

CSL6010 : Cybersecurity

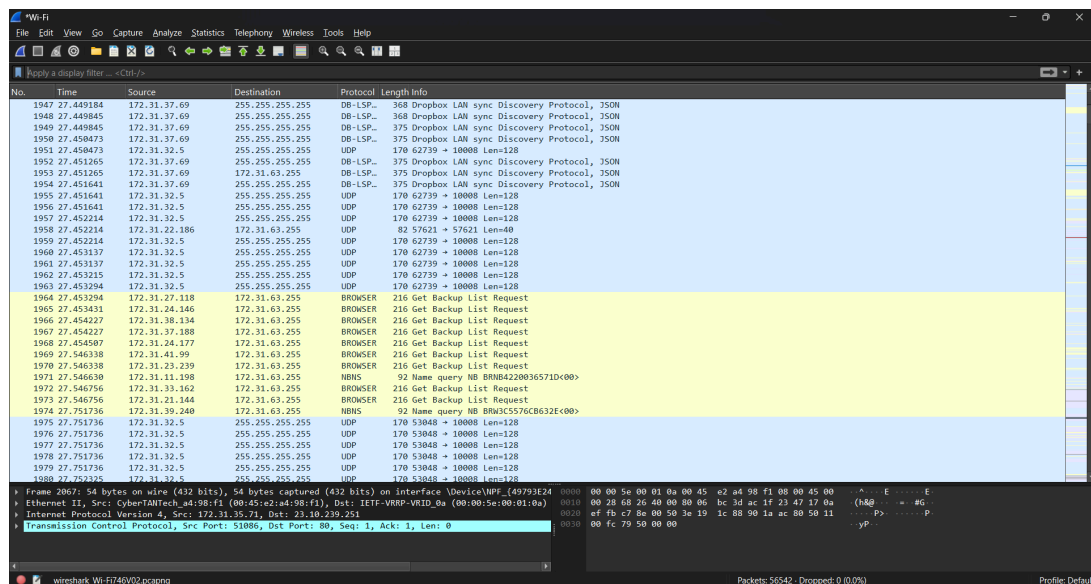
Assignment - III

Name : Abhay Kashyap

Roll no : B22CS001

1. Start packet capture in Wireshark on your wireless interface. What do you observe?

This is the traffic overview captured by Wireshark, displaying detailed information such as source and destination addresses, along with the protocols in use. Wireshark serves as a comprehensive tool for analyzing network traffic and diagnosing potential issues in the communication flow.



2. Now visit a local website, say www.iitj.ac.in. Subsequently stop the packet capture and record your observations. Are you able to see the DNS request? What about TCP and HTTP? What is the IP address of the IITJ server? Are you able to see different HTTP requests/responses? Please justify your answer with relevant screenshots.

When opening the iitj.ac.in website, we can see the network requesting the IP address of the IITJ server and receiving it in response

Time	Source	Destination	Protocol	Length	Info
41	136.159932394	192.168.164.128	DNS	75	Standard query 0xf6b8 A www.youtube.com
42	136.159999286	192.168.164.128	DNS	75	Standard query 0xf6b6 AAAA www.youtube.com
43	136.160857425	192.168.164.128	DNS	70	Standard query 0xf02c A iitj.ac.in
44	136.161211441	192.168.164.128	DNS	365	Standard query response 0xf6b8 A www.youtube.com CNAME youtub...
45	136.161211610	192.168.164.128	DNS	221	Standard query response 0xf6b6 AAAA www.youtube.com CNAME you...
46	136.161227718	192.168.164.128	DNS	70	Standard query 0xf23 A iitj.ac.in
47	136.161263374	192.168.164.128	DNS	70	Standard query 0xe322 AAAA iitj.ac.in
48	136.161607716	192.168.164.128	DNS	75	Standard query 0xa102 A aide.iitj.ac.in
49	136.161937678	192.168.164.128	DNS	86	Standard query response 0xf02c A iitj.ac.in A 172.16.100.5
50	136.162467012	192.168.164.128	DNS	86	Standard query response 0xf23 A iitj.ac.in A 172.16.100.5
51	136.162467156	192.168.164.128	DNS	118	Standard query response 0xe322 AAAA iitj.ac.in SOA ns-sec.iit...
52	136.162467184	192.168.164.128	DNS	91	Standard query response 0xa102 A aide.iitj.ac.in A 172.16.100...

The following screenshot shows the TCP handshake using three packets in sequence :-

SYN (Client to Server)

SYN-ACK (Server to Client)

ACK (Client to Server)

54	136.172013278	192.168.164.128	172.16.100.5	TCP	74	48354 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PER...
55	136.188223137	172.16.100.5	192.168.164.128	TCP	60	443 → 48354 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=14...
56	136.188257247	192.168.164.128	172.16.100.5	TCP	54	48354 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
57	136.189194782	192.168.164.128	172.16.100.5	TLSv1.2	635	Client Hello (SNI=iitj.ac.in)
58	136.189370851	172.16.100.5	192.168.164.128	TCP	60	443 → 48354 [ACK] Seq=1 Ack=582 Win=64240 Len=0
59	136.191052455	172.16.100.5	192.168.164.128	TLSv1.2	191	Server Hello, Change cipher Spec, Encrypted Handshake Me...

Here is the HTTP request and response after a successful handshake

263	136.343842128	192.168.164.128	172.16.100.5	HTTP	415	GET /uploaded_docs/new3.gif HTTP/1.1
265	136.345854561	172.16.100.5	192.168.164.128	HTTP	529	HTTP/1.1 302 Found (text/html)
315	136.413202238	192.168.164.128	172.16.100.5	HTTP	422	GET /uploaded_docs/new_latest4.png HTTP/1.1
321	136.415159346	172.16.100.5	192.168.164.128	HTTP	542	HTTP/1.1 302 Found (text/html)
326	136.415445207	192.168.164.128	172.16.100.5	HTTP	413	GET /uploaded_docs/fb.png HTTP/1.1
330	136.417100517	172.16.100.5	192.168.164.128	HTTP	524	HTTP/1.1 302 Found (text/html)
332	136.429201009	192.168.164.128	172.16.100.5	HTTP	416	GET /uploaded_docs/tweet.png HTTP/1.1
334	136.429673413	192.168.164.128	172.16.100.5	HTTP	417	GET /uploaded_docs/youtub.png HTTP/1.1
338	136.430798990	172.16.100.5	192.168.164.128	HTTP	530	HTTP/1.1 302 Found (text/html)
339	136.431038654	192.168.164.128	172.16.100.5	HTTP	416	GET /uploaded_docs/insta.png HTTP/1.1
342	136.431233629	172.16.100.5	192.168.164.128	HTTP	533	HTTP/1.1 302 Found (text/html)
346	136.431423983	192.168.164.128	172.16.100.5	HTTP	436	GET /uploaded_docs/G20%20logo_theme_00062023.jpg HTTP/1.1
347	136.431521598	192.168.164.128	172.16.100.5	HTTP	440	GET /uploaded_docs/logo_azadikamahotsav_02092021.jpg HTTP/1.1
354	136.432370853	172.16.100.5	192.168.164.128	HTTP	530	HTTP/1.1 302 Found (text/html)
355	136.432943500	172.16.100.5	192.168.164.128	HTTP	570	HTTP/1.1 302 Found (text/html)
356	136.432943568	172.16.100.5	192.168.164.128	HTTP	579	HTTP/1.1 302 Found (text/html)

3. What does a packet highlighted in 'black' color signify?

Black packets represent some sort of error in the transmission of the packet. Black packets in network analysis tools (such as Wireshark) indicate TCP retransmissions, duplicate ACKs, or lost segments. These events often signal potential network issues like packet loss or congestion, which can degrade network performance.

1. TCP Retransmissions

- TCP ensures reliable data transfer by using sequence numbers and acknowledgments (ACKs).
- When a sender transmits a packet, it expects an acknowledgment from the receiver.
- If the sender does not receive an ACK within a specific time (due to packet loss or excessive delay), it retransmits the same packet.
- This retransmission is detected in network analysis as a black packet, indicating that the original packet may have been lost or delayed.

2. Duplicate Acknowledgments (Dup ACKs)

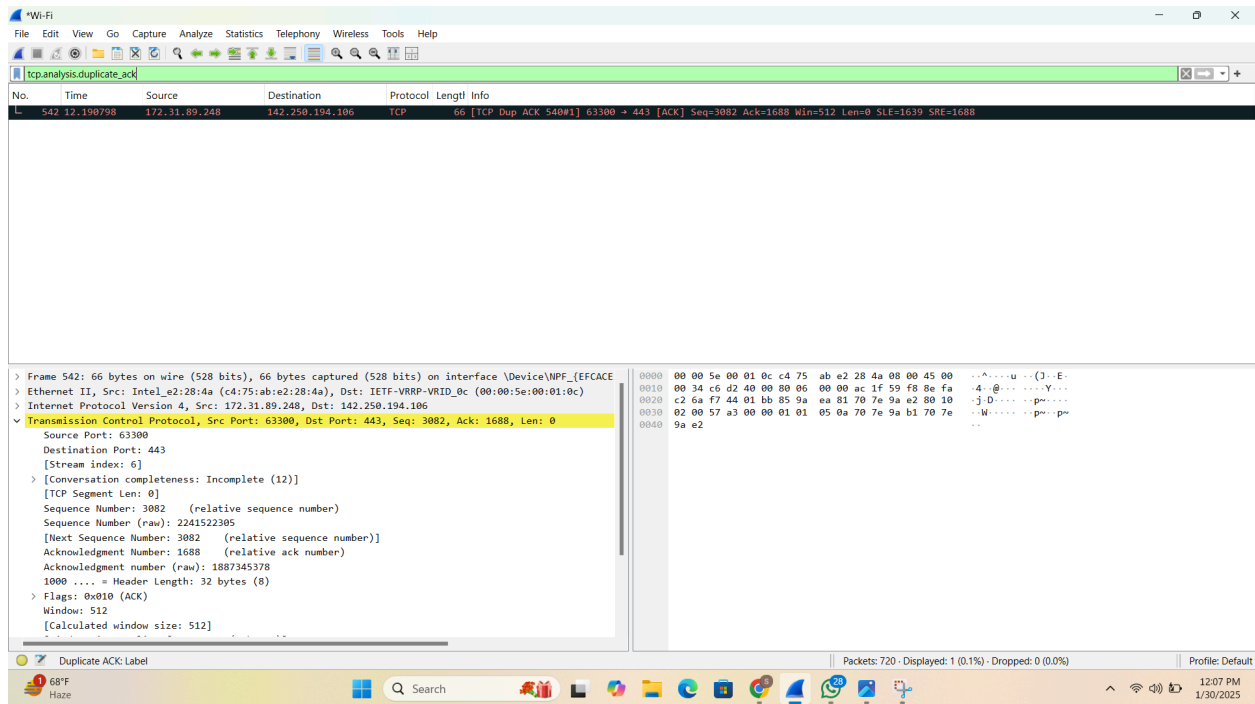
- When a receiver gets an out-of-order segment (because a previous packet was lost), it sends an acknowledgment for the last successfully received packet.
- If the sender receives three consecutive duplicate ACKs, it assumes the missing packet is lost and performs fast retransmission.
- A high number of duplicate ACKs suggests packet loss or network instability.

3. Lost Segments

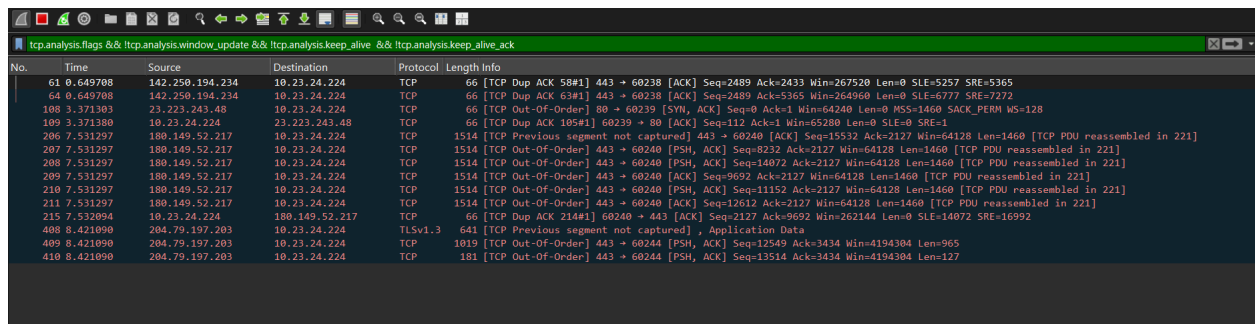
- If a packet is lost in transit (due to congestion or network failure), the sender does not receive an acknowledgment.
- This results in either retransmissions or duplicate ACKs.
- Persistent lost segments can cause slow performance, increased latency, and reduced throughput.

Apply a display filter ... <Ctrl-/>					
No.	Time	Source	Destination	Protocol	Length Info
530	11.899186	142.250.194.106	172.31.89.248	TCP	54 443 → 63300 [ACK] Seq=1572 Ack=1487 Win=785 Len=0
531	11.899186	142.250.194.106	172.31.89.248	TCP	54 443 → 63300 [ACK] Seq=1572 Ack=2899 Win=780 Len=0
532	11.899186	142.250.194.106	172.31.89.248	TCP	54 443 → 63300 [ACK] Seq=1572 Ack=3082 Win=780 Len=0
533	11.906081	142.250.192.170	172.31.89.248	TCP	54 443 → 63512 [ACK] Seq=179 Ack=311 Win=1039 Len=0
534	11.992360	172.31.23.108	172.31.63.255	UDP	82 55205 → 1947 Len=40
535	11.992360	172.31.23.108	255.255.255.255	UDP	82 55204 → 1947 Len=40
536	12.084244	172.31.28.35	255.255.255.255	DB-LSP..	237 Dropbox LAN sync Discovery Protocol, JSON
537	12.084244	172.31.28.35	172.31.63.255	DB-LSP..	237 Dropbox LAN sync Discovery Protocol, JSON
538	12.188112	142.250.194.106	172.31.89.248	TLSv1.2	121 Application Data
539	12.188112	142.250.194.106	172.31.89.248	TLSv1.2	103 Application Data
540	12.188222	172.31.89.248	142.250.194.106	TCP	54 63300 → 443 [ACK] Seq=3082 Ack=1688 Win=512 Len=0
541	12.190761	142.250.194.106	172.31.89.248	TCP	103 [TCP Spurious Retransmission] 443 → 63300 [PSH, ACK] Seq=1639 Ack=3082 Win=780 Len=49
542	12.190796	172.31.89.248	142.250.194.106	TCP	69 [TCP Dup ACK 540#1] 63300 → 443 [ACK] Seq=3082 Ack=1688 Win=512 Len=0 SRTT=1639 SRE=1688
543	12.288999	172.31.26.77	255.255.255.255	UDP	125 10004 → 10004 Len=83
544	12.288999	172.31.26.77	255.255.255.255	UDP	70 57444 → 22222 Len=28
545	12.288999	172.31.26.77	255.255.255.255	UDP	70 57443 → 22222 Len=28
546	12.288999	172.31.26.77	172.31.63.255	UDP	70 57442 → 22222 Len=28
547	12.288999	172.31.26.77	172.31.63.255	UDP	70 57445 → 22222 Len=28

Duplicate Acknowledgment



`tcp.analysis.flags && !tcp.analysis.window_update && !tcp.analysis.keep_alive`
`&& !tcp.analysis.keep_alive_ac` : Wireshark filter shows TCP packets that are not related to window updates or keep-alive messages. It filters out packets with the *window_update* or *keep_alive* flags, focusing on other types of TCP traffic.



4. Explore at least 5 different filters in Wireshark (<https://wiki.wireshark.org/DisplayFilters>). Ex. “http” would give you only HTTP traffic.

dns :

No.	dns dissever	Source	Destination	Protocol	Length	Info
119	172.16.100.205	172.31.35.71	172.16.100.205	DNS	75	Standard query 0x636f A windows.msn.com
119	172.16.100.205	172.31.35.71	172.31.35.71	DNS	218	Standard query response 0x636f A windows.msn.com CNAME www-msn-com-a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.2...
1293	172.16.100.205	172.31.35.71	172.16.100.205	DNS	79	Standard query 0x8fbb A ev2-ring.msedge.net
1312	172.16.100.3	172.31.35.71	172.16.100.3	DNS	79	Standard query 0x8fbb A ev2-ring.msedge.net
1314	172.16.100.205	172.31.35.71	172.31.35.71	DNS	304	Standard query response 0x8fbb A ev2-ring.msedge.net CNAME Ev2-ring.Ev2-9999.Ev2-msedge.net CNAME Ev2-9999.Ev2-msedge.net A 150...
1360	172.16.100.3	172.31.35.71	172.31.35.71	DNS	327	Standard query response 0x8fbb A ev2-ring.msedge.net CNAME Ev2-ring.Ev2-9999.Ev2-msedge.net CNAME Ev2-9999.Ev2-dc-msedge.net A 4.1...
1363	172.31.35.71	172.16.100.205	172.16.100.205	DNS	81	Standard query 0x2507 A teams-ring.msedge.net
1366	172.16.100.205	172.31.35.71	172.31.35.71	DNS	228	Standard query response 0x2507 A teams-ring.msedge.net CNAME teams-ring.teams-9999.teams-msedge.net CNAME teams-9999.teams-msed...
1412	172.16.100.205	172.31.35.71	172.16.100.205	DNS	79	Standard query 0xe8bd A arc-ring.msedge.net
1417	172.31.35.71	172.16.100.3	172.16.100.3	DNS	79	Standard query 0xe8bd A arc-ring.msedge.net
1434	172.16.100.205	172.31.35.71	172.31.35.71	DNS	304	Standard query response 0xe8bd A arc-ring.msedge.net CNAME arc-ring.arc-9999.arc-msedge.net CNAME arc-9999.arc-msedge.net A 172...
1493	172.16.100.3	172.31.35.71	172.31.35.71	DNS	304	Standard query response 0xe8bd A arc-ring.msedge.net CNAME arc-ring.arc-9999.arc-msedge.net CNAME arc-9999.arc-msedge.net A 172...
2830	172.31.35.71	172.16.100.205	172.16.100.205	DNS	79	Standard query 0x4cb5 A arm-ring.msedge.net
2838	172.31.35.71	172.16.100.3	172.16.100.3	DNS	79	Standard query 0x4cb5 A arm-ring.msedge.net
2855	172.16.100.205	172.31.35.71	172.31.35.71	DNS	220	Standard query response 0x4cb5 A arm-ring.msedge.net CNAME arm-ring.arm-9999.arm-msedge.net CNAME arm-9999.arm-msedge.net A 4.1...
2871	172.16.100.3	172.31.35.71	172.31.35.71	DNS	220	Standard query response 0x4cb5 A arm-ring.msedge.net CNAME arm-ring.arm-9999.arm-msedge.net CNAME arm-9999.arm-msedge.net A 4.1...
2913	172.31.35.71	172.16.100.205	172.16.100.205	DNS	113	Standard query 0x9f0f A f767217333a703830665d3fba0f74772.azr.footprintdns.com
2914	172.31.35.71	172.16.100.3	172.16.100.3	DNS	113	Standard query 0x9f0f A f767217333a703830665d3fba0f74772.azr.footprintdns.com
2938	172.31.35.71	172.16.100.205	172.16.100.205	DNS	88	Standard query 0xc145 A static.edge.microsoftapp.net
2939	172.31.35.71	172.16.100.205	172.16.100.205	DNS	88	Standard query 0x8398 HTTPS static.edge.microsoftapp.net
2940	172.16.100.205	172.31.35.71	172.31.35.71	DNS	360	Standard query response 0x8398 HTTPS static.edge.microsoftapp.net CNAME edge-cloud-resource-static.azureedge.net CNAME edge-clo...
2941	172.16.100.3	172.31.35.71	172.31.35.71	DNS	508	Standard query response 0x9f0f A f767217333a703830665d3fba0f74772.azr.footprintdns.com CNAME azperfmptargets-prod.trafficmanag...
2958	172.16.100.205	172.31.35.71	172.31.35.71	DNS	384	Standard query response 0xc145 A static.edge.microsoftapp.net CNAME edge-cloud-resource-static.azureedge.net CNAME edge-cloud-r...
2974	172.16.100.205	172.31.35.71	172.31.35.71	DNS	542	Standard query response 0x9f0f A f767217333a703830665d3fba0f74772.azr.footprintdns.com CNAME azperfmptargets-prod.trafficmanag...
3086	172.31.35.71	172.16.100.205	172.16.100.205	DNS	74	Standard query 0xc752 A login.live.com
3087	172.16.100.205	172.31.35.71	172.31.35.71	DNS	510	Standard query response 0xc752 A login.live.com CNAME login.msa.msidentity.com CNAME www.tn.lg.prod.aadms.trafficmanager.net C...
3112	172.31.35.71	172.16.100.205	172.16.100.205	DNS	78	Standard query 0x4910 A edge.microsoft.com
3113	172.31.35.71	172.16.100.205	172.16.100.205	DNS	78	Standard query 0x7df1 HTTPS edge.microsoft.com
3114	172.31.35.71	172.16.100.205	172.16.100.205	DNS	80	Standard query 0xc9b7 A substrate.office.com
3115	172.31.35.71	172.16.100.205	172.16.100.205	DNS	80	Standard query 0x1a4b HTTPS substrate.office.com
3116	172.16.100.205	172.31.35.71	172.31.35.71	DNS	249	Standard query response 0x9140 A edge.microsoft.com CNAME edge-microsoft.com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedg...
3117	172.16.100.205	172.31.35.71	172.31.35.71	DNS	422	Standard query response 0xc9b7 A substrate.office.com CNAME outlook.office365.com CNAME ooc-g2.tm4.office.com CNAME outlook.ms...
3118	172.16.100.205	172.31.35.71	172.31.35.71	DNS	259	Standard query response 0x1a4b HTTPS substrate.office.com CNAME outlook.office365.com CNAME ooc-g2.tm4.office.com CNAME outlook.ms...
3119	172.16.100.205	172.31.35.71	172.31.35.71	DNS	106	Standard query response 0x7df1 HTTPS edge.microsoft.com CNAME edge-microsoft.com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-m...
<pre> Frame 1493: 304 bytes on wire (2432 bits), 304 bytes captured (2432 bits) on interface \Device\NPF... Ethernet II, Src: Cisco B9:03:62 (08:e5:9e:b9:03:62), Dst: CyberIANTech_a4:98:f1 (00:45:e2:a4:98:f1) Internet Protocol Version 4, Src: 172.16.100.3, Dst: 172.31.35.71 User Datagram Protocol, Src Port: 53, Dst Port: 56433 Domain Name System (response) 0000 00 45 e2 a4 98 f1 68 e5 9e b9 03 62 08 00 45 00 E...h...b...E 0010 01 22 0d 5a 00 00 3e 11 f0 6e ac 10 6d 03 ac 1f Z...d... 0020 23 47 08 5a 3d 71 01 0e 04 3c e0 8d 81 80 00 01 #G 5 q...X... 0030 00 04 00 04 00 04 08 61 72 63 2d 72 69 6e 67 06 ...a-r-c-r-i-n-g 0040 6d 73 65 64 67 65 03 6e 65 74 00 00 01 00 01 c0 msedge.net 0050 0c 00 05 00 01 00 00 00 3c 00 1f 08 61 72 63 2d <...arc- 0060 72 69 6e 67 08 61 72 63 2d 39 39 39 0a 61 72 ring arc-9999 ar 0070 63 2d 6d 73 65 64 67 65 c0 1c 31 00 05 00 01 c-msedge 1... </pre>						

tcp :

No.	Time	Source	Destination	Protocol	Length	Info
2067	29.084381	23.10.239.251	23.10.239.251	TCP	54	S1086 → 80 [FIN, ACK] Seq=1 Ack=1 Win=252 Len=0
2074	29.098067	23.10.239.251	172.31.35.71	TCP	54	80 → S1086 [FIN, ACK] Seq=1 Ack=2 Win=501 Len=0
2075	29.098164	172.31.35.71	23.10.239.251	TCP	54	S1086 → 80 [ACK] Seq=2 Ack=2 Win=252 Len=0
20603	104.049580	172.31.35.71	151.101.38.172	TCP	66	51239 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM WS=512
20608	104.254186	151.101.38.172	172.31.35.71	TCP	66	80 → 51239 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM WS=512
20609	104.254402	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
2070	104.255146	172.31.35.71	151.101.38.172	HTTP	409	HEAD /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
2091	104.452149	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=1 Ack=356 Win=147456 Len=0
2092	104.452149	151.101.38.172	172.31.35.71	HTTP	648	HTTP/1.1 200 OK
2096	104.492356	172.31.35.71	151.101.38.172	HTTP	481	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
2151	104.678537	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=595 Ack=783 Win=148480 Len=1456 [TCP PDU reassembled in 22161]
2152	104.678537	151.101.38.172	172.31.35.71	TCP	1510	80 → 51239 [ACK] Seq=595 Ack=783 Win=148480 Len=1456 [TCP PDU reassembled in 22161]
2153	104.731371	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=783 Ack=2051 Win=65280 Len=0
2161	104.833464	151.101.38.172	172.31.35.71	HTTP	359	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
2163	104.885980	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=783 Ack=2356 Win=65024 Len=0
22576	110.641040	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
22589	110.864026	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=2356 Ack=1213 Win=149504 Len=0
22590	110.864026	151.101.38.172	172.31.35.71	HTTP	716	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
22598	110.910593	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=1213 Ack=3018 Win=64512 Len=0
23128	119.126427	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
23131	119.303072	151.101.38.172	172.31.35.71	HTTP	922	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
23133	119.357340	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=1643 Ack=3886 Win=65280 Len=0
23549	124.183326	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
23561	124.365644	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=3886 Ack=2073 Win=151552 Len=0
23562	124.365644	151.101.38.172	172.31.35.71	HTTP	712	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
23566	124.414522	172.31.35.71	151.101.38.172	TCP	54	51239 → 80 [ACK] Seq=2073 Ack=4544 Win=64768 Len=0
24159	132.272491	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
24164	132.445715	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=4544 Ack=2593 Win=152576 Len=0
24165	132.445715	151.101.38.172	172.31.35.71	HTTP	917	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
24166	132.488956	172.31.35.71	151.101.38.172	HTTP	54	51239 → 80 [ACK] Seq=2593 Ack=5407 Win=65280 Len=0
24968	140.350175	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/zed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PHdUwJxg2bPngQd1ObRk2FDFO...
24970	140.532136	151.101.38.172	172.31.35.71	TCP	54	80 → 51239 [ACK] Seq=5407 Ack=2933 Win=153600 Len=0
24971	140.532475	151.101.38.172	172.31.35.71	HTTP	943	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
<pre> Frame 2067: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF... Ethernet II, Src: CyberIANTech_a4:98:f1 (00:45:e2:a4:98:f1), Dst: IETF-VRRP-VRID_0a (00:00:5e:00:01:0a) Internet Protocol Version 4, Src: 172.31.35.71, Dst: 23.10.239.251 Transmission Control Protocol, Src Port: 51086, Dst Port: 80, Seq: 1, Ack: 1, Len: 0 0000 00 00 5e 00 01 0a 00 45 e2 a4 98 f1 08 00 45 00 E...E...E...E 0010 00 28 68 26 40 00 00 06 bc 3d ac 1f 23 47 17 0a (h8&...P...#G... 0020 ef fb c7 8e 00 50 3e 19 1c 88 90 1a ac 80 50 11 ...P...P...P... 0030 00 fc 79 50 00 00 </pre>						

smb :

No.	Time	Source	Destination	Protocol	Length	Info
1862	26.526291	172.31.38.134	172.31.63.255	BROWSER	216	Get Backup List Request
1866	26.527474	172.31.24.177	172.31.63.255	BROWSER	216	Get Backup List Request
1896	26.529770	172.31.41.99	172.31.63.255	BROWSER	216	Get Backup List Request
1898	26.531148	172.31.23.239	172.31.63.255	BROWSER	216	Get Backup List Request
1913	27.240157	172.31.8.31	172.31.63.255	BROWSER	216	Get Backup List Request
1964	27.453294	172.31.23.64	172.31.63.255	BROWSER	216	Get Backup List Request
1965	27.453431	172.31.27.118	172.31.63.255	BROWSER	216	Get Backup List Request
1966	27.454227	172.31.24.146	172.31.63.255	BROWSER	216	Get Backup List Request
1967	27.454227	172.31.38.134	172.31.63.255	BROWSER	216	Get Backup List Request
1968	27.454507	172.31.37.188	172.31.63.255	BROWSER	216	Get Backup List Request
1969	27.454507	172.31.24.177	172.31.63.255	BROWSER	216	Get Backup List Request
1969	27.456338	172.31.41.99	172.31.63.255	BROWSER	216	Get Backup List Request
1970	27.456338	172.31.23.239	172.31.63.255	BROWSER	216	Get Backup List Request
1972	27.546756	172.31.33.162	172.31.63.255	BROWSER	216	Get Backup List Request
2079	29.185917	172.31.45.20	172.31.63.255	BROWSER	216	Get Backup List Request
2094	29.595106	172.31.45.67	172.31.63.255	BROWSER	216	Get Backup List Request
2110	29.699906	172.31.37.188	172.31.63.255	BROWSER	216	Get Backup List Request
2111	29.699906	172.31.33.162	172.31.63.255	BROWSER	216	Get Backup List Request
2112	29.708635	172.31.21.144	172.31.63.255	BROWSER	216	Get Backup List Request
2114	29.701620	172.31.27.118	172.31.63.255	BROWSER	216	Get Backup List Request
2115	29.701620	172.31.38.134	172.31.63.255	BROWSER	216	Get Backup List Request
2116	29.701856	172.31.24.146	172.31.63.255	BROWSER	216	Get Backup List Request
2117	29.701856	172.31.25.36	172.31.63.255	BROWSER	216	Get Backup List Request
2118	29.701856	172.31.24.177	172.31.63.255	BROWSER	216	Get Backup List Request
2119	29.799837	172.31.41.99	172.31.63.255	BROWSER	216	Get Backup List Request
2123	29.902504	172.31.23.239	172.31.63.255	BROWSER	216	Get Backup List Request
2163	30.620195	172.31.34.193	172.31.63.255	BROWSER	216	Get Backup List Request
2236	31.541372	172.31.25.36	172.31.63.255	BROWSER	216	Get Backup List Request
2239	31.542936	172.31.45.194	172.31.63.255	BROWSER	216	Get Backup List Request
2318	32.674509	172.31.54.142	172.31.63.255	BROWSER	216	Get Backup List Request
2319	32.268730	172.31.19.129	172.31.63.255	BROWSER	216	Get Backup List Request
Frame 1973: 216 bytes on wire (1728 bits), 216 bytes captured (1728 bits) on interface \Device\NPF_{49...}						
Ethernet II, Src: F2:38:2F:9E:9E:a9 (F2:38:2F:9E:9E:a9), Dst: Broadcast (ff:ff:ff:ff:ff:ff)						
Internet Protocol Version 4, Src: 172.31.21.144, Dst: 172.31.63.255						
User Datagram Protocol, Src Port: 138, Dst Port: 138						
NetBIOS Datagram Service						
SMB (Server Message Block Protocol)						
SMB MailSlot Protocol						

http :

No.	Time	Source	Destination	Protocol	Length	Info
22070	104.255146	172.31.35.71	151.101.38.172	HTTP	409	HEAD /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
22092	104.452149	151.101.38.172	172.31.35.71	HTTP	648	HTTP/1.1 200 OK
22096	104.492356	172.31.35.71	151.101.38.172	HTTP	481	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
22161	104.833464	151.101.38.172	172.31.35.71	HTTP	359	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
22576	110.641040	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
22590	110.864026	151.101.38.172	172.31.35.71	HTTP	716	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
23128	119.126427	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
23132	119.303072	151.101.38.172	172.31.35.71	HTTP	922	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
23549	124.183326	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
23562	124.365644	151.101.38.172	172.31.35.71	HTTP	712	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
24159	132.272491	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
24165	132.445715	151.101.38.172	172.31.35.71	HTTP	917	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
24968	140.359175	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
24971	140.532475	151.101.38.172	172.31.35.71	HTTP	943	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
25018	141.368642	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
25028	141.541713	151.101.38.172	172.31.35.71	HTTP	1142	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
25148	143.377889	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
25168	143.680999	151.101.38.172	172.31.35.71	HTTP	1321	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
25201	144.387837	172.31.35.71	151.101.38.172	HTTP	484	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
25216	144.581398	151.101.38.172	172.31.35.71	HTTP	56	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
25289	145.618348	172.31.35.71	151.101.38.172	HTTP	485	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
25372	146.808863	172.31.35.71	151.101.38.172	HTTP	486	GET /filestreamingservice/files/2ed1297e-f6c9-4355-aec4-433ea371b1167P1=17384813678P2=4048P3=28P4=03PhdUwJxgX2bPngQd10bRk2FDOFH7Zxg...
25437	147.157275	151.101.38.172	172.31.35.71	HTTP	593	HTTP/1.1 206 Partial Content (application/x-chrome-extension)
25528	148.404673	172.31.35.71	151.101.38.172	HTTP	409	HEAD /filestreamingservice/files/2a0d597c-a09c-4400-be86-87596dd2e6967P1=17384813748P2=4048P3=28P4=Tz233yuoTPFrZQt1H5jjzLYC95vXRaWvG...
25537	148.578522	151.101.38.172	172.31.35.71	HTTP	646	HTTP/1.1 200 OK
25545	148.615691	172.31.35.71	151.101.38.172	HTTP	460	GET /filestreamingservice/files/2a0d597c-a09c-4400-be86-87596dd2e6967P1=17384813748P2=4048P3=28P4=Tz233yuoTPFrZQt1H5jjzLYC95vXRaWvG...
25570	148.813223	151.101.38.172	172.31.35.71	HTTP	482	HTTP/1.1 200 OK (application/x-chrome-extension)
Frame 22070: 409 bytes on wire (3272 bits), 409 bytes captured (3272 bits) on interface \Device\NPF_{49...}						
Ethernet II, Src: CybenANTech_a4:98:f1 (00:45:e2:a4:98:f1), Dst: IETF-VRRP-VRID_0a (00:00:5e:00:01:0a)						
Internet Protocol Version 4, Src: 172.31.35.71, Dst: 151.101.38.172						
Transmission Control Protocol, Src Port: 51239, Dst Port: 80, Seq: 1, Ack: 1, Len: 355						
Hypertext Transfer Protocol						

udp :

No.	Source	Destination	Protocol	Length	Info
10	172.31.34.193	172.31.63.255	UDP	82	53033 → 1947 Len=40
11	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
5 0.102571	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
6 0.102680	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
7 0.103092	172.31.20.8	172.31.63.255	UDP	82	57621 → 57621 Len=40
8 0.103092	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
9 0.103185	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
10 0.103924	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
11 0.103924	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
12 0.103924	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
13 0.104384	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
14 0.104384	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
16 0.205299	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
17 0.205299	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
18 0.205299	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
19 0.205564	172.31.34.193	255.255.255.255	UDP	82	53034 → 1947 Len=40
20 0.205564	172.31.34.193	172.31.63.255	UDP	82	53033 → 1947 Len=40
21 0.205626	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
22 0.206357	172.31.32.5	255.255.255.255	UDP	170	61155 → 10000 Len=128
23 0.206357	172.31.30.233	172.31.63.255	UDP	305	54915 → 54915 Len=263
24 0.409782	172.31.34.193	172.31.63.255	UDP	82	53033 → 1947 Len=40
25 0.409782	172.31.34.193	255.255.255.255	UDP	82	53034 → 1947 Len=40
26 0.409782	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
27 0.410254	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
28 0.512331	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
29 0.512331	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
30 0.512438	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
31 0.512601	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
32 0.512601	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
33 0.512650	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
34 0.513290	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
35 0.513290	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
36 0.513373	172.31.32.5	255.255.255.255	UDP	170	57497 → 10000 Len=128
Frame 1: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface \Device\NPF_{49793E24-CE}					
Ethernet II, Src: 00:2e:2d:10:06:68 (00:2e:2d:10:06:68), Dst: Broadcast (ff:ff:ff:ff:ff:ff)					
Internet Protocol Version 4, Src: 172.31.34.193, Dst: 172.31.63.255					
User Datagram Protocol, Src Port: 53033, Dst Port: 1947					
Data (40 bytes)					
ff ff ff ff ff ff ff ff 2d 10 06 68 08 00 45 00 h: E					
00 44 ae 6a 00 00 00 11 d1 3f ac 1f 22 c1 ac 1f .D.j....?..					
3f ff cf 29 07 9b 00 30 dd 30 7a 7a 77 57 47 34 ?)...0 @zzwG4					
4f 6e 44 52 47 53 35 45 52 4b 77 59 71 32 41 45 OnDRGSSE RKwYq2AE					
68 6a 4a 39 74 77 4e 59 4e 6d 32 4b 32 48 61 67 hJ39twiY Nm2K2Hag					
41 41 AA					

5. What is the filter command for listing all outgoing traffic?

```
ip.src == your_IP (IP address of the device in use)
```

Microsoft Windows [Version 10.0.26100.3037]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Link-local IPv6 Address : fe80::cdb5:1e8c:f2d7:745b%9
IPv4 Address. : 10.23.24.224
Subnet Mask : 255.255.248.0
Default Gateway : 10.23.24.1

Unknown adapter Local Area Connection:

Media State : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

Media State : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 1:

Media State : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 10:

Connection-specific DNS Suffix . :
Link-local IPv6 Address : fe80::7bec:83d1:b873:afa6%4
IPv4 Address. : 192.168.137.1
Subnet Mask : 255.255.255.0
Default Gateway :

C:\Users\user>

ip.src == 10.23.24.224					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000	10.23.24.224	172.16.100.3	DNS	83 Standard query 0xe581 A www.msftconnecttest.com
3	0.002671	10.23.24.224	23.48.245.178	TCP	66 54358 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
6	0.055531	10.23.24.224	23.48.245.178	TCP	54 54358 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
7	0.055907	10.23.24.224	23.48.245.178	HTTP	165 GET /connecttest.txt HTTP/1.1
11	0.109261	10.23.24.224	23.48.245.178	TCP	54 54358 → 80 [ACK] Seq=112 Ack=189 Win=65280 Len=0
12	0.109325	10.23.24.224	23.48.245.178	TCP	54 54358 → 80 [FIN, ACK] Seq=112 Ack=189 Win=65280 Len=0
40	1.975376	10.23.24.224	180.149.59.13	TCP	1466 54246 → 443 [ACK] Seq=1 Ack=1 Win=16383 Len=1412 [TCP PDU reassembled in 41]
41	1.975376	10.23.24.224	180.149.59.13	TLSv1.2	570 Application Data
42	1.975683	10.23.24.224	180.149.59.13	TCP	1466 54246 → 443 [ACK] Seq=1929 Ack=1 Win=16383 Len=1412 [TCP PDU reassembled in 43]
43	1.975683	10.23.24.224	180.149.59.13	TLSv1.2	700 Application Data
59	1.994586	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=12721 Win=16383 Len=0
60	1.994802	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=16957 Win=16383 Len=0
68	1.995054	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=26156 Win=16383 Len=0
85	1.995357	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=40276 Win=16383 Len=0
86	1.995692	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=48748 Win=16383 Len=0
104	1.995947	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=62868 Win=16383 Len=0
105	1.996068	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=71492 Win=16383 Len=0
122	1.996280	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=85612 Win=16383 Len=0
123	1.996387	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=94084 Win=16350 Len=0
135	1.996561	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=107228 Win=16299 Len=0
136	1.996618	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=108640 Win=16293 Len=0
140	1.996774	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=3987 Ack=110169 Win=16383 Len=0
141	2.023246	10.23.24.224	180.149.59.13	TCP	1466 54246 → 443 [ACK] Seq=3987 Ack=110169 Win=16383 Len=1412 [TCP PDU reassembled in 142]
142	2.023246	10.23.24.224	180.149.59.13	TLSv1.2	570 Application Data
143	2.023522	10.23.24.224	180.149.59.13	TCP	1466 54246 → 443 [ACK] Seq=5915 Ack=110169 Win=16383 Len=1412 [TCP PDU reassembled in 144]
144	2.023522	10.23.24.224	180.149.59.13	TLSv1.2	700 Application Data
150	2.041936	10.23.24.224	180.149.59.13	TCP	54 54246 → 443 [ACK] Seq=7973 Ack=111623 Win=16383 Len=0
152	2.333196	10.23.24.224	172.253.118.188	TCP	55 53997 → 5228 [ACK] Seq=1 Ack=1 Win=253 Len=1
170	4.408090	10.23.24.224	142.250.194.100	TLSv1.2	441 Application Data
171	4.498226	10.23.24.224	142.250.194.100	TLSv1.2	93 Application Data
172	4.498969	10.23.24.224	142.250.194.100	TLSv1.2	217 Application Data
177	4.567112	10.23.24.224	142.250.194.100	TCP	54 54334 → 443 [ACK] Seq=590 Ack=40 Win=254 Len=0
180	4.632488	10.23.24.224	142.250.194.100	TCP	54 54334 → 443 [ACK] Seq=590 Ack=1371 Win=255 Len=0
183	4.633424	10.23.24.224	142.250.194.100	TCP	54 54334 → 443 [ACK] Seq=590 Ack=1434 Win=255 Len=0

6. Start a new packet capture to now visit an external website, say www.cricinfo.com. Can you show the 3-way TCP handshake happening? Can you see your IITJ proxy in between? What is its IP address?

tcp.flags.syn == 1 || tcp.flags.ack == 1: This filter is used in Wireshark to capture packets involved in the TCP 3-way handshake by focusing on SYN and ACK flags.

tcp.flags.syn == 1 tcp.flags.ack == 1					
No.	Time	Source	Destination	Protocol	Length Info
11	0.791199	10.23.24.224	23.63.110.74	TCP	66 54663 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
12	0.831233	23.63.110.74	10.23.24.224	TCP	66 443 → 54663 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
13	0.831447	10.23.24.224	23.63.110.74	TCP	54 54663 → 443 [ACK] Seq=1 Ack=1 Win=65280 Len=0
14	0.832843	10.23.24.224	23.63.110.74	TCP	1438 54663 → 443 [ACK] Seq=1 Ack=1 Win=65280 Len=1384 [TCP PDU reassembled in 15]
15	0.832843	10.23.24.224	23.63.110.74	TLSv1.3	795 Client Hello (SNI=hs-consumer-api.espn.cricinfo.com)
16	0.833869	23.63.110.74	10.23.24.224	TCP	60 443 → 54663 [ACK] Seq=1 Ack=1385 Win=1151488 Len=0
17	0.835946	23.63.110.74	10.23.24.224	TCP	66 443 → 54663 [ACK] Seq=1 Ack=2136 Win=1151488 Len=0 SLE=1 SRE=2126
18	0.873690	23.63.110.74	10.23.24.224	TLSv1.3	318 Server Hello, Change Cipher Spec, Application Data, Application Data
19	0.874615	10.23.24.224	23.63.110.74	TLSv1.3	134 Change Cipher Spec, Application Data
20	0.874904	10.23.24.224	23.63.110.74	TLSv1.3	146 Application Data
21	0.875321	10.23.24.224	23.63.110.74	TLSv1.3	492 Application Data
22	0.914863	23.63.110.74	10.23.24.224	TCP	60 443 → 54663 [ACK] Seq=265 Ack=2206 Win=64128 Len=0
23	0.914863	23.63.110.74	10.23.24.224	TLSv1.3	357 Application Data
24	0.915007	23.63.110.74	10.23.24.224	TCP	60 443 → 54663 [ACK] Seq=568 Ack=2298 Win=64128 Len=0
25	0.915028	23.63.110.74	10.23.24.224	TLSv1.3	115 Application Data
26	0.915028	23.63.110.74	10.23.24.224	TLSv1.3	85 Application Data
27	0.915077	10.23.24.224	23.63.110.74	TCP	54 54663 → 443 [ACK] Seq=2736 Ack=660 Win=64768 Len=0
28	0.915633	10.23.24.224	23.63.110.74	TLSv1.3	85 Application Data
29	0.916276	23.63.110.74	10.23.24.224	TLSv1.3	367 Application Data
30	0.916276	23.63.110.74	10.23.24.224	TLSv1.3	87 Application Data
31	0.916385	10.23.24.224	23.63.110.74	TCP	54 54663 → 443 [ACK] Seq=2767 Ack=1006 Win=64512 Len=0
32	0.919124	10.23.24.224	23.63.110.74	TLSv1.3	331 Application Data
33	0.960149	23.63.110.74	10.23.24.224	TCP	60 443 → 54663 [ACK] Seq=1006 Ack=3044 Win=64128 Len=0
34	0.960259	23.63.110.74	10.23.24.224	TLSv1.3	442 Application Data
35	0.960259	23.63.110.74	10.23.24.224	TLSv1.3	144 Application Data
36	0.960350	10.23.24.224	23.63.110.74	TCP	54 54663 → 443 [ACK] Seq=3044 Ack=1484 Win=65280 Len=0
57	2.875470	10.23.24.224	142.250.194.100	TCP	66 54664 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
58	2.894175	142.250.194.100	10.23.24.224	TCP	66 443 → 54664 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
59	2.894415	10.23.24.224	142.250.194.100	TCP	54 54664 → 443 [ACK] Seq=1 Ack=1 Win=65280 Len=0
60	2.895884	10.23.24.224	142.250.194.100	TCP	1466 54664 → 443 [ACK] Seq=1 Ack=1 Win=65280 Len=1412 [TCP PDU reassembled in 61]
61	2.895884	10.23.24.224	142.250.194.100	TLSv1.3	884 Client Hello (SNI=www.google.com)
62	2.896871	142.250.194.100	10.23.24.224	TCP	60 443 → 54664 [ACK] Seq=1 Ack=1413 Win=1174784 Len=0
63	2.914833	142.250.194.100	10.23.24.224	TCP	66 443 → 54664 [ACK] Seq=1 Ack=2243 Win=1174784 Len=0 SLE=1 SRE=2243

Syn packets :

No.	Time	Source	Destination	Protocol	Length	Info
126	3.111167	172.31.35.71	13.215.166.201	TCP	66	51669 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
127	3.113477	172.31.35.71	13.215.166.201	TCP	66	51670 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
143	3.125040	172.31.35.71	142.250.193.10	TCP	66	51671 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
153	3.131443	172.31.35.71	13.215.166.201	TCP	66	51672 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
163	3.137938	172.31.35.71	142.250.193.10	TCP	66	51673 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
165	3.139092	142.250.193.10	172.31.35.71	TCP	66	443 → 51673 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
169	3.151746	142.250.193.10	172.31.35.71	TCP	66	443 → 51673 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
180	3.201149	13.215.166.201	172.31.35.71	TCP	66	80 → 51670 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=256
182	3.202595	13.215.166.201	172.31.35.71	TCP	66	80 → 51669 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=256
184	3.222921	13.215.166.201	172.31.35.71	TCP	66	443 → 51672 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM WS=256
238	3.375398	172.31.35.71	8.8.8.8	TCP	66	51674 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
240	3.389900	8.8.8.8	172.31.35.71	TCP	66	443 → 51674 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
291	3.499418	172.31.35.71	142.250.206.110	TCP	66	51675 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
298	3.513412	142.250.206.110	172.31.35.71	TCP	66	443 → 51675 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
316	3.528350	172.31.35.71	23.63.110.99	TCP	66	51676 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
317	3.542605	23.63.110.99	172.31.35.71	TCP	66	443 → 51676 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
476	3.627640	172.31.35.71	3.124.119.57	TCP	66	51677 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
477	3.627798	172.31.35.71	3.124.119.57	TCP	66	51678 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
478	3.627985	172.31.35.71	184.86.112.89	TCP	66	51679 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
479	3.628103	172.31.35.71	184.86.112.89	TCP	66	51680 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
480	3.628275	172.31.35.71	96.17.194.249	TCP	66	51681 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
481	3.628395	172.31.35.71	96.17.194.249	TCP	66	51682 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
482	3.628506	172.31.35.71	96.17.194.249	TCP	66	51683 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
483	3.628648	172.31.35.71	96.17.194.249	TCP	66	51684 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
484	3.628770	172.31.35.71	96.17.194.249	TCP	66	51685 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
485	3.628876	172.31.35.71	96.17.194.249	TCP	66	51686 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
498	3.641762	96.17.194.249	172.31.35.71	TCP	66	443 → 51681 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
499	3.641762	96.17.194.249	172.31.35.71	TCP	66	443 → 51684 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
500	3.641762	96.17.194.249	172.31.35.71	TCP	66	443 → 51682 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
506	3.642443	96.17.194.249	172.31.35.71	TCP	66	443 → 51686 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
515	3.644015	96.17.194.249	172.31.35.71	TCP	66	443 → 51685 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
524	3.653116	96.17.194.249	172.31.35.71	TCP	66	443 → 51683 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
567	3.650992	184.86.112.89	172.31.35.71	TCP	66	443 → 51680 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128
579	3.663556	184.86.112.89	172.31.35.71	TCP	66	443 → 51679 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128

7. Why does DNS follow the UDP stream while HTTP follows the TCP stream?

DNS primarily relies on UDP because it allows for faster query resolution. Since DNS lookups involve small request-and-response messages, using UDP keeps things lightweight and efficient. UDP doesn't require a handshake to establish a connection, reducing latency. Additionally, occasional packet loss is generally acceptable because DNS clients can simply resend the request if needed.

On the other hand, HTTP uses TCP to ensure reliable and ordered delivery of web content. Web pages include various elements like HTML, CSS, and JavaScript, all of which must arrive in the correct sequence to render properly. TCP's three-way handshake and error correction mechanisms make it ideal for delivering web content without corruption or missing data.

8. Execute the socket program (both server and client) to demonstrate TCP communication on different ports. Capture the network packets using Wireshark and analyze them to justify the communication process.

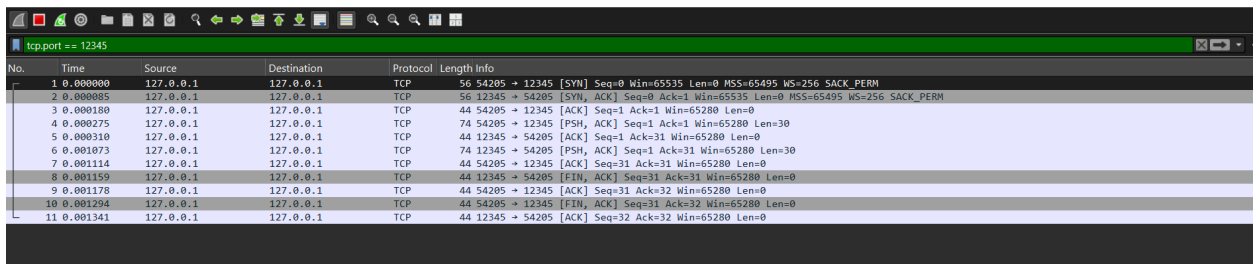
This Wireshark capture documents a local TCP connection (127.0.0.1) between the client (source port 47194) and the server (destination port 12345), demonstrating a standard TCP communication flow.

1. **Packets 6 & 8:** The client transmits application-layer data using **PSH**, **ACK** flags, ensuring immediate processing by the server. The **ACK** confirms receipt of previous segments.
2. **Packets 7 & 9:** The server acknowledges the client's data transmission with dedicated **ACK** segments, maintaining TCP's reliability mechanism.
3. **Packet 10:** The server initiates a **graceful connection termination** by sending a **FIN**, **ACK**, signaling that no further data will be transmitted.
4. **Packet 11:** The client responds with an **ACK**, formally completing the **four-way termination handshake**, adhering to TCP's standard teardown procedure.

The communication sequence follows the expected **TCP lifecycle**, including **connection establishment**, **reliable data transfer**, and **an orderly termination**. Additionally, the **TCP Timestamps (TSval, TSecr)** provide temporal markers, aiding in round-trip time (RTT) analysis and delay assessment.

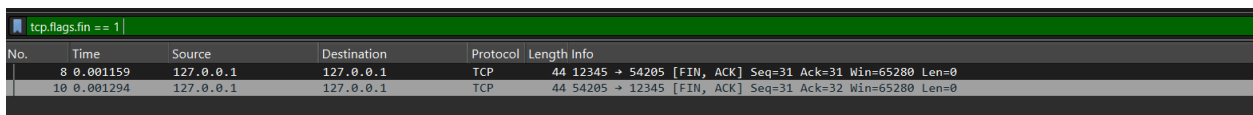
Note: The client-server architecture was pre-configured in a prior lab session.

tcp.port == 12345 : This filter isolates all TCP packets where either the source or destination port is 12345.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	56	54205 → 12345 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
2	0.000085	127.0.0.1	127.0.0.1	TCP	56	12345 → 54205 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
3	0.000180	127.0.0.1	127.0.0.1	TCP	44	54205 → 12345 [ACK] Seq=1 Ack=1 Win=65280 Len=0
4	0.000275	127.0.0.1	127.0.0.1	TCP	74	54205 → 12345 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=30
5	0.000310	127.0.0.1	127.0.0.1	TCP	44	12345 → 54205 [ACK] Seq=1 Ack=31 Win=65280 Len=0
6	0.001073	127.0.0.1	127.0.0.1	TCP	74	12345 → 54205 [PSH, ACK] Seq=1 Ack=31 Win=65280 Len=30
7	0.001114	127.0.0.1	127.0.0.1	TCP	44	54205 → 12345 [ACK] Seq=31 Ack=31 Win=65280 Len=0
8	0.001159	127.0.0.1	127.0.0.1	TCP	44	12345 → 54205 [FIN, ACK] Seq=31 Ack=31 Win=65280 Len=0
9	0.001178	127.0.0.1	127.0.0.1	TCP	44	54205 → 12345 [ACK] Seq=31 Ack=32 Win=65280 Len=0
10	0.001294	127.0.0.1	127.0.0.1	TCP	44	54205 → 12345 [FIN, ACK] Seq=31 Ack=32 Win=65280 Len=0
11	0.001341	127.0.0.1	127.0.0.1	TCP	44	12345 → 54205 [ACK] Seq=32 Ack=32 Win=65280 Len=0

tcp.flags.fin == 1 : This filter captures TCP FIN (Finish) packets, which indicate a connection termination request.



No.	Time	Source	Destination	Protocol	Length	Info
8	0.001159	127.0.0.1	127.0.0.1	TCP	44	12345 → 54205 [FIN, ACK] Seq=31 Ack=31 Win=65280 Len=0
10	0.001294	127.0.0.1	127.0.0.1	TCP	44	54205 → 12345 [FIN, ACK] Seq=31 Ack=32 Win=65280 Len=0

tcp.flags.syn == 1 || tcp.flags.ack == 1 : This filter captures:

- SYN (Synchronization) packets: Used in TCP's three-way handshake to establish a connection.
- ACK (Acknowledgment) packets: Sent to confirm receipt of data.

tcp.flags.syn == 1 tcp.flags.ack == 1					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	56 54205 → 12345 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
2	0.000003	127.0.0.1	127.0.0.1	TCP	56 12345 → 54205 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
3	0.000100	127.0.0.1	127.0.0.1	TCP	44 54205 → 12345 [ACK] Seq=1 Ack=1 Win=65280 Len=0
4	0.000275	127.0.0.1	127.0.0.1	TCP	74 54205 → 12345 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=30
5	0.000310	127.0.0.1	127.0.0.1	TCP	44 12345 → 54205 [ACK] Seq=1 Ack=31 Win=65280 Len=0
6	0.001073	127.0.0.1	127.0.0.1	TCP	74 12345 → 54205 [PSH, ACK] Seq=1 Ack=31 Win=65280 Len=30
7	0.001114	127.0.0.1	127.0.0.1	TCP	44 54205 → 12345 [ACK] Seq=31 Ack=31 Win=65280 Len=0
8	0.001159	127.0.0.1	127.0.0.1	TCP	44 12345 → 54205 [FIN, ACK] Seq=31 Ack=31 Win=65280 Len=0
9	0.001178	127.0.0.1	127.0.0.1	TCP	44 54205 → 12345 [ACK] Seq=31 Ack=32 Win=65280 Len=0
10	0.001294	127.0.0.1	127.0.0.1	TCP	44 54205 → 12345 [FIN, ACK] Seq=31 Ack=32 Win=65280 Len=0
11	0.001341	127.0.0.1	127.0.0.1	TCP	44 12345 → 54205 [ACK] Seq=32 Ack=32 Win=65280 Len=0

EEP3020 : Digital Systems Lab

Assignment - I

Name : Abhay Kashyap

Roll no : B22CS001

Question - 1. Check Greater between two input numbers

```
                PRESERVE8
                TTL      TEXT
                GLOBAL   main

                AREA     Data, DATA, READWRITE
                ALIGN
NUM1            DCD      7
NUM2            DCD      10

                AREA     Compare, CODE, READONLY
                ENTRY

main
    LDR         r0, =NUM1
    LDR         r1, =NUM2
    LDR         r2, [r0]
    LDR         r3, [r1]

    CMP         r2, r3
    BGT         num1_greater
    BLT         num2_greater
    BEQ         equal

num1_greater
    MOV         r4, r2
    B           end_comparison

num2_greater
    MOV         r4, r3
    B           end_comparison

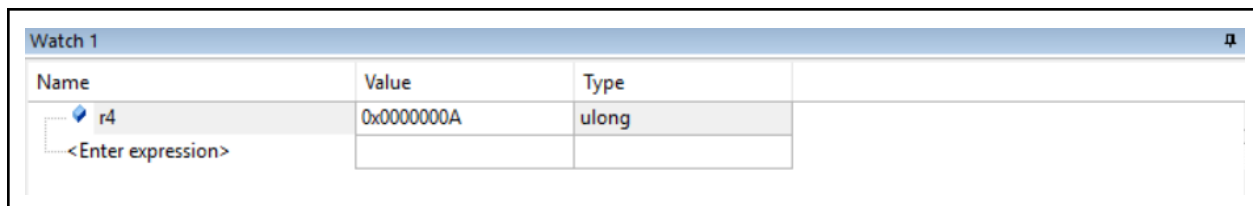
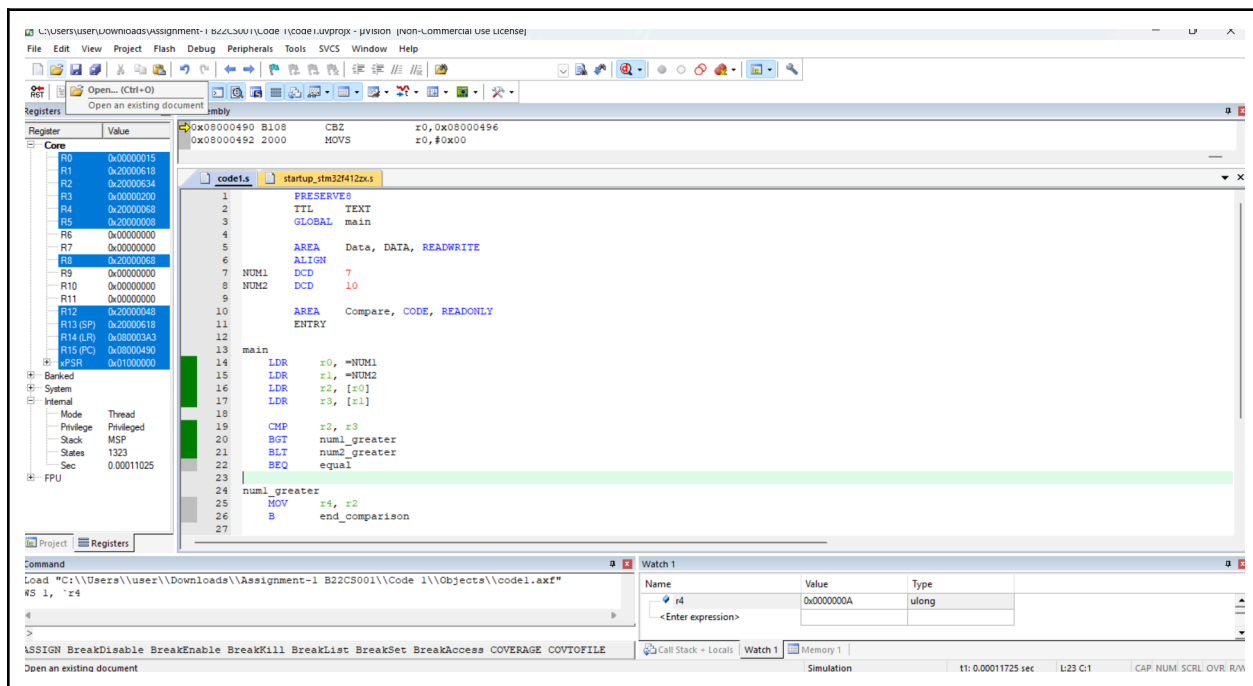
equal
    MOV         r4, r2

end_comparison
    MOV         r7, #1
    SWI         0

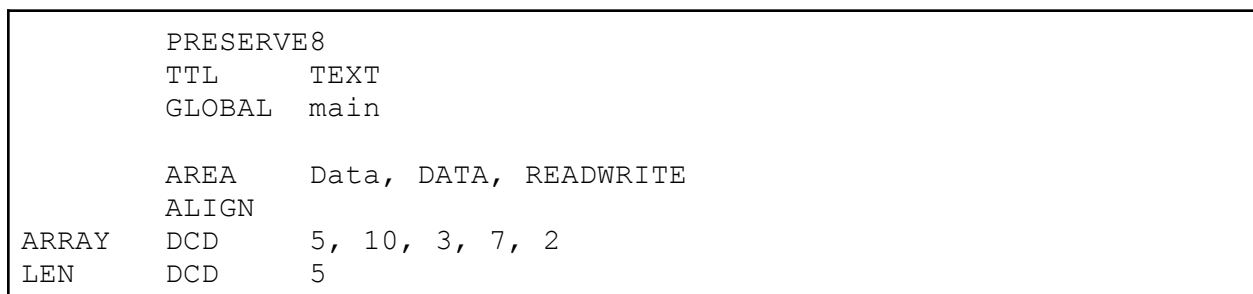
END
```

Explanation :

The ARM assembly code compares two numbers (7 and 10, stored in **NUM1** and **NUM2**) and stores the greater of the two in register **r4**. If both numbers are equal, **r4** is set to the value of **NUM1**. After the comparison, the program makes a system call to exit. The result of the comparison is controlled by conditional branching (**BGT**, **BLT**, **BEQ**).



Question - 2. Calculate the minimum one between elements of an array




```

        AREA    Compare, CODE, READONLY
        ENTRY

main
    LDR        r0, =ARRAY
    LDR        r1, =LEN
    LDR        r2, [r1]
    LDR        r3, [r0]
    MOV        r4, r3
    ADD        r0, r0, #4

loop
    CMP        r2, #1
    BEQ        end_comparison

    LDR        r5, [r0]
    CMP        r5, r4
    BGE        skip_update
    MOV        r4, r5

skip_update
    ADD        r0, r0, #4
    SUB        r2, r2, #1
    B          loop

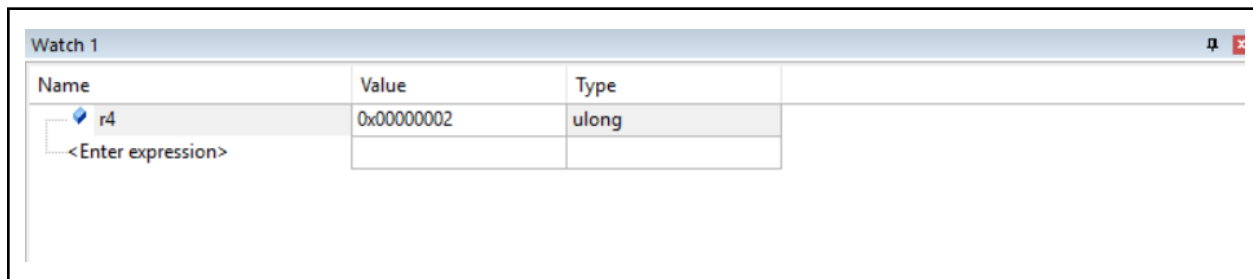
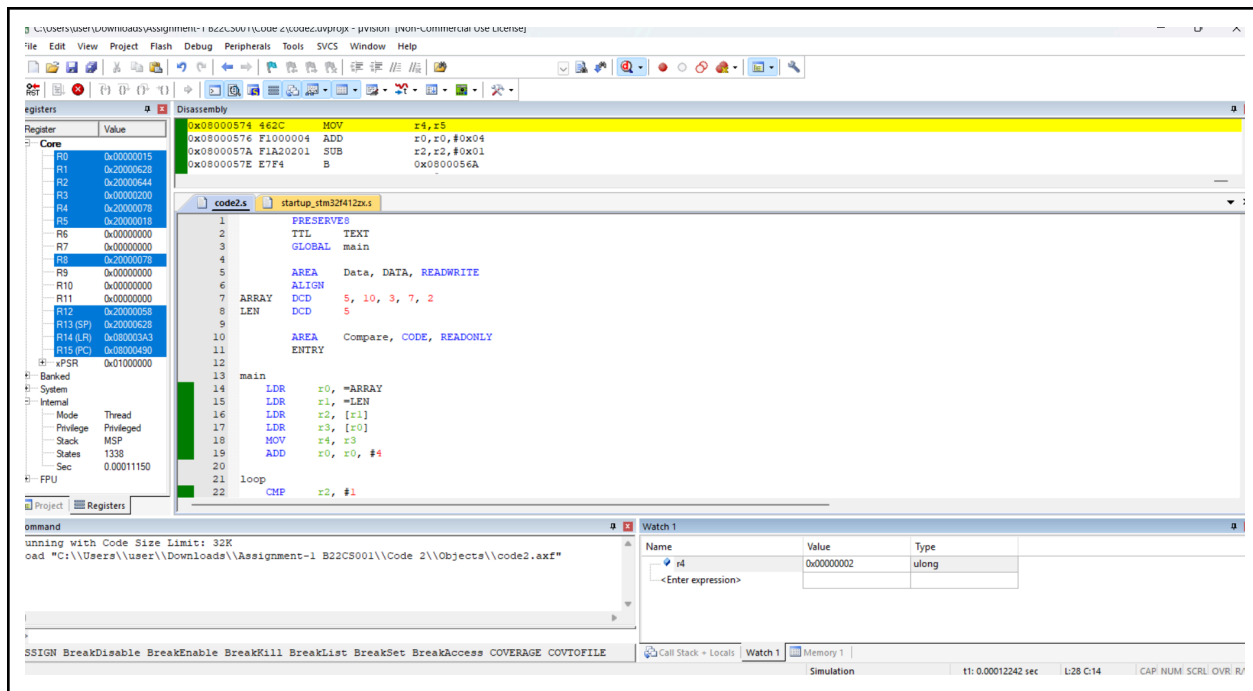
end_comparison
    MOV        r7, #1
    SWI        0

END

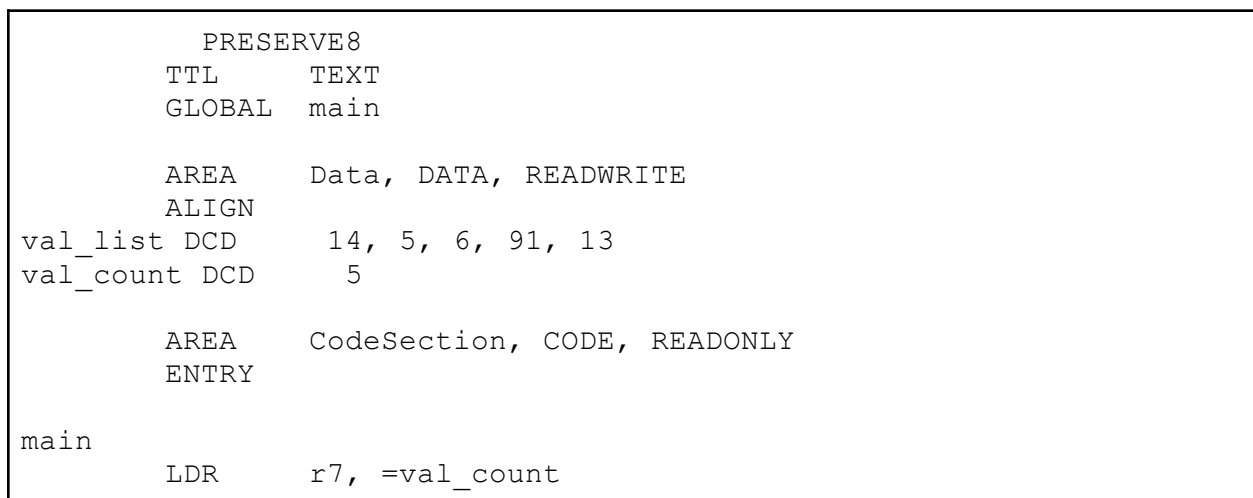
```

Explanation :

The ARM assembly code finds the minimum value in an array (**ARRAY** with values {5, 10, 3, 7, 2}) and stores it in register **r4**. It iterates through the array, comparing each element to the current minimum, and updating the minimum when a smaller value is found. The loop continues until all elements are checked, and then the program exits. The array has 5 elements, and its length is stored in **LEN**.



Question - 3. Check if a number is divisible by 7 from 5 input numbers



```

        LDR    r7, [r7]
        LDR    r8, =val_list
        MOV    r10, #0
        MOV    r11, #7

check_loop
        LDR    r12, [r8], #4
        UDIV   r2, r12, r11
        MLS    r2, r2, r11, r12
        CMP    r2, #0
        BEQ    save_val

next_loop
        SUBS   r7, r7, #1
        BNE    check_loop

end_loop
        B      end_loop

save_val
        CMP    r10, #0
        BEQ    store_r4
        CMP    r10, #1
        BEQ    store_r5
        CMP    r10, #2
        BEQ    store_r6
        CMP    r10, #3
        BEQ    store_r7
        CMP    r10, #4
        BEQ    store_r8
        B      next_loop

store_r4
        MOV    r4, r12
        ADD    r10, r10, #1
        B      next_loop

store_r5
        MOV    r5, r12
        ADD    r10, r10, #1
        B      next_loop

store_r6
        MOV    r6, r12
        ADD    r10, r10, #1
        B      next_loop

store_r7
        MOV    r7, r12
        ADD    r10, r10, #1
        B      next_loop

```

```

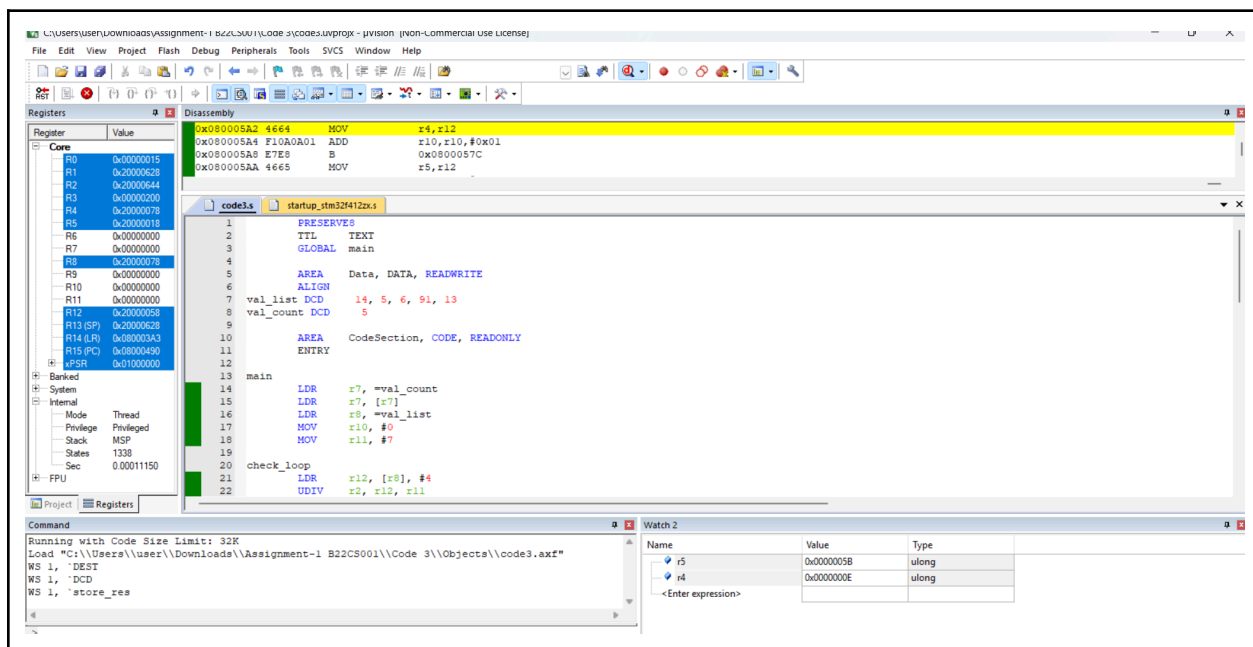
store_r8
    MOV     r8, r12
    ADD     r10, r10, #1
    B       next_loop

END

```

Explanation :

The code checks each number in the `val_list` to see if it's divisible by 7. If a number is divisible, it stores the number in one of the registers (`r4`, `r5`, `r6`, `r7`, `r8`). The loop continues until all numbers are checked, and the program ends once all divisible numbers are saved. The loop counter (`r7`) keeps track of how many numbers remain to be processed.



Watch 2		
Name	Value	Type
r5	0x0000005B	ulong
r4	0x0000000E	ulong
<Enter expression>		

Question - 4. Check if a string is a palindrome or not

```
PRESERVE8
TTL      TEXT
GLOBAL  main

AREA     RESET, CODE, READONLY
ENTRY

main
    LDR R0, =string
    MOV R1, #0
    MOV R2, #0

FIND_END
    LDRB R3, [R0, R2]
    CMP R3, #0
    BEQ BEGIN_COMPARISON
    ADD R2, R2, #1
    B FIND_END

BEGIN_COMPARISON
    SUB R2, R2, #1

LOOP
    LDRB R3, [R0, R1]
    LDRB R4, [R0, R2]
    CMP R3, R4
    BNE NOT_PALINDROME

    ADD R1, R1, #1
    SUB R2, R2, #1
    CMP R1, R2
    BNE LOOP

PALINDROME
    MOV R0, #1
    B END

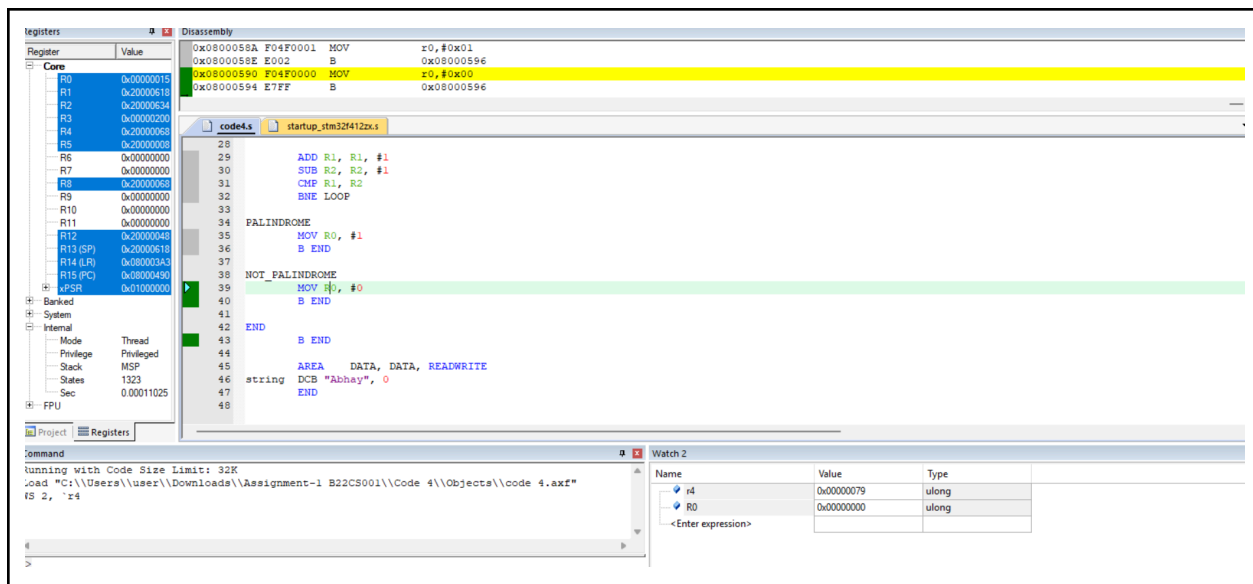
NOT_PALINDROME
    MOV R0, #0
    B END

END
    B END

AREA     DATA, DATA, READWRITE
string  DCB "Abhay", 0
END
```

Explanation :

This ARM assembly code checks if a string is a palindrome by comparing characters from both ends towards the center. It first loads the string and calculates its length by finding the null terminator. Then, it compares corresponding characters from the left and right, updating indices until either a mismatch is found (resulting in R0 being set to 0 for not a palindrome) or all characters match (resulting in R0 being set to 1 for a palindrome).



The screenshot shows the ARM Disassembler interface. The registers window on the left lists registers R0 through R15, with R0 and R1 highlighted. The disassembly window shows the assembly code for a palindrome checker. The code starts with MOV r0, #0x01 and MOV r1, #0x00. It then enters a loop to find the length of the string 'Abhay' and then compares characters from both ends. The final result is stored in R0.

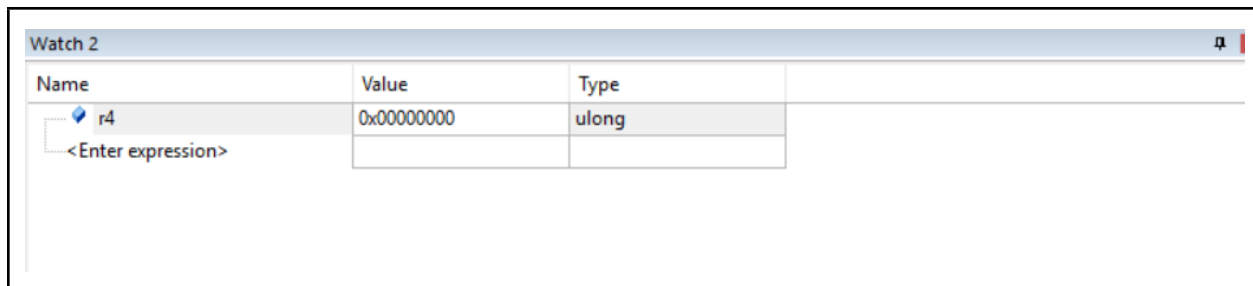
```
0x0800058A F04F0001 MOV r0, #0x01
0x0800058E E002 B 0x08000596
0x08000590 F04F0000 MOV r0, #0x00
0x08000594 E7FF B 0x08000596
```

The code then enters a loop to find the length of the string 'Abhay' and then compares characters from both ends. The final result is stored in R0.

```
28
29 ADD R1, R1, #1
30 SUB R2, R2, #1
31 CMP R1, R2
32 BNE LOOP
33
34 PALINDROME
35 MOV R0, #1
36 B END
37
38 NOT_PALINDROME
39 MOV R0, #0
40 B END
41
42 END
43
44 AREA DATA, DATA, READWRITE
45 string DSB "Abhay", 0
46
47
48
```

The Watch 2 window shows the values of R4 and R0. R4 is 0x00000079 and R0 is 0x00000000.

Name	Value	Type
r4	0x00000079	ulong
R0	0x00000000	ulong
<Enter expression>		



The Watch 2 window shows the values of R4 and R0. R4 is 0x00000000 and R0 is 0x00000000.

Name	Value	Type
r4	0x00000000	ulong
<Enter expression>		

registers

Register	Value
Core	
R0	0x00000019
R1	0x20000619
R2	0x20000634
R3	0x00000200
R4	0x20000063
R5	0x20000008
R6	0x00000000
R7	0x00000000
R8	0x20000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x20000049
R13 (SP)	0x20000619
R14 (LR)	0x00000200
R15 (PC)	0x00000490
PSR	0x01000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	1323
Sec	0 00011025
FPU	

Disassembly

0x00000490	B108	CBZ	r0,0x00000496
0x00000492	2000	MOV	r0,#0x00
0x00000494	BD1C	POP	{r2-r4,pc}
0x00000496	4620	MOV	r0,r4

code.s startup_stm32f412xx.s




28			
29	ADD	R1, R1, #1	
30	SUB	R2, R2, #1	
31	CMF	R1, R2	
32	BNE	LOOP	
33			
34	PALINDROME		
35	MOV	R0, #1	
36	B	END	
37			
38	NOT_PALINDROME		
39	MOV	R0, #0	
40	B	END	
41			
42	END		
43			
44			
45	AREA	DATA, DATA, READWRITE	
46	string	DCB "madam", 0	
47	END		
48			

command

Running with Code Size Limit: 32K
oad "C:\Users\user\Downloads\Assignment-1 B22CS001\Code 4\Objects\code 4.axf"
IS 2, 'r4'
IS 2, 'R0'

Watch 2

Name	Value	Type
r4	0x00000061	ulong
R0	0x00000001	ulong
<Enter expression>		

Watch 2		
Name	Value	Type
 r4	0x00000061	ulong
 R0	0x00000001	ulong
 <Enter expression>		