COMPUTER ENGINEERING: 4DN4

Lab 1 Report

Zhaobo Wang – 400188525
Lifeng Mei – 400256678
Zhaohan Wang – 400188640

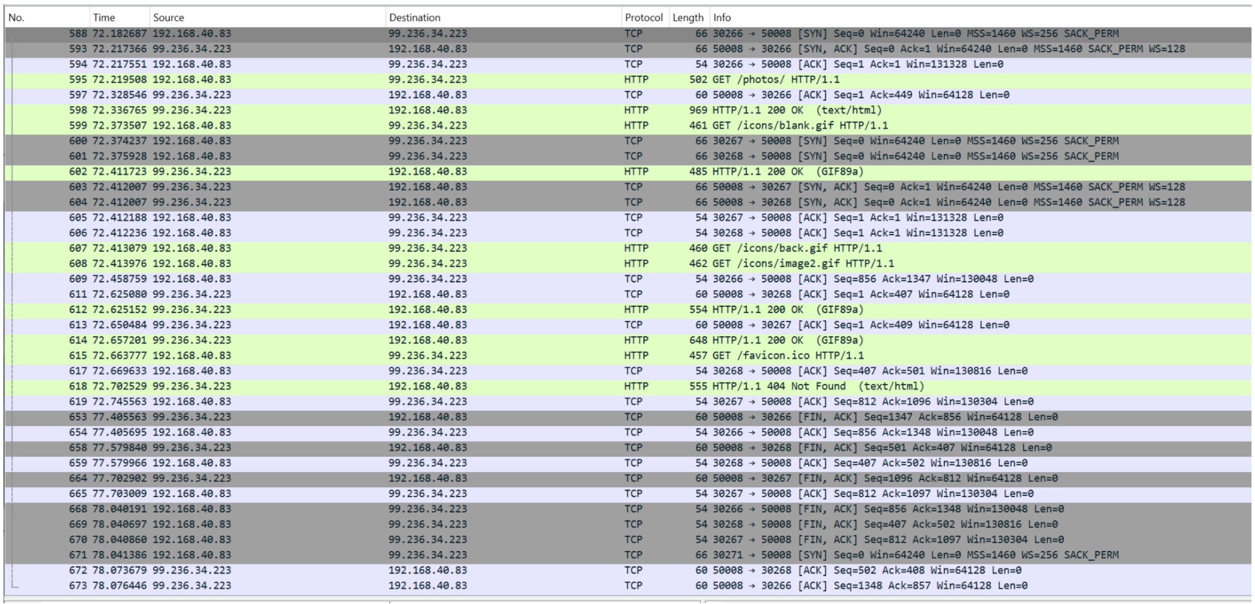
As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario. Submitted by **[Zhaobo Wang, 400188525]**

As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario. Submitted by **[LiFeng Mei, 400256678]**

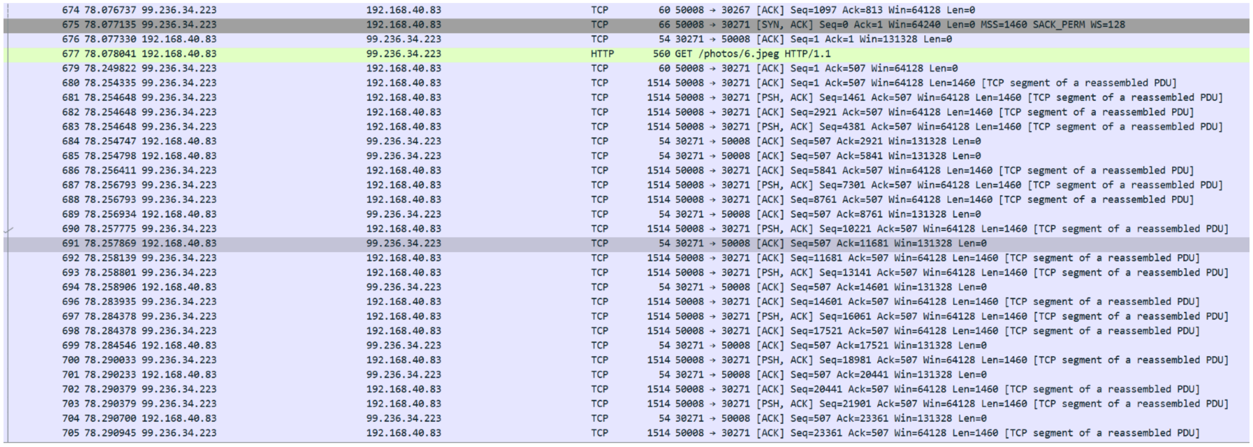
As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario. Submitted by **[Zhaohan Wang, 400188640]**

1.1 TCP

I clicked to download the above image.



No.	Time	Source	Destination	Protocol	Length	Info
588	72.182687	192.168.40.83	99.236.34.223	TCP	66	30266 → 50008 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
593	72.217366	99.236.34.223	192.168.40.83	TCP	66	50008 → 30266 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
594	72.217551	192.168.40.83	99.236.34.223	TCP	54	30266 → 50008 [ACK] Seq=1 Ack=1 Win=131328 Len=0
595	72.219508	192.168.40.83	99.236.34.223	HTTP	502	GET /photos/ HTTP/1.1
597	72.328546	99.236.34.223	192.168.40.83	TCP	60	50008 → 30266 [ACK] Seq=1 Ack=449 Win=64128 Len=0
598	72.336765	99.236.34.223	192.168.40.83	HTTP	969	HTTP/1.1 200 OK (text/html)
599	72.373507	192.168.40.83	99.236.34.223	HTTP	461	GET /icons/blank.gif HTTP/1.1
600	72.374237	192.168.40.83	99.236.34.223	TCP	66	30267 → 50008 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
601	72.375928	192.168.40.83	99.236.34.223	TCP	66	30268 → 50008 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
602	72.411723	99.236.34.223	192.168.40.83	HTTP	485	HTTP/1.1 200 OK (GIF89a)
603	72.412007	99.236.34.223	192.168.40.83	TCP	66	50008 → 30267 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
604	72.412007	99.236.34.223	192.168.40.83	TCP	66	50008 → 30268 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
605	72.412188	192.168.40.83	99.236.34.223	TCP	54	30267 → 50008 [ACK] Seq=1 Ack=1 Win=131328 Len=0
606	72.412236	192.168.40.83	99.236.34.223	TCP	54	30268 → 50008 [ACK] Seq=1 Ack=1 Win=131328 Len=0
607	72.413079	192.168.40.83	99.236.34.223	HTTP	460	GET /icons/back.gif HTTP/1.1
608	72.413976	192.168.40.83	99.236.34.223	HTTP	462	GET /icons/image2.gif HTTP/1.1
609	72.458759	192.168.40.83	99.236.34.223	TCP	54	30266 → 50008 [ACK] Seq=856 Ack=1347 Win=130048 Len=0
611	72.625080	99.236.34.223	192.168.40.83	TCP	60	50008 → 30268 [ACK] Seq=1 Ack=407 Win=64128 Len=0
612	72.625152	99.236.34.223	192.168.40.83	HTTP	554	HTTP/1.1 200 OK (GIF89a)
613	72.650484	99.236.34.223	192.168.40.83	TCP	60	50008 → 30267 [ACK] Seq=1 Ack=409 Win=64128 Len=0
614	72.657201	99.236.34.223	192.168.40.83	HTTP	648	HTTP/1.1 200 OK (GIF89a)
615	72.663777	192.168.40.83	99.236.34.223	HTTP	457	GET /favicon.ico HTTP/1.1
617	72.669633	192.168.40.83	99.236.34.223	TCP	54	30268 → 50008 [ACK] Seq=407 Ack=501 Win=130816 Len=0
618	72.702529	99.236.34.223	192.168.40.83	HTTP	555	HTTP/1.1 404 Not Found (text/html)
619	72.745563	192.168.40.83	99.236.34.223	TCP	54	30267 → 50008 [ACK] Seq=812 Ack=1096 Win=130304 Len=0
653	77.405563	99.236.34.223	192.168.40.83	TCP	60	50008 → 30266 [FIN, ACK] Seq=1347 Ack=856 Win=64128 Len=0
654	77.405695	192.168.40.83	99.236.34.223	TCP	54	30266 → 50008 [ACK] Seq=856 Ack=1348 Win=130048 Len=0
658	77.579840	99.236.34.223	192.168.40.83	TCP	60	50008 → 30268 [FIN, ACK] Seq=501 Ack=407 Win=64128 Len=0
659	77.579966	192.168.40.83	99.236.34.223	TCP	54	30268 → 50008 [ACK] Seq=407 Ack=502 Win=130816 Len=0
664	77.702902	99.236.34.223	192.168.40.83	TCP	60	50008 → 30267 [FIN, ACK] Seq=1096 Ack=812 Win=64128 Len=0
665	77.703009	192.168.40.83	99.236.34.223	TCP	54	30267 → 50008 [ACK] Seq=812 Ack=1097 Win=130304 Len=0
668	78.040191	192.168.40.83	99.236.34.223	TCP	54	30266 → 50008 [FIN, ACK] Seq=856 Ack=1348 Win=130048 Len=0
669	78.040697	192.168.40.83	99.236.34.223	TCP	54	30268 → 50008 [FIN, ACK] Seq=407 Ack=502 Win=130816 Len=0
670	78.040860	192.168.40.83	99.236.34.223	TCP	54	30267 → 50008 [FIN, ACK] Seq=812 Ack=1097 Win=130304 Len=0
671	78.041386	192.168.40.83	99.236.34.223	TCP	66	30271 → 50008 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
672	78.073679	99.236.34.223	192.168.40.83	TCP	60	50008 → 30268 [ACK] Seq=502 Ack=408 Win=64128 Len=0
673	78.076446	99.236.34.223	192.168.40.83	TCP	60	50008 → 30266 [ACK] Seq=1348 Ack=857 Win=64128 Len=0



674	78.076737	99.236.34.223	192.168.40.83	TCP	60	50008 → 30267 [ACK] Seq=1097 Ack=813 Win=64128 Len=0
675	78.077135	99.236.34.223	192.168.40.83	TCP	66	50008 → 30271 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
676	78.077330	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=1 Ack=1 Win=131328 Len=0
677	78.078041	192.168.40.83	99.236.34.223	HTTP	560	GET /photos/6.jpeg HTTP/1.1
679	78.249822	99.236.34.223	192.168.40.83	TCP	60	50008 → 30271 [ACK] Seq=1 Ack=507 Win=64128 Len=0
680	78.254335	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=1 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
681	78.254648	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=1461 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
682	78.254648	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=2921 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
683	78.254648	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=4381 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
684	78.254747	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=2921 Win=131328 Len=0
685	78.254798	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=5841 Win=131328 Len=0
686	78.256411	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=5841 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
687	78.256793	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=7301 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
688	78.256793	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=8761 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
689	78.256934	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=8761 Win=131328 Len=0
690	78.257775	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=10221 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
691	78.257869	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=11681 Win=131328 Len=0
692	78.258139	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=11681 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
693	78.258801	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=13141 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
694	78.258906	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=14601 Win=131328 Len=0
696	78.283935	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=14601 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
697	78.284378	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=16061 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
698	78.284378	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=17521 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
699	78.284546	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=17521 Win=131328 Len=0
700	78.290033	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=18981 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
701	78.290233	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=20441 Win=131328 Len=0
702	78.290379	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=20441 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
703	78.290379	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [PSH, ACK] Seq=21901 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
704	78.290700	192.168.40.83	99.236.34.223	TCP	54	30271 → 50008 [ACK] Seq=507 Ack=23361 Win=131328 Len=0
705	78.290945	99.236.34.223	192.168.40.83	TCP	1514	50008 → 30271 [ACK] Seq=23361 Ack=507 Win=64128 Len=1460 [TCP segment of a reassembled PDU]

Here is the wireshark packet capture

My source IP address is 192.168.40.83, and the destination IP address is 99.236.34.223. After setting the capture filter in Wireshark, I initiated an HTTP request to the website http://compeng4dn4.mooo.com/.

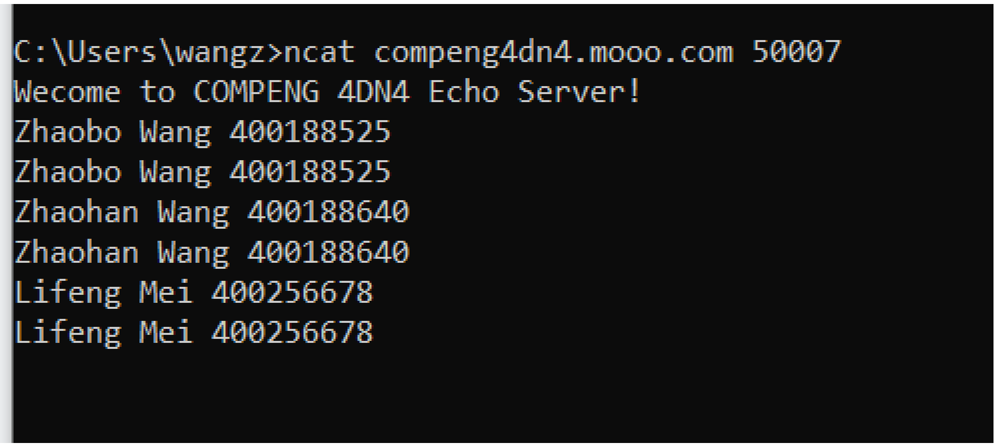
TCP communication begins with a three-way handshake, which includes SYN, SYN-ACK, and ACK packets. For instance, entry No. 593 illustrates a SYN-ACK packet, marking the second step of the handshake process.

The capture shows an HTTP GET method for fetching the URL, which retrieves items from the /icons directory with an HTTP status code of 200 OK. Entry No. 608 indicates that the client has requested to download an image.

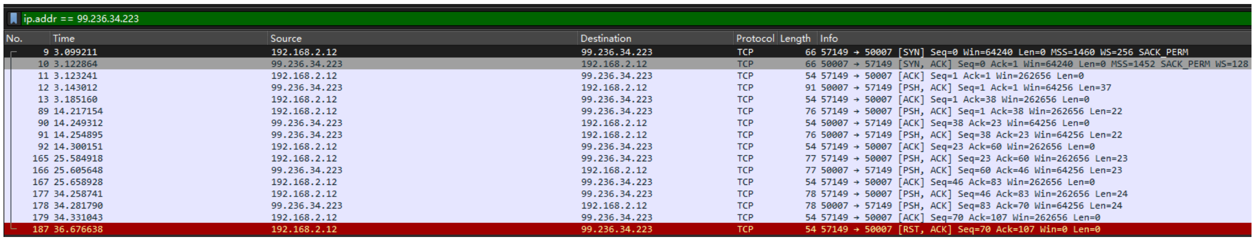
TCP acknowledgments (ACKs) are sent by the client to confirm the successful reception of packets from the server. For example, entry No. 679 displays an ACK packet, acknowledging the data previously received.

Sequence numbers (Seq) and acknowledgment numbers (Ack) are used by TCP to maintain the order and integrity of data transmission. Each TCP packet carries a sequence number, while acknowledgment packets contain the sequence number of the next expected byte.

1.2 TCP



```
C:\Users\wangz>ncat compeng4dn4.mooo.com 50007
Wecome to COMPENG 4DN4 Echo Server!
Zhaobo Wang 400188525
Zhaobo Wang 400188525
Zhaohan Wang 400188640
Zhaohan Wang 400188640
Lifeng Mei 400256678
Lifeng Mei 400256678
```



ip.addr == 99.236.34.223

No.	Time	Source	Destination	Protocol	Length	Info
9	3.099211	192.168.2.12	99.236.34.223	TCP	66	57149 → 50007 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
10	3.122864	99.236.34.223	192.168.2.12	TCP	66	50007 → 57149 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1452 SACK_PERM WS=128
11	3.123241	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [ACK] Seq=1 Ack=1 Win=262656 Len=0
12	3.143012	99.236.34.223	192.168.2.12	TCP	91	50007 → 57149 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=37
13	3.185160	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [ACK] Seq=1 Ack=38 Win=262656 Len=0
89	14.217154	192.168.2.12	99.236.34.223	TCP	76	57149 → 50007 [PSH, ACK] Seq=1 Ack=38 Win=262656 Len=22
90	14.249312	99.236.34.223	192.168.2.12	TCP	54	50007 → 57149 [ACK] Seq=38 Ack=23 Win=64256 Len=0
91	14.254895	99.236.34.223	192.168.2.12	TCP	76	50007 → 57149 [PSH, ACK] Seq=38 Ack=23 Win=64256 Len=22
92	14.300151	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [ACK] Seq=23 Ack=60 Win=262656 Len=0
165	25.584918	192.168.2.12	99.236.34.223	TCP	77	57149 → 50007 [PSH, ACK] Seq=23 Ack=60 Win=262656 Len=23
166	25.605648	99.236.34.223	192.168.2.12	TCP	77	50007 → 57149 [PSH, ACK] Seq=60 Ack=46 Win=64256 Len=23
167	25.658928	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [ACK] Seq=46 Ack=83 Win=262656 Len=0
177	34.258741	192.168.2.12	99.236.34.223	TCP	78	57149 → 50007 [PSH, ACK] Seq=46 Ack=83 Win=262656 Len=24
178	34.281790	99.236.34.223	192.168.2.12	TCP	78	50007 → 57149 [PSH, ACK] Seq=83 Ack=70 Win=64256 Len=24
179	34.331043	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [ACK] Seq=70 Ack=107 Win=262656 Len=0
187	36.676638	192.168.2.12	99.236.34.223	TCP	54	57149 → 50007 [RST, ACK] Seq=70 Ack=107 Win=0 Len=0

Three-way handshake TCP connection

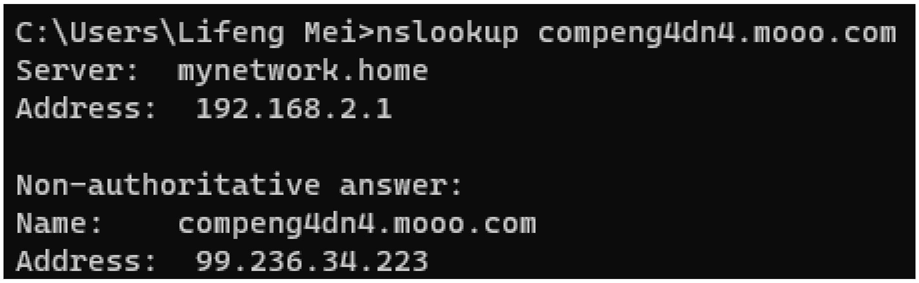
No.9/10/11: This is likely the initial SYN packet from the source to the destination to request a connection. The SYN flag is set, which means the source is initiating a TCP connection. No. 4600: This is the SYN-ACK packet from the destination to the source. The SYN and ACK flags are set No. 4601: This is the final ACK packet from the source to the destination. The ACK flag is set, which completes the three-way handshake process and establishes the TCP connection.

Data Transfer

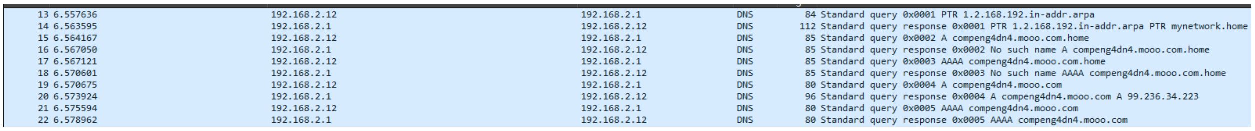
Packets with [PSH, ACK]: These packets carry the actual data payload. The PSH (Push) flag tells the receiver to push the received data to the application as soon as possible, and the ACK (Acknowledge) flag is used for acknowledging the received data. The sequence and acknowledgment numbers indicate the order of the bytes in the stream and the bytes that have been successfully received.

No.187: Utilizing the Ctrl + C keyboard shortcut terminated the terminal process, which triggered an sudden transition to the TCP RST (Reset) state, resulting in an immediate cessation of the TCP session. This action led to an ungraceful termination of the TCP connection rather than an orderly shutdown. The graceful termination should end with FIN.

<u>1.3 DNS</u>



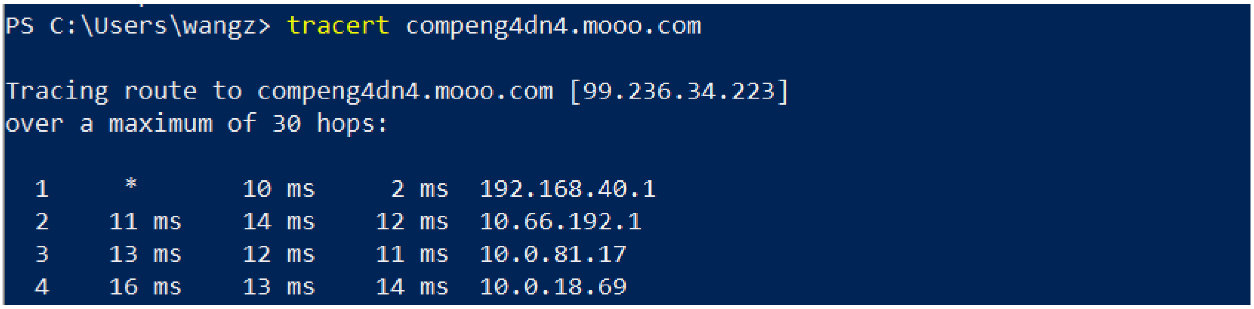
```
C:\Users\Lifeng Mei>nslookup compeng4dn4.mooo.com
Server:  mynetwork.home
Address:  192.168.2.1

Non-authoritative answer:
Name:    compeng4dn4.mooo.com
Address:  99.236.34.223
```



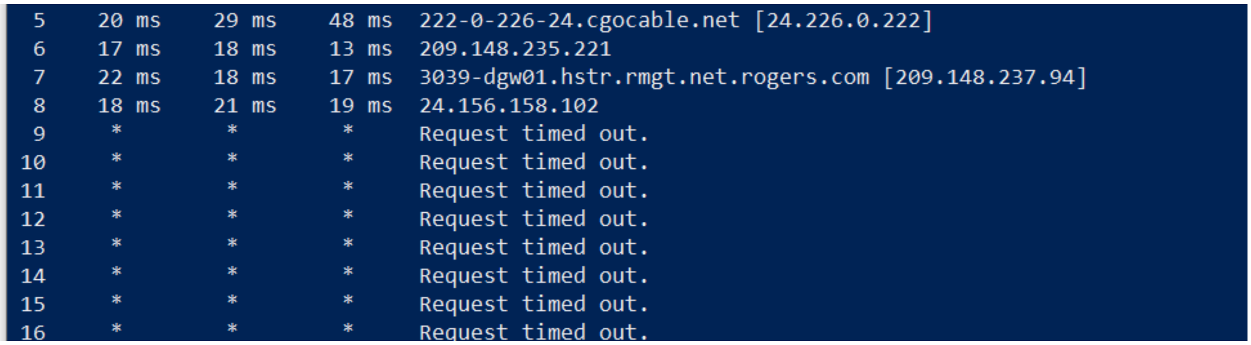
```
13 6.557636   192.168.2.12   192.168.2.1    DNS   84 Standard query 0x0001 PTR 1.2.168.192.in-addr.arpa
14 6.563595   192.168.2.1    192.168.2.12   DNS  112 Standard query response 0x0001 PTR 1.2.168.192.in-addr.arpa PTR mynetwork.home
15 6.564167   192.168.2.12   192.168.2.1    DNS   85 Standard query 0x0002 A compeng4dn4.mooo.com.home
16 6.567050   192.168.2.1    192.168.2.12   DNS   85 Standard query response 0x0002 No such name A compeng4dn4.mooo.com.home
17 6.567121   192.168.2.12   192.168.2.1    DNS   85 Standard query 0x0003 AAAA compeng4dn4.mooo.com.home
18 6.570601   192.168.2.1    192.168.2.12   DNS   85 Standard query response 0x0003 No such name AAAA compeng4dn4.mooo.com.home
19 6.570675   192.168.2.12   192.168.2.1    DNS   80 Standard query 0x0004 A compeng4dn4.mooo.com
20 6.573924   192.168.2.1    192.168.2.12   DNS   96 Standard query response 0x0004 A compeng4dn4.mooo.com A 99.236.34.223
21 6.575594   192.168.2.12   192.168.2.1    DNS   80 Standard query 0x0005 AAAA compeng4dn4.mooo.com
22 6.578962   192.168.2.1    192.168.2.12   DNS   80 Standard query response 0x0005 AAAA compeng4dn4.mooo.com
```

No.13 It shows PTR record queries for an IPv4 address, which are typically used for reverse DNS lookups.
No.15 shows a standard DNS query where the client requests the A record for the domain name “compeng4dn4.mooo.com”
No.18&19 shows AAAA record queries for the same domain name, where the client is requesting the IPv4 address of the domain.

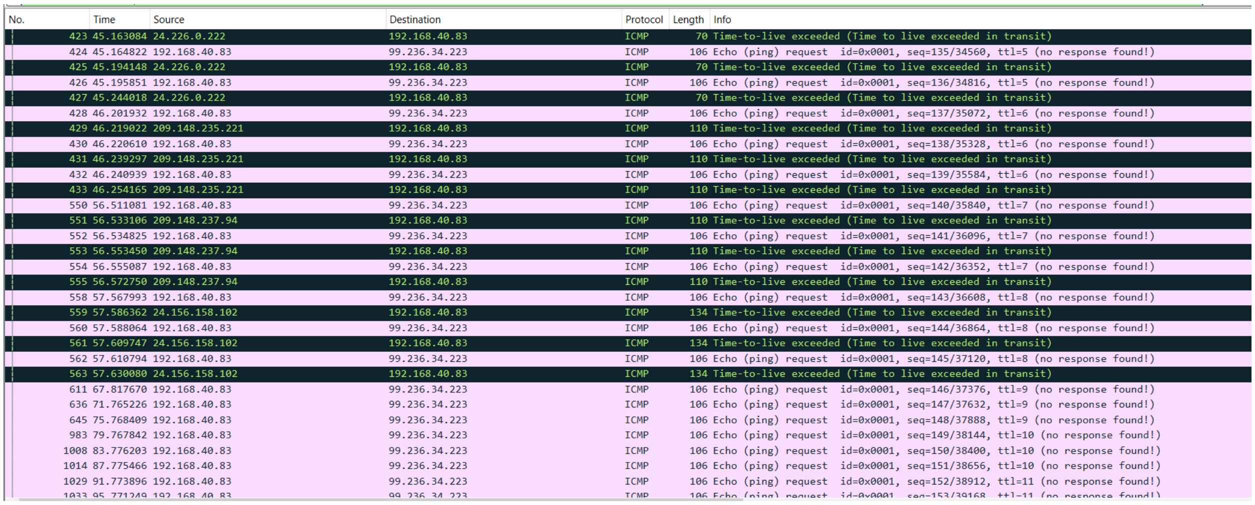
1.4 Traceroute



```
PS C:\Users\wangz> tracert compeng4dn4.mooo.com

Tracing route to compeng4dn4.mooo.com [99.236.34.223]
over a maximum of 30 hops:

  1     *       10 ms     2 ms  192.168.40.1
  2    11 ms    14 ms    12 ms  10.66.192.1
  3    13 ms    12 ms    11 ms  10.0.81.17
  4    16 ms    13 ms    14 ms  10.0.18.69
```



```
  5    20 ms    29 ms    48 ms  222-0-226-24.cgocable.net [24.226.0.222]
  6    17 ms    18 ms    13 ms  209.148.235.221
  7    22 ms    18 ms    17 ms  3039-dgw01.hstr.rmgt.net.rogers.com [209.148.237.94]
  8    18 ms    21 ms    19 ms  24.156.158.102
  9     *        *        *     Request timed out.
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15     *        *        *     Request timed out.
 16     *        *        *     Request timed out.
```



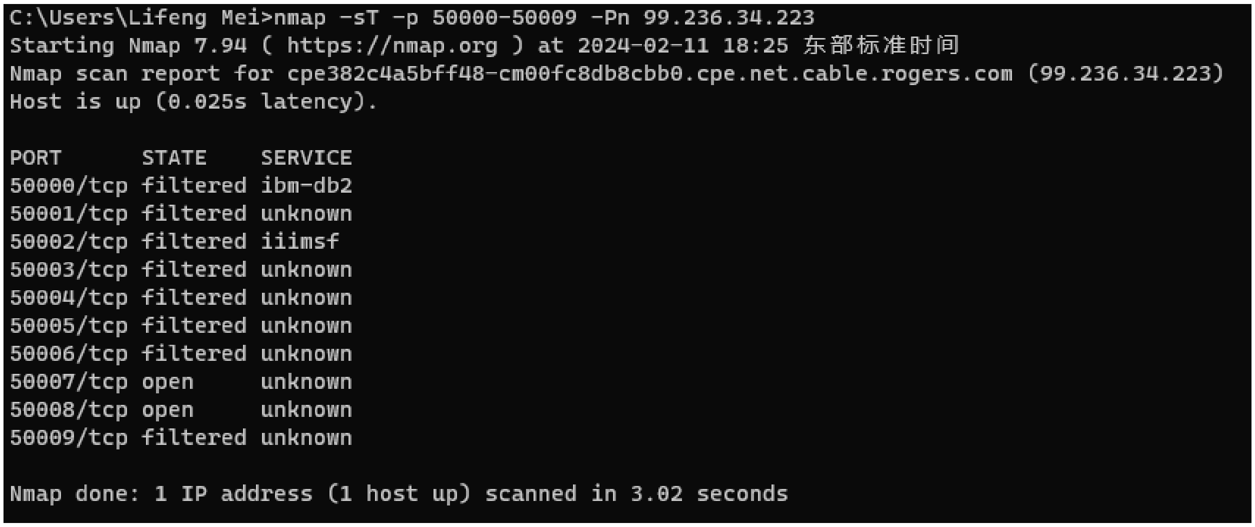
```
No.   Time      Source           Destination     Protocol Length Info
  423 45.163084 24.226.0.222     192.168.40.83   ICMP      70 Time-to-live exceeded (Time to live exceeded in transit)
  424 45.164822 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=135/34560, ttl=5 (no response found!)
  425 45.194148 24.226.0.222     192.168.40.83   ICMP      70 Time-to-live exceeded (Time to live exceeded in transit)
  426 45.195851 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=136/34816, ttl=5 (no response found!)
  427 45.244018 24.226.0.222     192.168.40.83   ICMP      70 Time-to-live exceeded (Time to live exceeded in transit)
  428 46.201932 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=137/35072, ttl=6 (no response found!)
  429 46.219022 209.148.235.221  192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  430 46.220610 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=138/35328, ttl=6 (no response found!)
  431 46.239297 209.148.235.221  192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  432 46.240939 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=139/35584, ttl=6 (no response found!)
  433 46.254165 209.148.235.221  192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  550 56.511081 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=140/35840, ttl=7 (no response found!)
  551 56.533106 209.148.237.94   192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  552 56.534825 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=141/36096, ttl=7 (no response found!)
  553 56.553450 209.148.237.94   192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  554 56.555087 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=142/36352, ttl=7 (no response found!)
  555 56.572750 209.148.237.94   192.168.40.83   ICMP     110 Time-to-live exceeded (Time to live exceeded in transit)
  558 57.567993 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=143/36608, ttl=8 (no response found!)
  559 57.586362 24.156.158.102   192.168.40.83   ICMP     134 Time-to-live exceeded (Time to live exceeded in transit)
  560 57.588064 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=144/36864, ttl=8 (no response found!)
  561 57.609747 24.156.158.102   192.168.40.83   ICMP     134 Time-to-live exceeded (Time to live exceeded in transit)
  562 57.610794 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=145/37120, ttl=8 (no response found!)
  563 57.630080 24.156.158.102   192.168.40.83   ICMP     134 Time-to-live exceeded (Time to live exceeded in transit)
  611 67.817670 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=146/37376, ttl=9 (no response found!)
  636 71.765226 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=147/37632, ttl=9 (no response found!)
  645 75.768409 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=148/37888, ttl=9 (no response found!)
  983 79.767842 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=149/38144, ttl=10 (no response found!)
 1008 83.776203 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=150/38400, ttl=10 (no response found!)
 1014 87.775466 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=151/38656, ttl=10 (no response found!)
 1029 91.773896 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=152/38912, ttl=11 (no response found!)
 1033 95.771249 192.168.40.83    99.236.34.223   ICMP     106 Echo (ping) request  id=0x0001, seq=153/39168, ttl=11 (no response found!)
```

Tracert is a tool used to trace the path to a target host by sending echo requests, it is tracing each hop across the routers. When an echo request is sent from my host with the source IP of 192.168.40.83, the routers being traced respond immediately with an ICMP message using their source IP, to the destination IP address of 192.168.40.83. As the TTL of each packet is exhausted, it's necessary to increment the TTL to extend the route's path, eventually discovering the complete route. However, from the image above, we observe that halfway through there is a timeout. The echo request sent from 192.168.40.83 to 99.236.34.223 did not receive a response, which could be due to a variety of reasons, such as hop count limits or network congestion and so on.

We think tracert is finding the root by constantly sending ICMP packets with increasing TTL.

2 NMAP

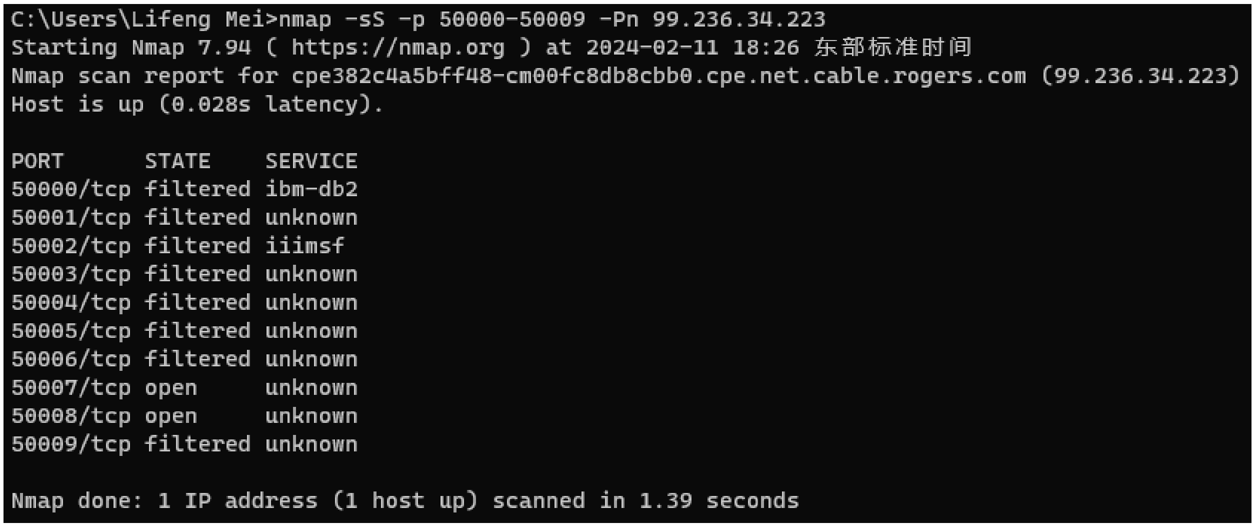
1.



```
C:\Users\Lifeng Mei>nmap -sT -p 50000-50009 -Pn 99.236.34.223
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 18:25 东部标准时间
Nmap scan report for cpe382c4a5bff48-cm00fc8db8cbb0.cpe.net.cable.rogers.com (99.236.34.223)
Host is up (0.025s latency).

PORT      STATE    SERVICE
50000/tcp filtered ibm-db2
50001/tcp filtered unknown
50002/tcp filtered iiimsf
50003/tcp filtered unknown
50004/tcp filtered unknown
50005/tcp filtered unknown
50006/tcp filtered unknown
50007/tcp open     unknown
50008/tcp open     unknown
50009/tcp filtered unknown

Nmap done: 1 IP address (1 host up) scanned in 3.02 seconds
```

We can see a Nmap scan report for the server over the port 50000-50009. The figure above shows the state of each port, port 50007 and 50008 are open, this is also why we were connecting port 50007 in the previous experiment. And the report was generated in 3.02 seconds.

2.

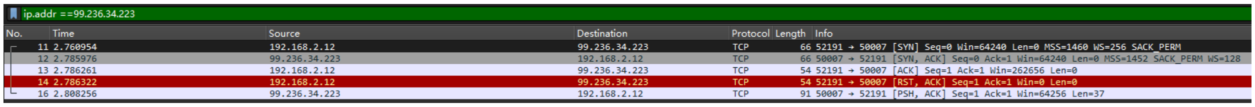


```
C:\Users\Lifeng Mei>nmap -sS -p 50000-50009 -Pn 99.236.34.223
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 18:26 东部标准时间
Nmap scan report for cpe382c4a5bff48-cm00fc8db8cbb0.cpe.net.cable.rogers.com (99.236.34.223)
Host is up (0.028s latency).

PORT      STATE    SERVICE
50000/tcp filtered ibm-db2
50001/tcp filtered unknown
50002/tcp filtered iiimsf
50003/tcp filtered unknown
50004/tcp filtered unknown
50005/tcp filtered unknown
50006/tcp filtered unknown
50007/tcp open     unknown
50008/tcp open     unknown
50009/tcp filtered unknown

Nmap done: 1 IP address (1 host up) scanned in 1.39 seconds
```

Same as what we saw in standard scan, we can see the report with the state of ports. The time to generate the report with SYN scan is 1.39 seconds and it’s faster than standard scan.

3.

Standard Scan:

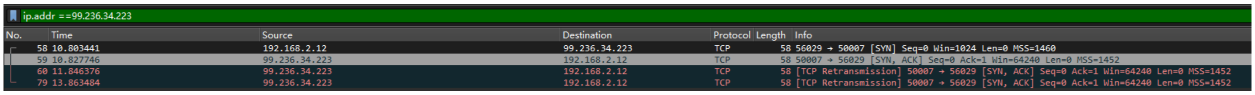


ip.addr ==99.236.34.223

No.	Time	Source	Destination	Protocol	Length	Info
11	2.760954	192.168.2.12	99.236.34.223	TCP	66	52191 → 50007 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
12	2.785976	99.236.34.223	192.168.2.12	TCP	66	50007 → 52191 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1452 SACK_PERM WS=128
13	2.786261	192.168.2.12	99.236.34.223	TCP	54	52191 → 50007 [ACK] Seq=1 Ack=1 Win=262656 Len=0
14	2.786322	192.168.2.12	99.236.34.223	TCP	54	52191 → 50007 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16	2.808256	99.236.34.223	192.168.2.12	TCP	91	50007 → 52191 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=37

As we can see in the above figure, the standard scan was trying to establish a full TCP connection with the target port, that is why we can see the 3-way handshakes.

SYN Scan:

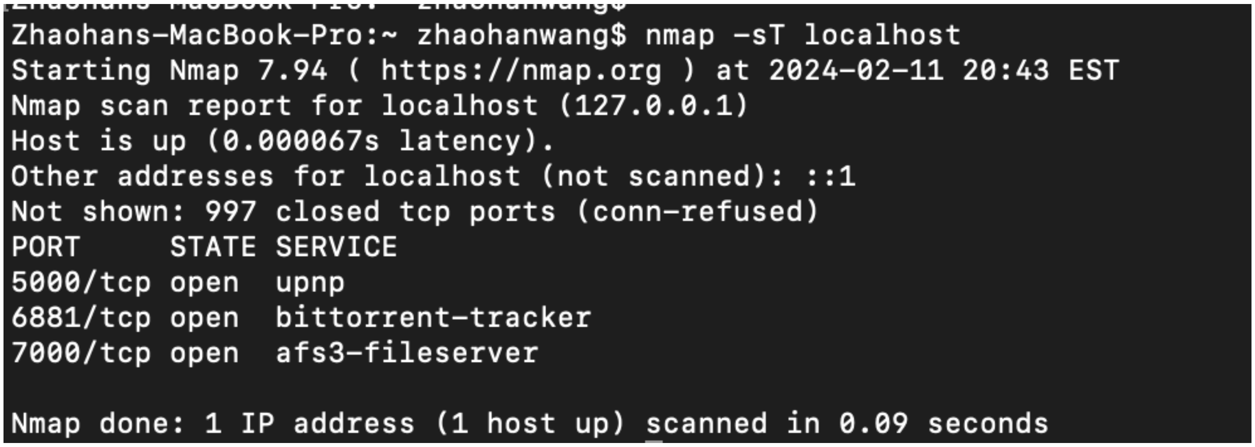


ip.addr ==99.236.34.223

No.	Time	Source	Destination	Protocol	Length	Info
58	10.803441	192.168.2.12	99.236.34.223	TCP	58	56029 → 50007 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
59	10.827746	99.236.34.223	192.168.2.12	TCP	58	50007 → 56029 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1452
60	11.846376	99.236.34.223	192.168.2.12	TCP	58	[TCP Retransmission] 50007 → 56029 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1452
79	13.863484	99.236.34.223	192.168.2.12	TCP	58	[TCP Retransmission] 50007 → 56029 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1452

A full connection is not established in SYN Scan, it relies on the response of the first SYN to identify the state of the port.

4.



```
Zhaohans-MacBook-Pro:~ zhaohanwang$
Zhaohans-MacBook-Pro:~ zhaohanwang$ nmap -sT localhost
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 20:43 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000067s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 997 closed tcp ports (conn-refused)
PORT     STATE SERVICE
5000/tcp open  upnp
6881/tcp open  bittorrent-tracker
7000/tcp open  afs3-fileserver

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
```

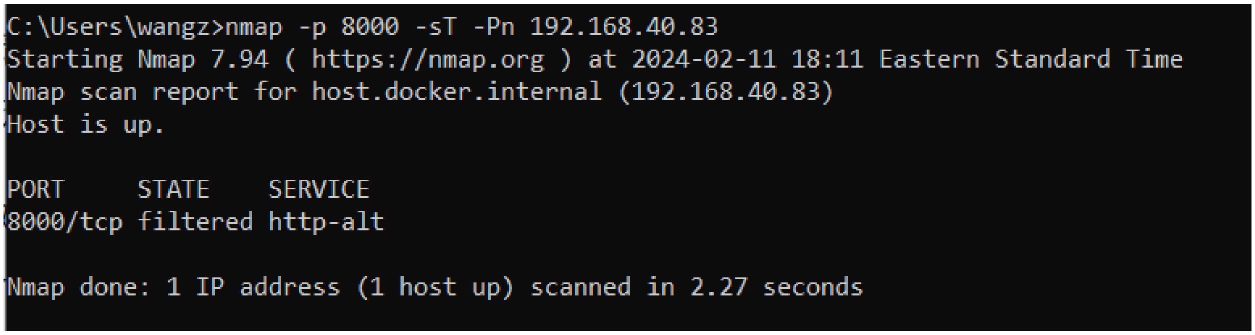
If I scan my own host we can see the port 5000, 6881, and 7000 are open, and the IP address scanned in 0.09 sec which is faster than scanning the echo server host.
In my local host, i got upnp, bittorrent-tracker, and afs3-fileserver which means,

UPnP (Universal Plug and Play)
UPnP enables automatic discovery and connection of network devices, simplifying data sharing and communication without manual configuration.

BitTorrent Tracker
A BitTorrent Tracker facilitates the connection between peers in the BitTorrent network to speed up and manage file sharing efficiently.

AFS3 Fileserver (Andrew File System Version 3 File Server)
The AFS3 Fileserver offers a scalable and secure distributed file system, enhancing file access and management across networks.

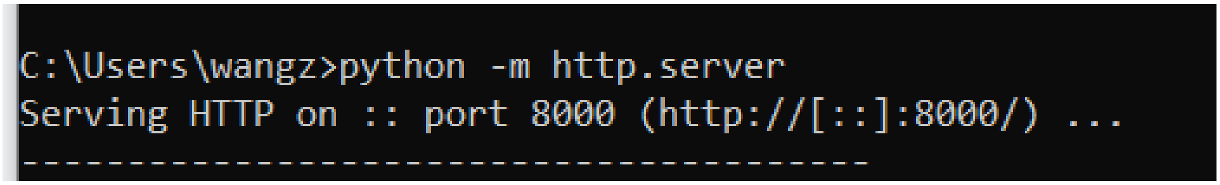
5.



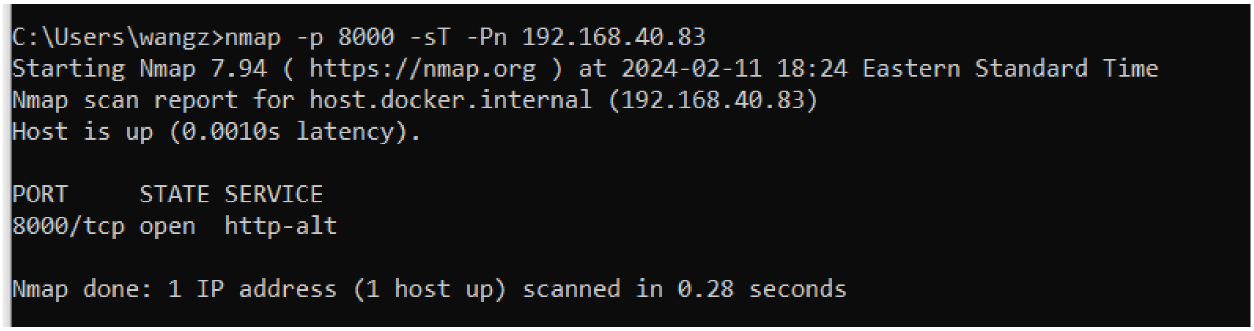
```
C:\Users\wangz>nmap -p 8000 -sT -Pn 192.168.40.83
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 18:11 Eastern Standard Time
Nmap scan report for host.docker.internal (192.168.40.83)
Host is up.

PORT     STATE    SERVICE
8000/tcp filtered http-alt

Nmap done: 1 IP address (1 host up) scanned in 2.27 seconds
```

Using nmap command to do a standard scan in my IP address for port 8000, as the screenshot shown above, the Port 8000 has filtered state. There will contain three possible results when I scan a port. They are labelled as Open/ Closed/ Filtered. Due to it showed a filtered state, it indicated that Port 8000 is not available at this time.

6.



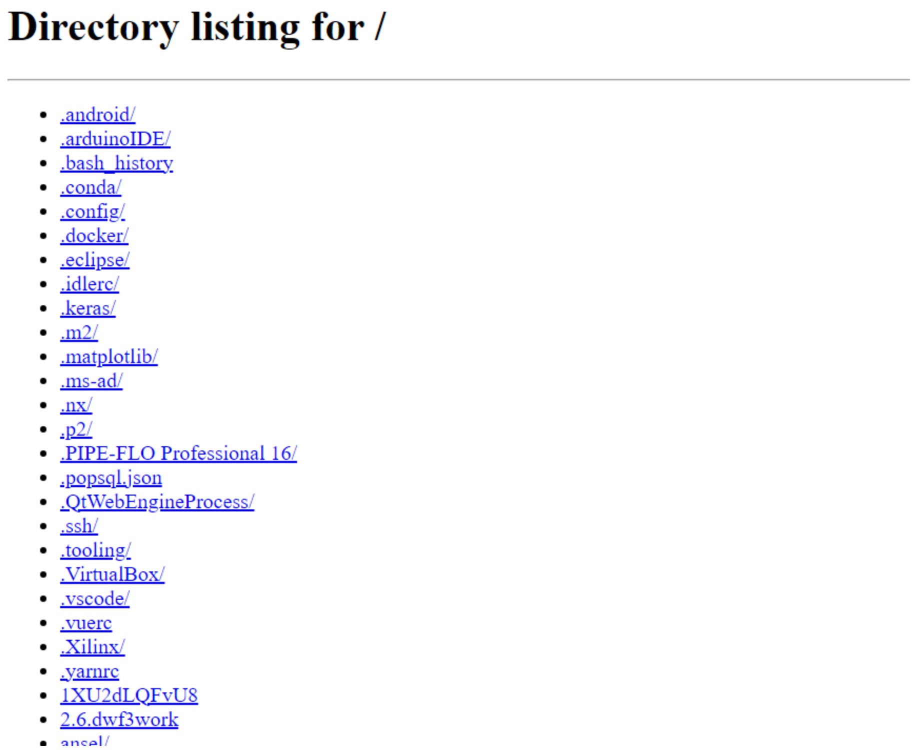
```
C:\Users\wangz>python -m http.server
Serving HTTP on :: port 8000 (http://[::]:8000/) ...
----------------------------------------
```



```
C:\Users\wangz>nmap -p 8000 -sT -Pn 192.168.40.83
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 18:24 Eastern Standard Time
Nmap scan report for host.docker.internal (192.168.40.83)
Host is up (0.0010s latency).

PORT     STATE SERVICE
8000/tcp open  http-alt

Nmap done: 1 IP address (1 host up) scanned in 0.28 seconds
```

Using python to create a web server that is listening on port 8000, at this time doing a standard scanning port on Nmap, it shows the Open state. This is due to it is available at this time. The open status indicates that the port is actively accepting connections, due to the HTTP server currently running and listening on this port.

On website make a HTTP request on port 8000, it showed the following content:



Directory listing for /

- .android/
- .arduinoIDE/
- .bash_history
- .conda/
- .config/
- .docker/
- .eclipse/
- .idlerc/
- .keras/
- .m2/
- .matplotlib/
- .ms-ad/
- .nx/
- .p2/
- .PIPE-FLO Professional 16/
- .popsql.json
- .QtWebEngineProcess/
- .ssh/
- .tooling/
- .VirtualBox/
- .vscode/
- .vuerc
- .Xilinx/
- .yarnrc
- 1XU2dLQFvU8
- 2.6.dwf3work
- ansel/

If I do not create python web server at port 8000, there will be no connection when I make the HTTP request. It usually display an error message such as ”This site can not be reached”. However, after I create the web server at port 8000, I saw the web browser display a list of files and directories. The list is clickable, allowing me to do some navigation.