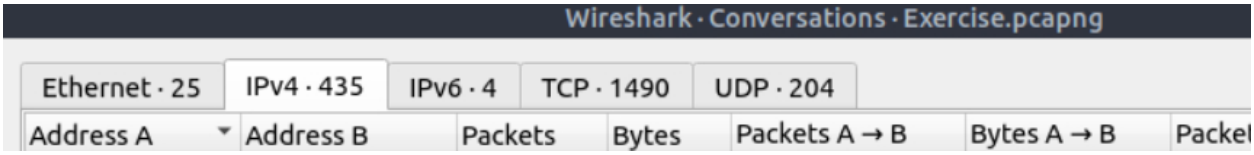
In this room, we will cover the fundamentals of packet analysis with Wireshark and investigate the event of interest at the packet-level.

Step 1: Investigate the resolved addresses. What is the IP address of the hostname starts with "bbc"?



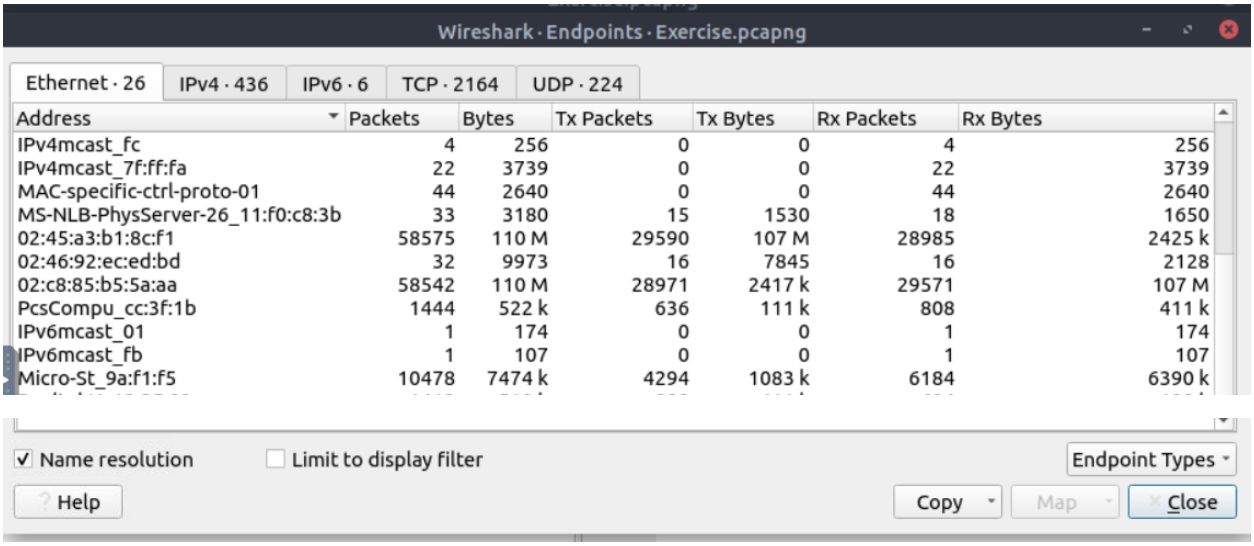
I started up the exercise and opened the pcap file through Wireshark. I clicked on 'statistics' > 'Resolved Addresses' then searched bbc. The IP address found - **199.232.24.81**.

Step 2: What is the number of IPv4 conversations?



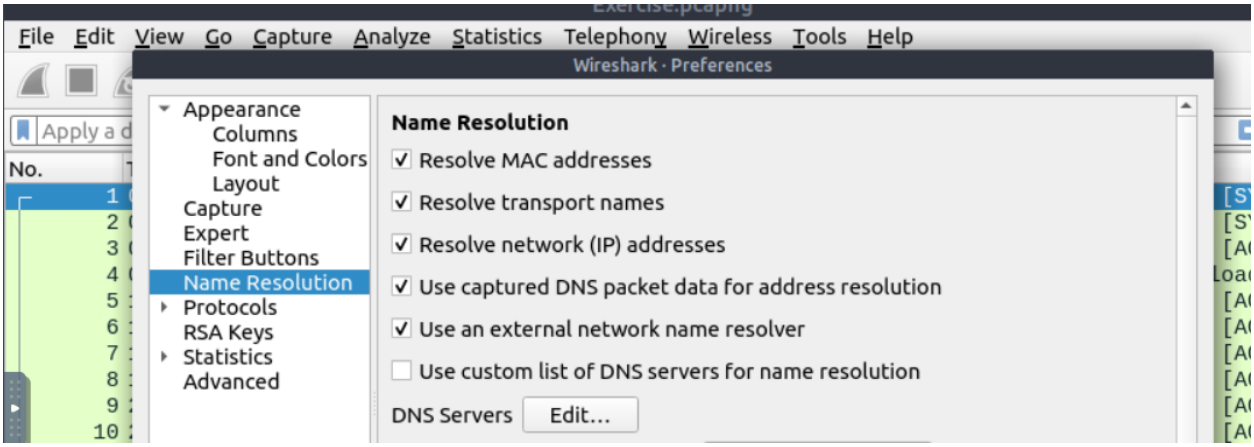
I clicked 'Statistics' > 'Conversations' then viewed the IPv4 address conversations – **435**.

Step 3: How many bytes (k) were transferred from the "Micro-St" MAC address?

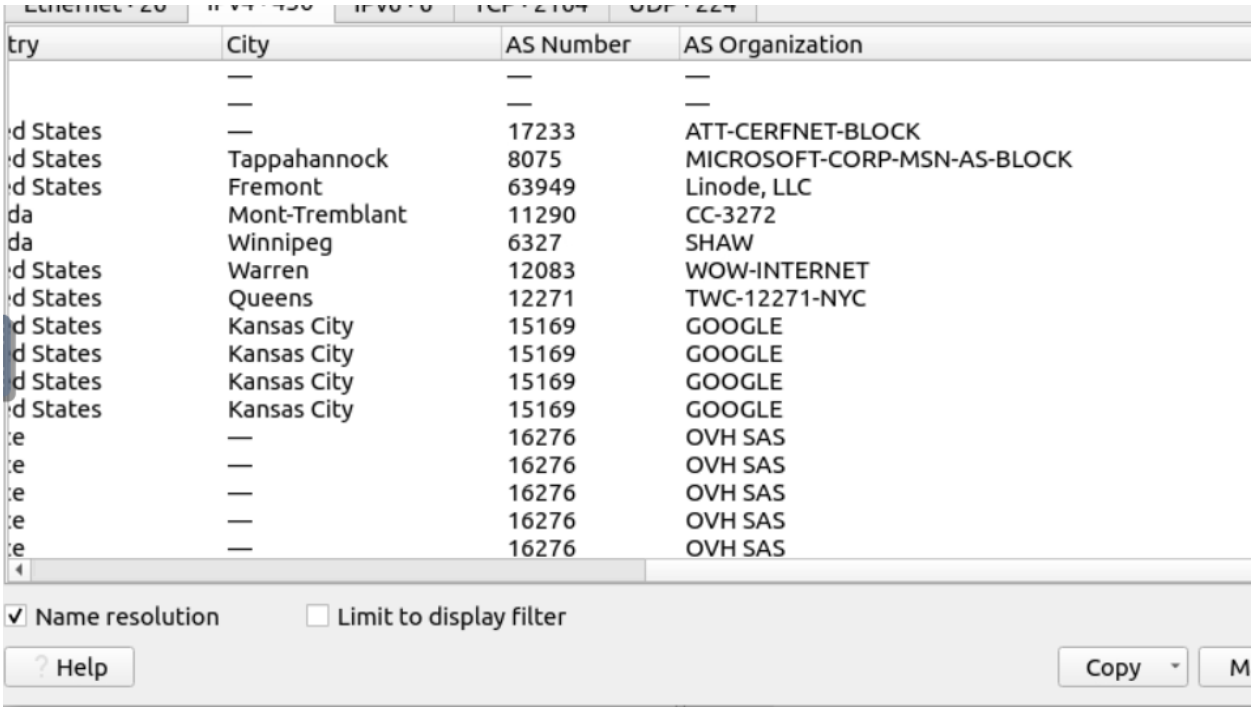


I clicked ‘Statistics’ > ‘Endpoints’ then clicked check on Name resolution at the bottom and found the number of bytes transferred – **7474 K**.

Step 4: What is the number of IP addresses linked with "Kansas City"?

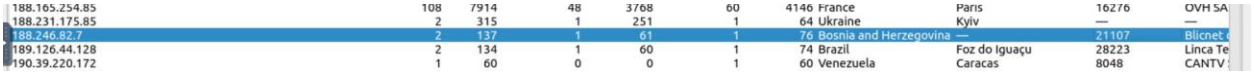


I started by clicking Edit > Preferences > Name Resolution and checked the boxes for **Resolve transport names** and **Resolve network (IP) addresses**.



I then clicked statistics > Endpoints and checked the Name resolution box and found **4** entries for Kansas City.

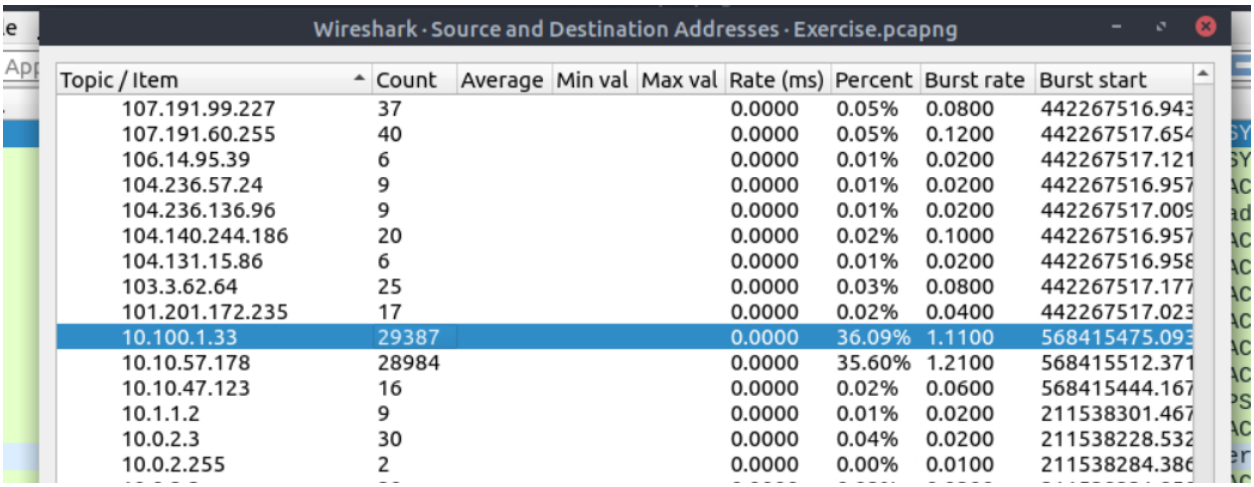
Step 5: Which IP address is linked with "Blicnet" AS Organisation?



I unchecked the Name resolution box and found the IP address after scrolling down - **188.246.82.7**.

Statistics | Protocol Details

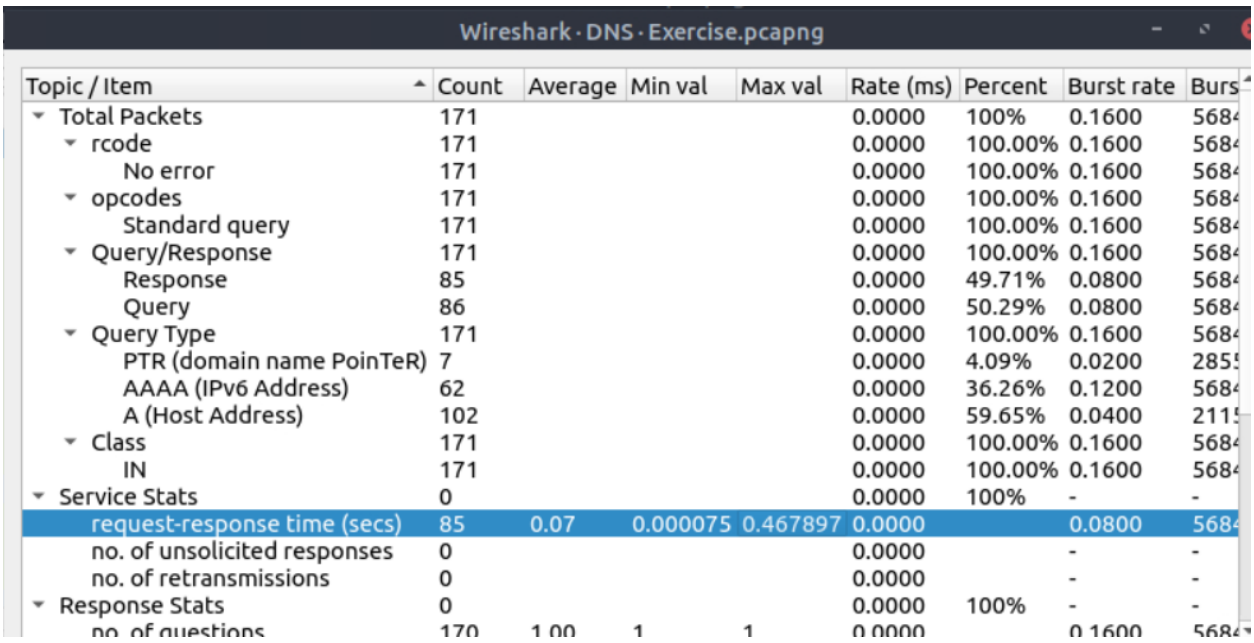
Step 1: What is the most used IPv4 destination address?



Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burst start
107.191.99.227	37				0.0000	0.05%	0.0800	442267516.943
107.191.60.255	40				0.0000	0.05%	0.1200	442267517.654
106.14.95.39	6				0.0000	0.01%	0.0200	442267517.121
104.236.57.24	9				0.0000	0.01%	0.0200	442267516.957
104.236.136.96	9				0.0000	0.01%	0.0200	442267517.009
104.140.244.186	20				0.0000	0.02%	0.1000	442267516.957
104.131.15.86	6				0.0000	0.01%	0.0200	442267516.958
103.3.62.64	25				0.0000	0.03%	0.0800	442267517.177
101.201.172.235	17				0.0000	0.02%	0.0400	442267517.023
10.100.1.33	29387				0.0000	36.09%	1.1100	568415475.093
10.10.57.178	28984				0.0000	35.60%	1.2100	568415512.371
10.10.47.123	16				0.0000	0.02%	0.0600	568415444.167
10.1.1.2	9				0.0000	0.01%	0.0200	211538301.467
10.0.2.3	30				0.0000	0.04%	0.0200	211538228.532
10.0.2.255	2				0.0000	0.00%	0.0100	211538284.386

I clicked Statistics > IPV4 Statistics > Source and Destination Address. I focused only on the destination addresses and closed the source addresses. I scrolled down investigating and managed to identify the IP address – **10.100.1.33**.

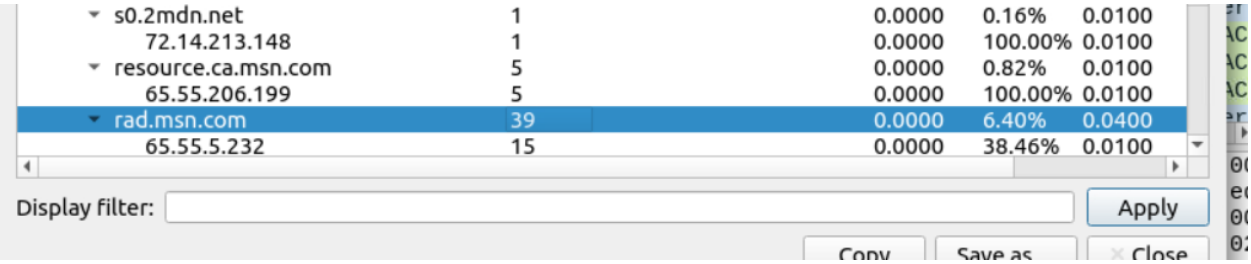
Step 2: What is the max service request-response time of the DNS packets?



Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burs
Total Packets	171				0.0000	100%	0.1600	568
rcode	171				0.0000	100.00%	0.1600	568
No error	171				0.0000	100.00%	0.1600	568
opcodes	171				0.0000	100.00%	0.1600	568
Standard query	171				0.0000	100.00%	0.1600	568
Query/Response	171				0.0000	100.00%	0.1600	568
Response	85				0.0000	49.71%	0.0800	568
Query	86				0.0000	50.29%	0.0800	568
Query Type	171				0.0000	100.00%	0.1600	568
PTR (domain name PoinTeR)	7				0.0000	4.09%	0.0200	285
AAAA (IPv6 Address)	62				0.0000	36.26%	0.1200	568
A (Host Address)	102				0.0000	59.65%	0.0400	211
Class	171				0.0000	100.00%	0.1600	568
IN	171				0.0000	100.00%	0.1600	568
Service Stats	0				0.0000	100%	-	-
request-response time (secs)	85	0.07	0.000075	0.467897	0.0000		0.0800	568
no. of unsolicited responses	0				0.0000		-	-
no. of retransmissions	0				0.0000		-	-
Response Stats	0				0.0000	100%	-	-
no. of questions	170	1.00	1	1	0.0000		0.1600	568

I clicked Statistics > DNS and went down to Service Stats > request-response time (secs) and identified the Max val - **0.467897**.

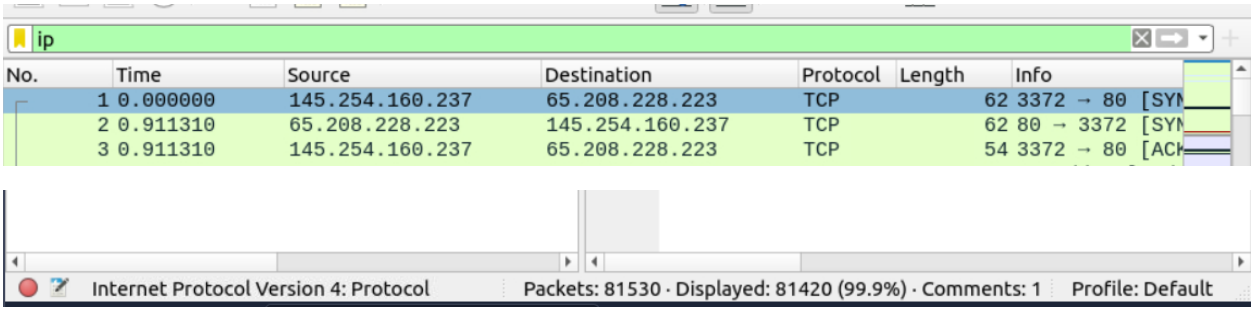
Step 3: What is the number of HTTP Requests accomplished by "rad[.]msn[.]com?



I clicked Statistics > HTTP > Load distribution and scrolled down to the domain. No. of requests accomplished – **39**.

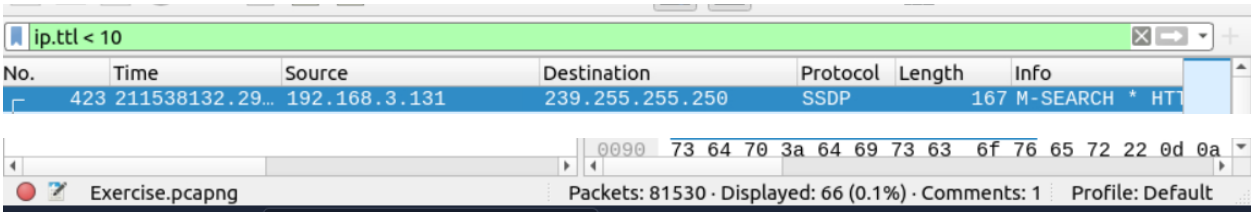
Packet Filtering | Protocol Filters

Step 1: What is the number of IP packets?



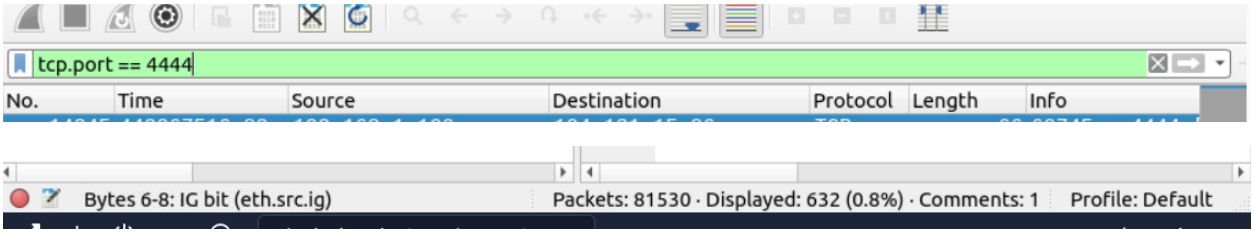
I entered **IP** in the display filter and clicked search and looked the bottom where I was able to find the no. of displayed IP packets – **81420**.

Step 2: What is the number of packets with a "TTL value less than 10"?



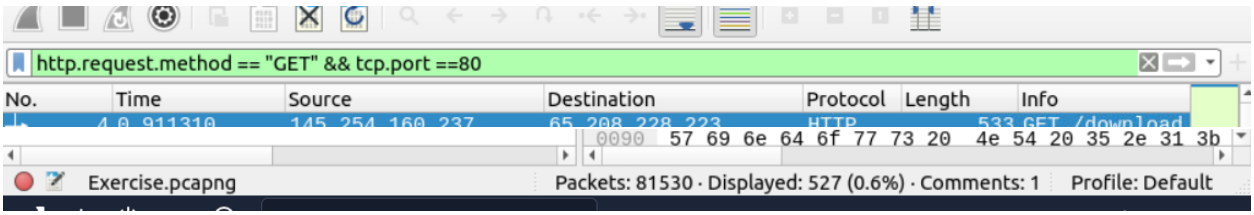
I entered the search **ip.ttl < 10** in the display filter and got number of packets – **66**.

Step 3: What is the number of packets which uses "TCP port 4444"?



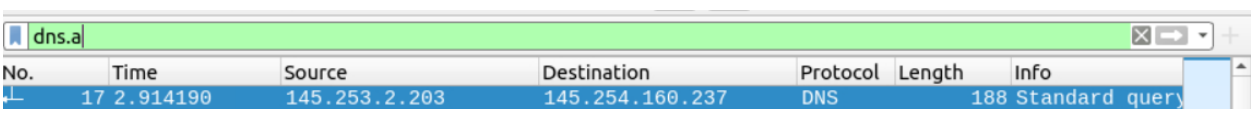
I applied the following filter **tcp.port == 4444** and searched. I looked at the bottom of the page and identified the number of packets displayed - **632.**

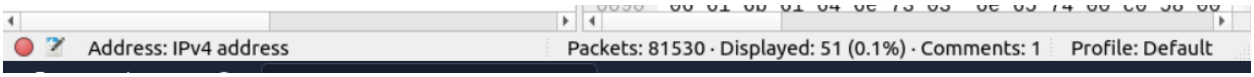
Step 4: What is the number of "HTTP GET" requests sent to port "80"?



I applied the filter **http.request.method == "GET" && tcp.port == 80** and hit search. The number of requests – **527**.

Step 5: What is the number of type A DNS Queries?

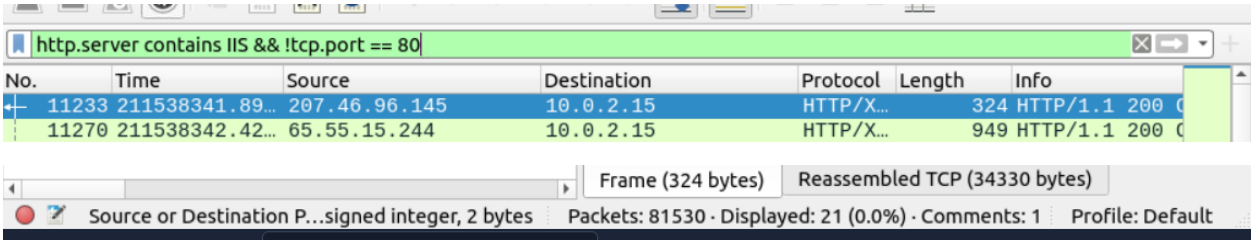




I applied the filter **dns.a** and got the number of queries – **51**.

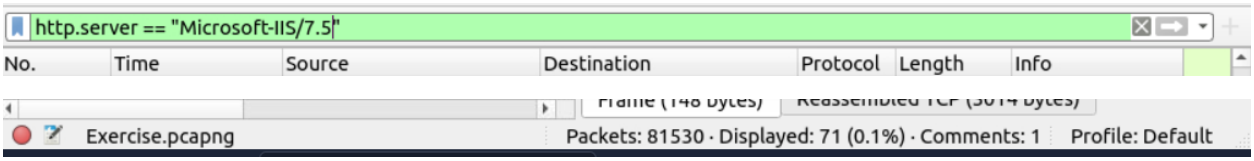
Advanced Filtering

Step 1: Find all Microsoft IIS servers. What is the number of packets that did not originate from "port 80"?



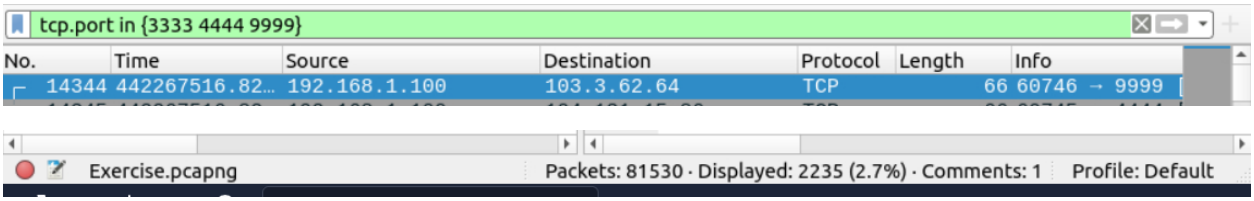
I applied the filter - **http.server contains IIS && !tcp.port == 80** and got the number of packets – **21**.

Step 2: Find all Microsoft IIS servers. What is the number of packets that have "version 7.5"?



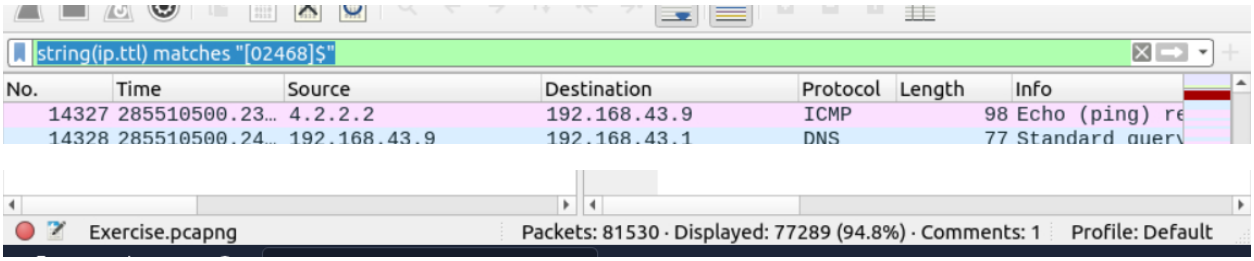
I applied the filter **http.server == "Microsoft-IIS/7.5"** and clicked search. The number of packets – **71**.

Step 3: What is the total number of packets that use ports 3333, 4444 or 9999?



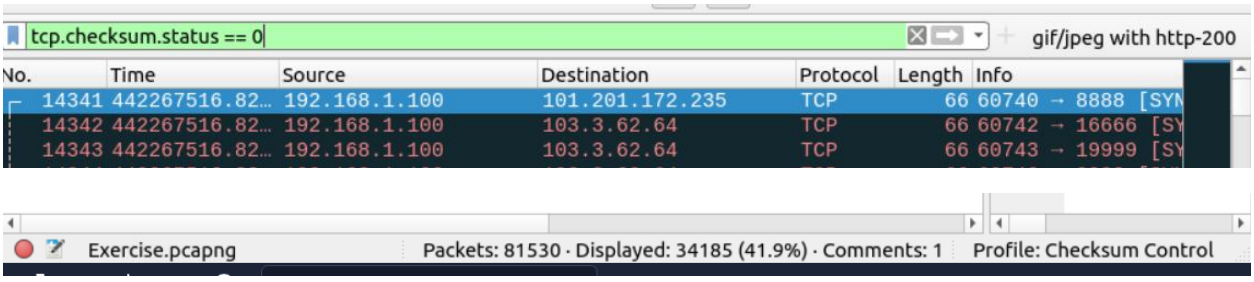
I applied the filter **tcp.port in {3333 4444 9999}** and clicked search. The number of packets identified – **2235**.

Step 4: What is the number of packets with "even TTL numbers"?



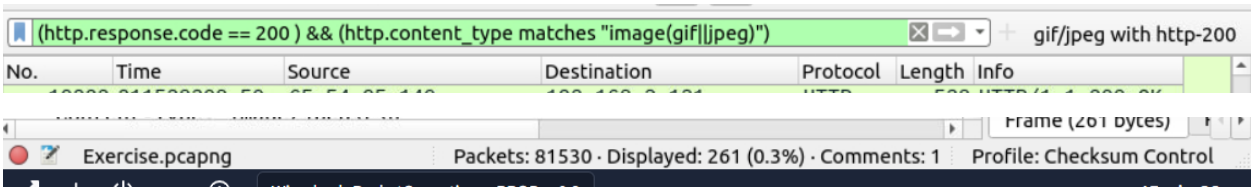
I applied the filter **string(ip.ttl) matches "[02468]$"** and clicked search. The number of packets – **77289**.

Step 5: Change the profile to "Checksum Control". What is the number of "Bad TCP Checksum" packets?



I applied the filter **tcp.checksum.status == 0**. I then went down and clicked on Profile at the bottom and changed it to ‘checksum control’ and clicked search – **34185**.

Step 6: Use the existing filtering button to filter the traffic. What is the number of displayed packets?



I cleared the display filter. I then clicked the existing button on the right. The number of packets – **261**.

Conclusion Summary

This exercise strengthened my practical understanding of packet-level network analysis using Wireshark by exploring traffic patterns through statistics, protocol analysis, and advanced filtering techniques. By analyzing resolved addresses, conversations, endpoints, and protocol usage, I was able to identify key network characteristics such as frequently contacted IP addresses, geographic associations, autonomous system ownership, and data transfer volumes.

Applying protocol-specific statistics and display filters allowed me to isolate and quantify DNS, HTTP, TCP, and IP traffic, reinforcing how analysts can efficiently narrow large packet captures to events of interest. The advanced filtering tasks further enhanced my ability to detect anomalies, investigate unusual ports, analyze TTL values, identify server technologies, and validate packet integrity using checksum controls.

Overall, this activity reinforced the importance of structured packet analysis, effective filtering, and statistical interpretation in network investigations, providing hands-on skills directly applicable to SOC monitoring, threat detection, and incident analysis workflows.