

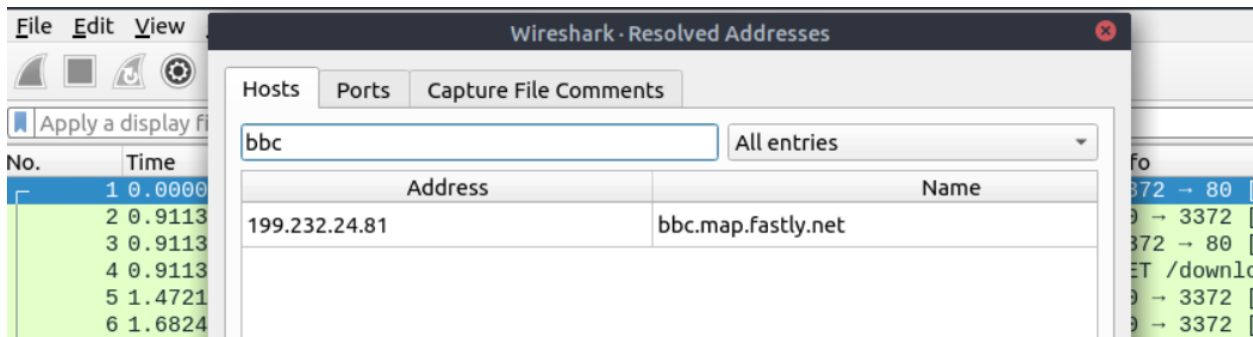
Wireshark: Packet Operations

2026/01/15

Statistics | Summary

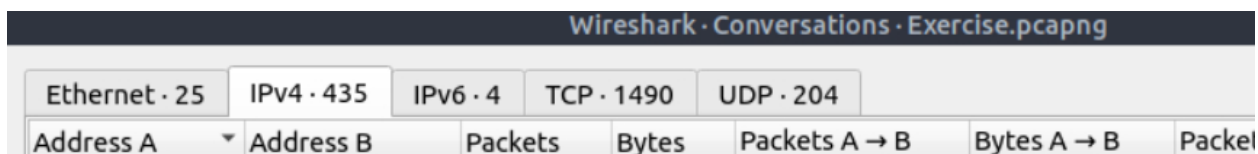
In this room, we will cover the fundamentals of packet analysis with Wireshark and investigate the event of interest at the packet-level.

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



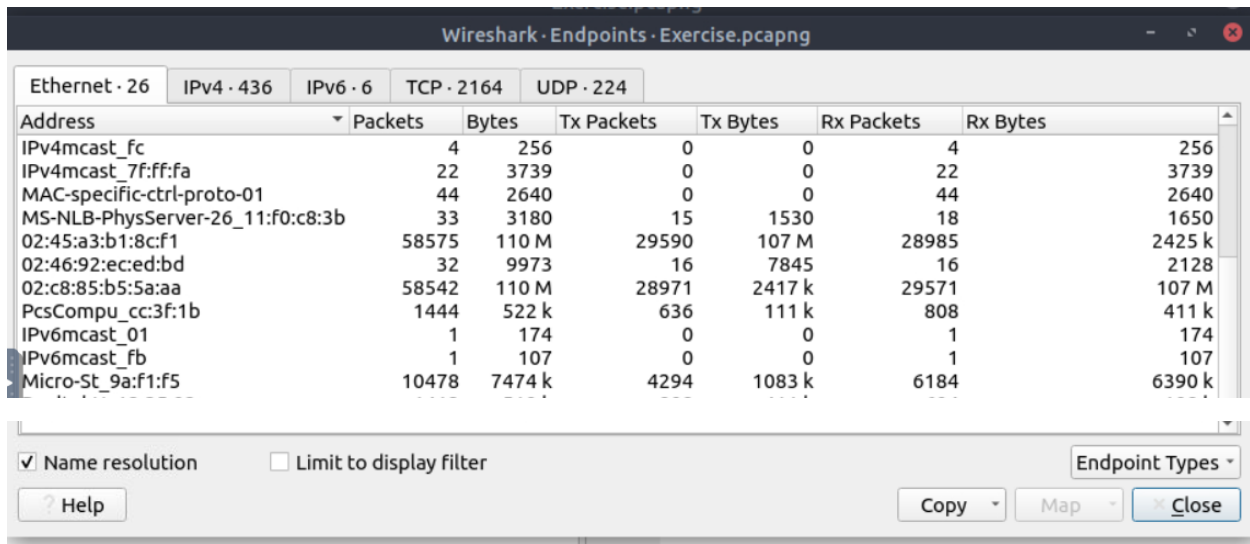
I started up the exercise and opened the pcap file through Wireshark. I clicked on 'statistics' > 'Resolved Addresses' then searched bbc. The IP address found - **199.232.24.81**.

Step 2: What is the number of IPv4 conversations?



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

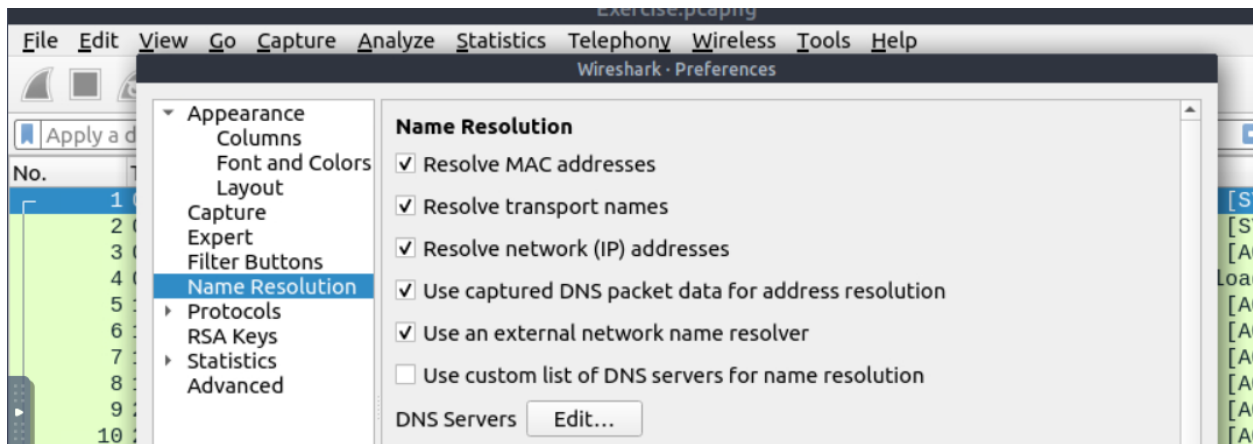
Step 3: How many bytes (k) were transferred from the "Micro-St" MAC address?



Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
IPv4mcast_fc	4	256	0	0	4	256
IPv4mcast_7f:ff:fa	22	3739	0	0	22	3739
MAC-specific-ctrl-proto-01	44	2640	0	0	44	2640
MS-NLB-PhysServer-26_11:f0:c8:3b	33	3180	15	1530	18	1650
02:45:a3:b1:8c:f1	58575	110 M	29590	107 M	28985	2425 k
02:46:92:ec:ed:bd	32	9973	16	7845	16	2128
02:c8:85:b5:5a:aa	58542	110 M	28971	2417 k	29571	107 M
PcsCompu_cc:3f:1b	1444	522 k	636	111 k	808	411 k
IPv6mcast_01	1	174	0	0	1	174
IPv6mcast_fb	1	107	0	0	1	107
Micro-St_9a:f1:f5	10478	7474 k	4294	1083 k	6184	6390 k

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**.

Country	City	AS Number	AS Organization
United States	—	—	—
United States	—	—	—
United States	Tappahannock	17233	ATT-CERFNET-BLOCK
United States	Fremont	8075	MICROSOFT-CORP-MSN-AS-BLOCK
Canada	Mont-Tremblant	63949	Linode, LLC
Canada	Winnipeg	11290	CC-3272
United States	Warren	6327	SHAW
United States	Queens	12083	WOW-INTERNET
United States	Queens	12271	TWC-12271-NYC
United States	Kansas City	15169	GOOGLE
United States	Kansas City	15169	GOOGLE
United States	Kansas City	15169	GOOGLE
United States	Kansas City	15169	GOOGLE
France	—	16276	OVH SAS
France	—	16276	OVH SAS
France	—	16276	OVH SAS
France	—	16276	OVH SAS
France	—	16276	OVH SAS

4

☒ Name resolution ☐ Limit to display filter

[? Help](#) [Copy](#) [M](#)

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

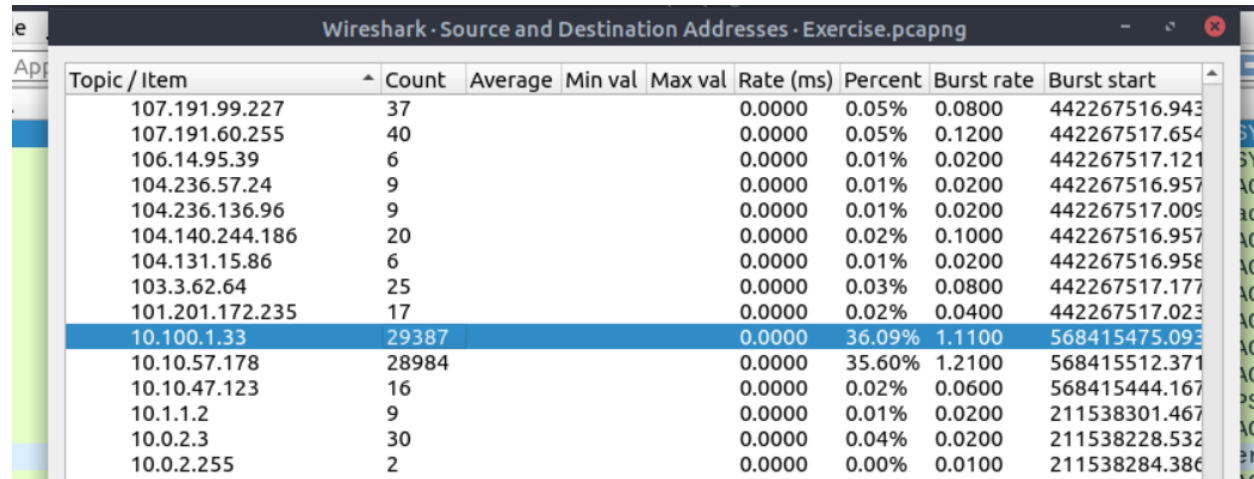
Step 5: Which IP address is linked with "Blicnet" AS Organisation?

188.165.254.85	108	7914	48	3768	60	4146 France	Paris	16276	OVH SAS
188.231.175.85	2	315	1	251	1	64 Ukraine	Kyiv	—	—
188.246.82.7	2	137	1	61	1	76 Bosnia and Herzegovina	—	21107	Blicnet
189.126.44.128	2	134	1	60	1	74 Brazil	Foz do Iguaçu	28223	Linca Te
190.39.220.172	1	60	0	0	1	60 Venezuela	Caracas	8048	CANTV

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?

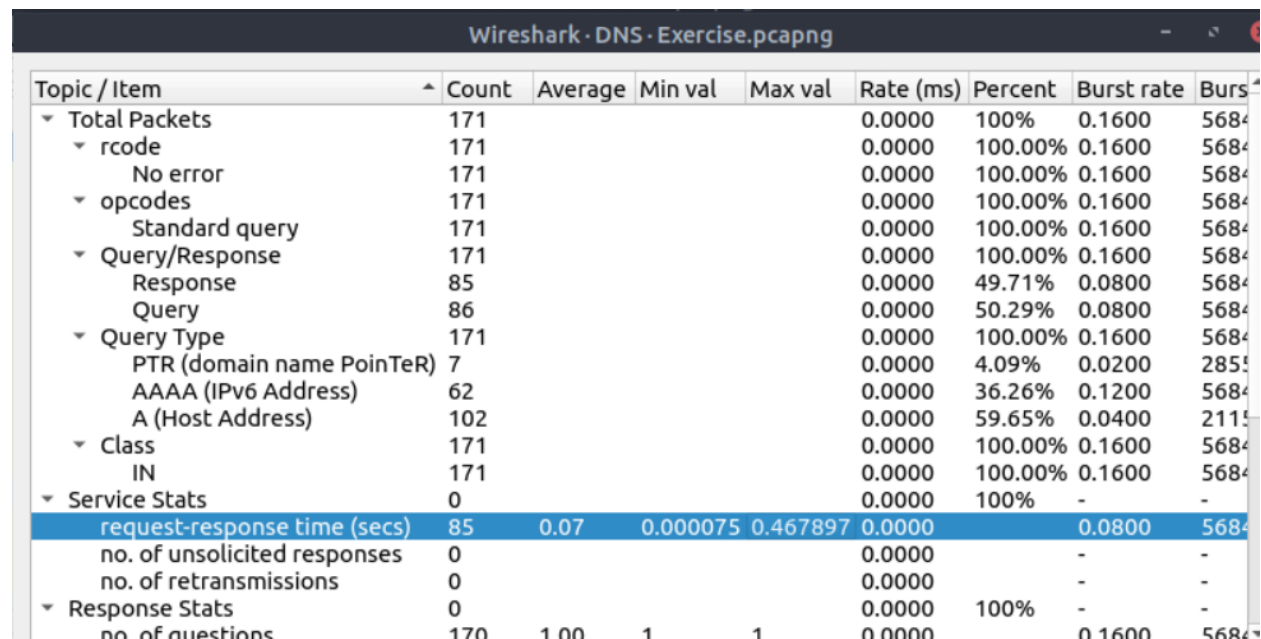


Wireshark · Source and Destination Addresses · Exercise.pcapng

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?

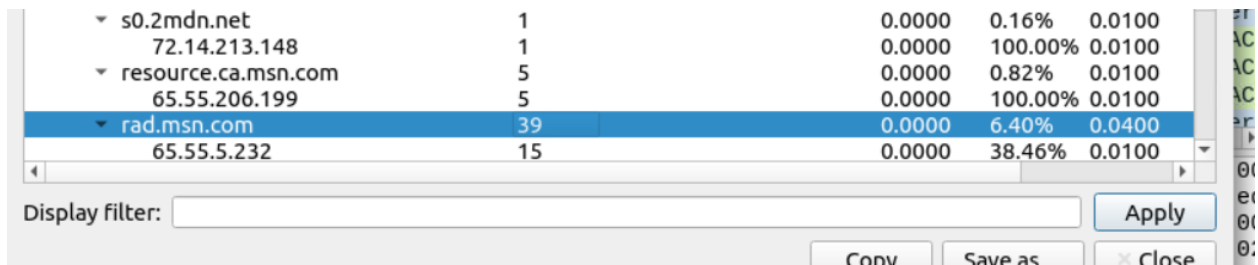


Wireshark · DNS · Exercise.pcapng

Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burst start
▼ Total Packets	171				0.0000	100%	0.1600	5684
▼ rcode	171				0.0000	100.00%	0.1600	5684
No error	171				0.0000	100.00%	0.1600	5684
▼ opcodes	171				0.0000	100.00%	0.1600	5684
Standard query	171				0.0000	100.00%	0.1600	5684
▼ Query/Response	171				0.0000	100.00%	0.1600	5684
Response	85				0.0000	49.71%	0.0800	5684
Query	86				0.0000	50.29%	0.0800	5684
▼ Query Type	171				0.0000	100.00%	0.1600	5684
PTR (domain name PoinTeR)	7				0.0000	4.09%	0.0200	2855
AAAA (IPv6 Address)	62				0.0000	36.26%	0.1200	5684
A (Host Address)	102				0.0000	59.65%	0.0400	2115
▼ Class	171				0.0000	100.00%	0.1600	5684
IN	171				0.0000	100.00%	0.1600	5684
▼ Service Stats	0				0.0000	100%	-	-
request-response time (secs)	85	0.07	0.000075	0.467897	0.0000		0.0800	5684
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	5684

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"?



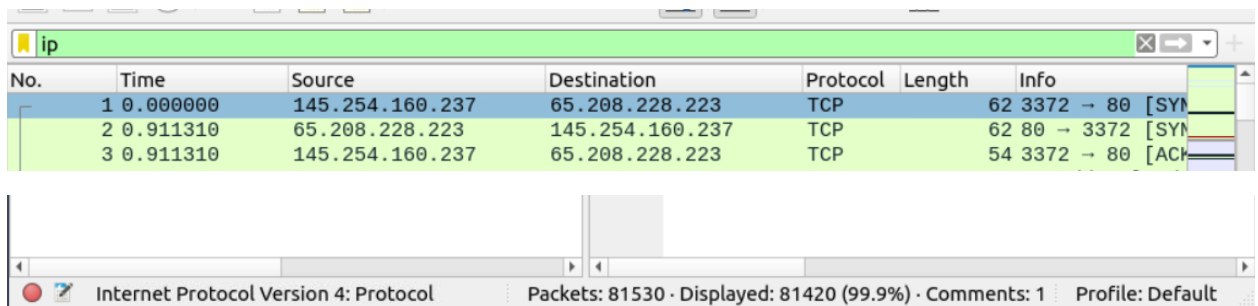
A screenshot of the Wireshark Statistics pane, specifically the HTTP Load distribution section. The table lists various domains and their corresponding request counts and percentages. The row for 'rad.msn.com' is highlighted in blue, showing 39 requests and 6.40% of the total. Below the table is a 'Display filter' input field and buttons for 'Apply', 'Conv', 'Save as', and 'Close'.

Domain	Count	Percentage
s0.2mdn.net	1	0.0000 0.16%
72.14.213.148	1	0.0000 100.00%
resource.ca.msn.com	5	0.0000 0.82%
65.55.206.199	5	0.0000 100.00%
rad.msn.com	39	0.0000 6.40%
65.55.5.232	15	0.0000 38.46%

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?

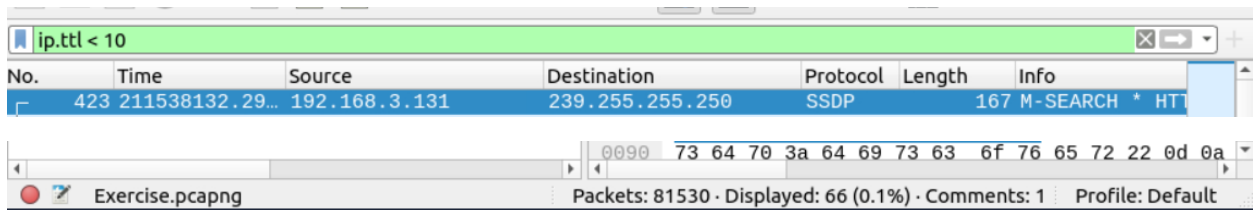


A screenshot of the Wireshark packet list pane. The display filter 'ip' is entered in the top bar. The table shows three packets, all of which are TCP. The bottom status bar indicates that 81420 out of 81530 packets are displayed (99.9%).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	145.254.160.237	65.208.228.223	TCP	62	3372 → 80 [SYN]
2	0.911310	65.208.228.223	145.254.160.237	TCP	62	80 → 3372 [SYN]
3	0.911310	145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK]

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"?

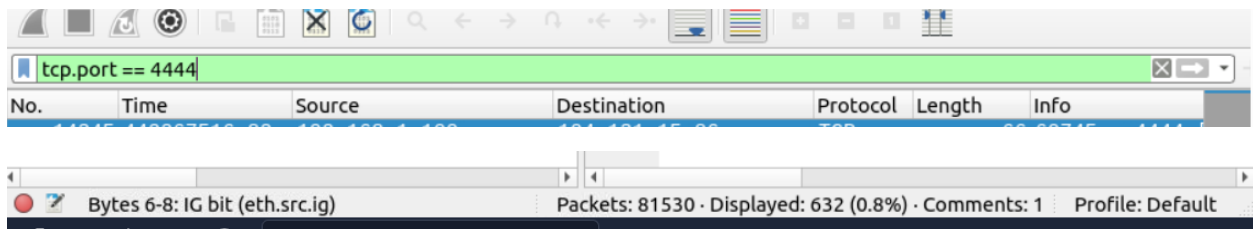


No.	Time	Source	Destination	Protocol	Length	Info
423	2.11538132	192.168.3.131	239.255.255.250	SSDP	167	M-SEARCH * HT

Exercise.pcapng Packets: 81530 · Displayed: 66 (0.1%) · Comments: 1 Profile: Default

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"?

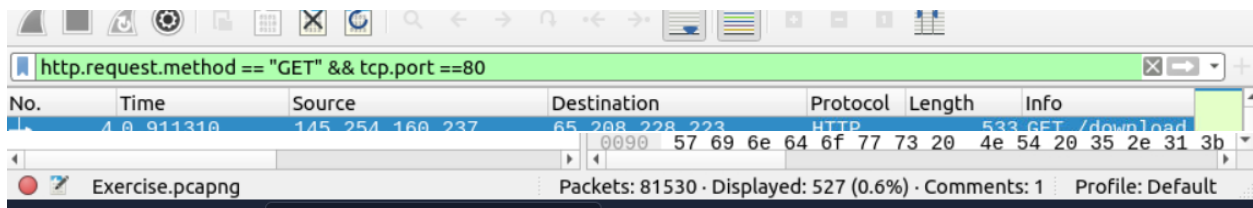


No.	Time	Source	Destination	Protocol	Length	Info
11845	4.18287512	192.168.1.131	192.168.1.131	TCP	60	80 → 4444 [RST] Seq=14141

Bytes 6-8: IG bit (eth.src.ig) Packets: 81530 · Displayed: 632 (0.8%) · Comments: 1 Profile: Default

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"?

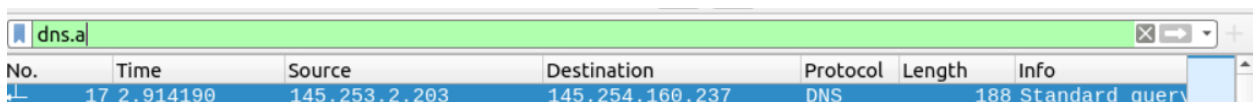


No.	Time	Source	Destination	Protocol	Length	Info
4	0.911310	145.254.160.237	65.208.228.223	HTTP	533	GET /download

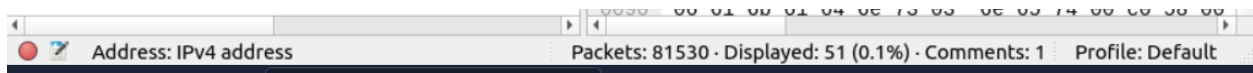
Exercise.pcapng Packets: 81530 · Displayed: 527 (0.6%) · Comments: 1 Profile: Default

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?



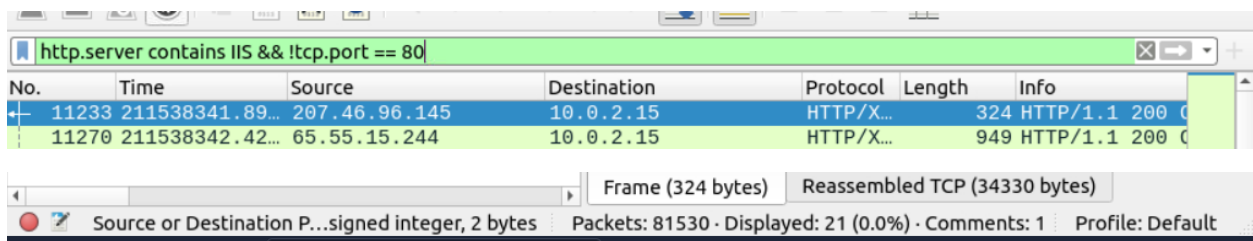
No.	Time	Source	Destination	Protocol	Length	Info
17	2.914190	145.253.2.203	145.254.160.237	DNS	188	Standard query



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

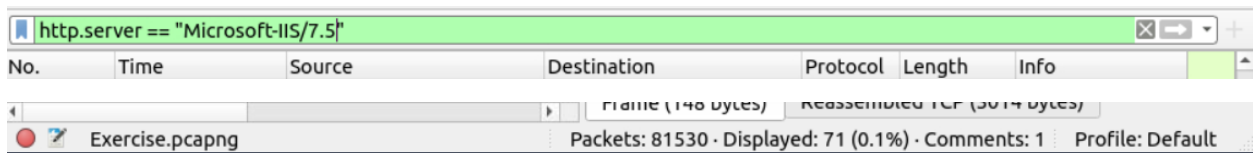
Advanced Filtering

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



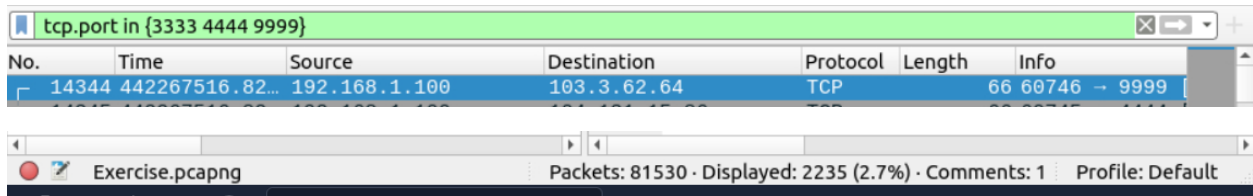
I applied the filter - **http.server contains IIS && !tcp.port == 80** and got the number of packets – **21**.

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?

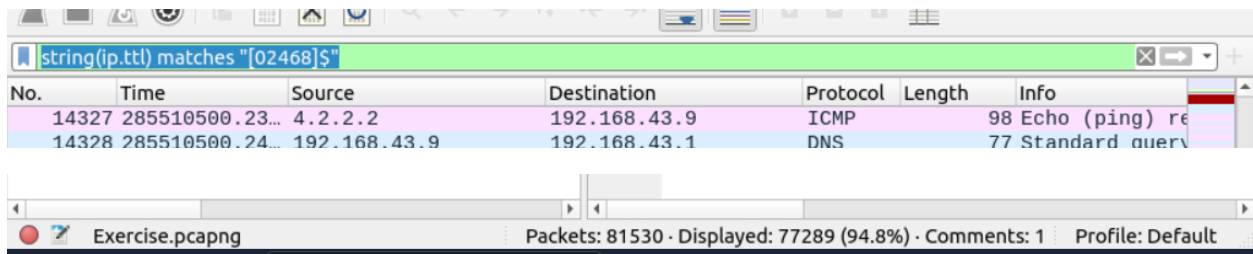


No.	Time	Source	Destination	Protocol	Length	Info
14344	442267516.82...	192.168.1.100	103.3.62.64	TCP	66	60746 -> 9999

Exercise.pcapng Packets: 81530 - Displayed: 2235 (2.7%) - Comments: 1 Profile: Default

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"?

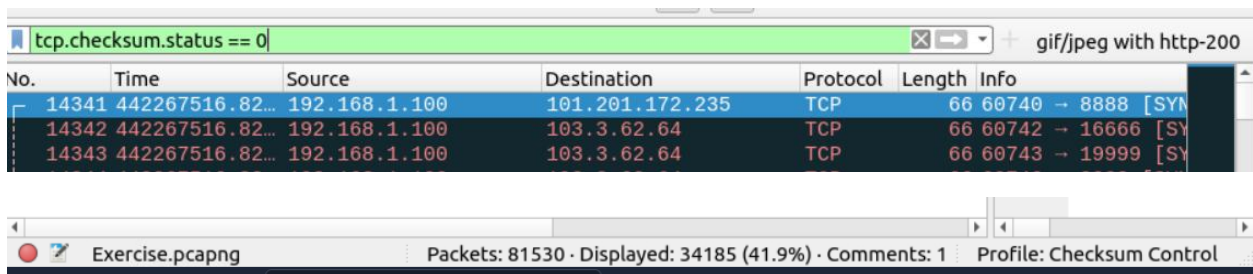


No.	Time	Source	Destination	Protocol	Length	Info
14327	285510500.23...	4.2.2.2	192.168.43.9	ICMP	98	Echo (ping) request
14328	285510500.24...	192.168.43.9	192.168.43.1	DNS	77	Standard query

Exercise.pcapng Packets: 81530 - Displayed: 77289 (94.8%) - Comments: 1 Profile: Default

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?

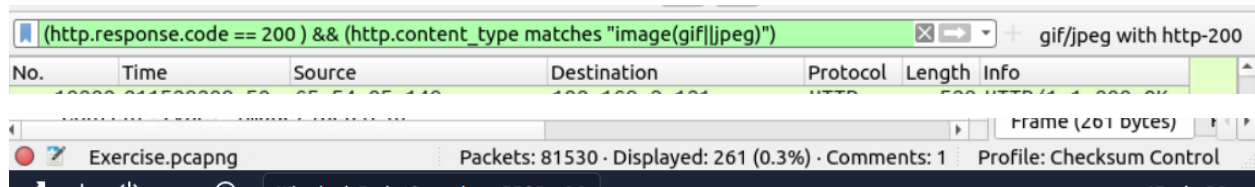


No.	Time	Source	Destination	Protocol	Length	Info
14341	442267516.82...	192.168.1.100	101.201.172.235	TCP	66	60740 -> 8888 [SYN]
14342	442267516.82...	192.168.1.100	103.3.62.64	TCP	66	60742 -> 16666 [SYN]
14343	442267516.82...	192.168.1.100	103.3.62.64	TCP	66	60743 -> 19999 [SYN]

Exercise.pcapng Packets: 81530 - Displayed: 34185 (41.9%) - Comments: 1 Profile: Checksum Control

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.