

Wireshark: Inspecting UDP Packets

Rwithik Manoj, Roll No: 53

Aim

Using Wireshark observe three way handshaking connection establishment, data transfer and three way handshaking connection termination in client server communication using TCP.

Theory

TCP provides reliable communication using the concept called Positive Acknowledgement with Re-transmission(PAR). A device using PAR resends the data unit until it receives an acknowledgement. If the data unit received at the receiver's end is damaged, then the receiver discards the segment. So the sender has to resend the data unit for which positive acknowledgement is not received. From this, we can understand that three segments are exchanged between sender(client) and receiver(server) for a reliable TCP connection to get established. These three steps can be explained as follows:

- Step 1(SYN): Here, the client wants to establish a connection with server, so it sends a segment with SYN(Synchronize Sequence Number) which informs the server that the client is likely to start communication and with what sequence number it starts the segments with.
- Step 2(SYN + ACK): In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number, and the sequence number that the server chooses for the packet is another random number.
- Step 3(ACK): Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value, and the acknowledgement number is set to one more than the received sequence number.

And finally FIN is sent to terminate the connection.

Output

No.	Time	Source	Destination	Protocol	Length	Info
22	3.923724923	192.168.43.164	185.199.109.153	TLV1.2	638	Client Hello
31	4.009716220	185.199.109.153	192.168.43.164	TCP	64	443 → 43986 [ACK] Seq=1 Ack=565 Win=30200 Len=0 TSval=402372988 TSecr=2712340983
32	4.009716501	185.199.109.153	192.168.43.164	TLV1.2	216	Server Hello, Change Cipher Spec, Encrypted Handshake Message
33	4.009743220	192.168.43.164	185.199.109.153	TCP	66	43986 → 443 [ACK] Seq=565 Ack=151 Win=64128 Len=0 TSval=2712340989 TSecr=402372988
34	4.010502771	192.168.43.164	185.199.109.153	TLV1.2	117	Change Cipher Spec, Encrypted Handshake Message
35	4.017448476	192.168.43.164	185.199.109.153	TLV1.2	243	Application Data
41	4.094068404	185.199.109.153	192.168.43.164	TCP	66	443 → 43986 [ACK] Seq=151 Ack=793 Win=31232 Len=0 TSval=402373008 TSecr=2712340990
42	4.094068527	185.199.109.153	192.168.43.164	TLV1.2	132	Application Data
43	4.094087829	192.168.43.164	185.199.109.153	TCP	66	43986 → 443 [ACK] Seq=793 Ack=217 Win=64128 Len=0 TSval=2712341074 TSecr=402373008
44	4.094233590	192.168.43.164	185.199.109.153	TLV1.2	164	Application Data
46	4.145400908	192.168.43.164	185.199.111.153	TCP	74	51736 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3540981496 TSecr=0 WS=128
50	4.225494398	185.199.109.153	192.168.43.164	TCP	66	443 → 43986 [ACK] Seq=217 Ack=831 Win=31232 Len=0 TSval=402373041 TSecr=2712341074
51	4.225494568	185.199.111.153	192.168.43.164	TCP	74	443 → 51736 [SYN, ACK] Seq=0 Ack=1 Win=22000 Len=0 MSS=1378 SACK_PERM=1 TSval=3621913149 TSecr=3540981496
52	4.225534934	192.168.43.164	185.199.111.153	TCP	66	51736 → 443 [ACK] Seq=1 Ack=1 Win=64236 Len=0 TSval=3540981576 TSecr=3621913149
53	4.230460999	192.168.43.164	185.199.111.153	TLV1.2	625	Client Hello
59	4.305403941	192.168.43.164	82.206.193.4	TCP	74	50172 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2120139382 TSecr=0 WS=128
59	4.301279738	2499:4073:2084:dccc::	2686:4780:8d41:0:3c::	TCP	94	33158 → 443 [SYN] Seq=0 Win=64800 Len=0 MSS=1440 SACK_PERM=1 TSval=542227961 TSecr=0 WS=128
61	4.302418793	192.168.43.164	172.242.163.170	TCP	74	52940 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=56824993 TSecr=0 WS=128
63	4.326257268	185.199.111.153	192.168.43.164	TCP	66	443 → 51736 [ACK] Seq=1 Ack=560 Win=30208 Len=0 TSval=3621913177 TSecr=3540981581
64	4.326257435	185.199.111.153	192.168.43.164	TLV1.2	216	Server Hello, Change Cipher Spec, Encrypted Handshake Message

Figure 1: SYN, SYN/ACK and ACK packets

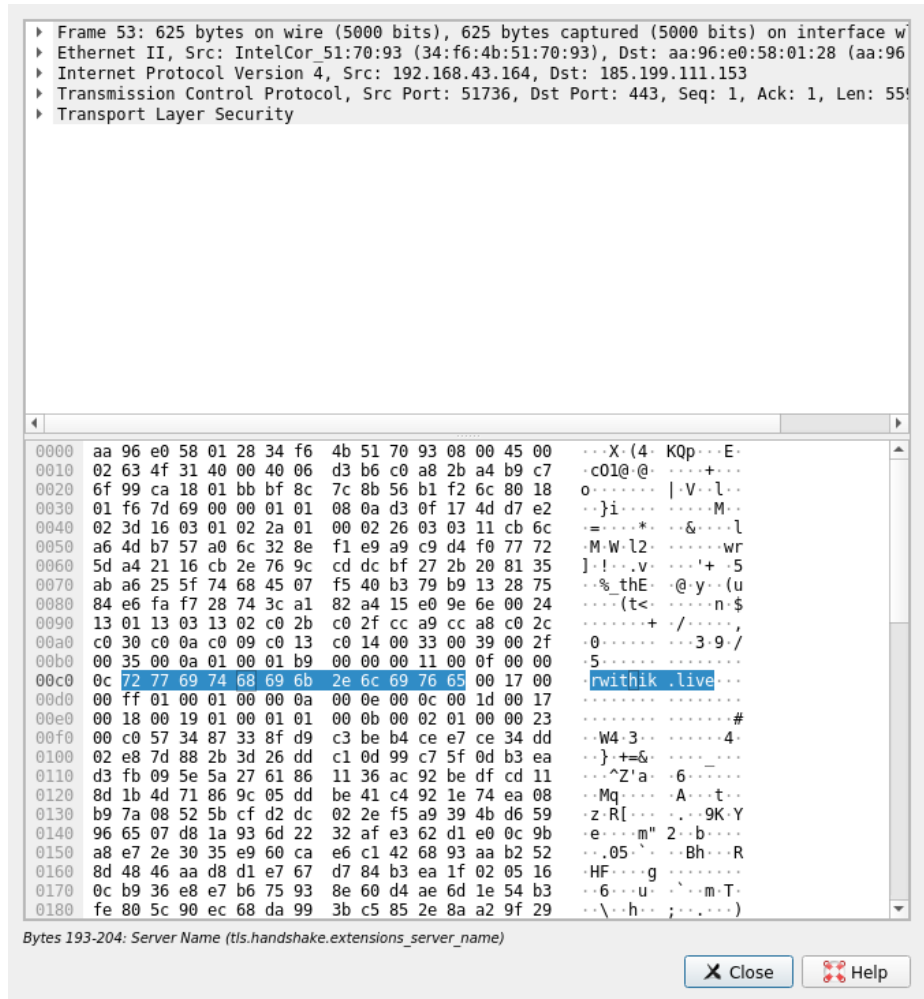


Figure 2: Data Transmitted by the packet

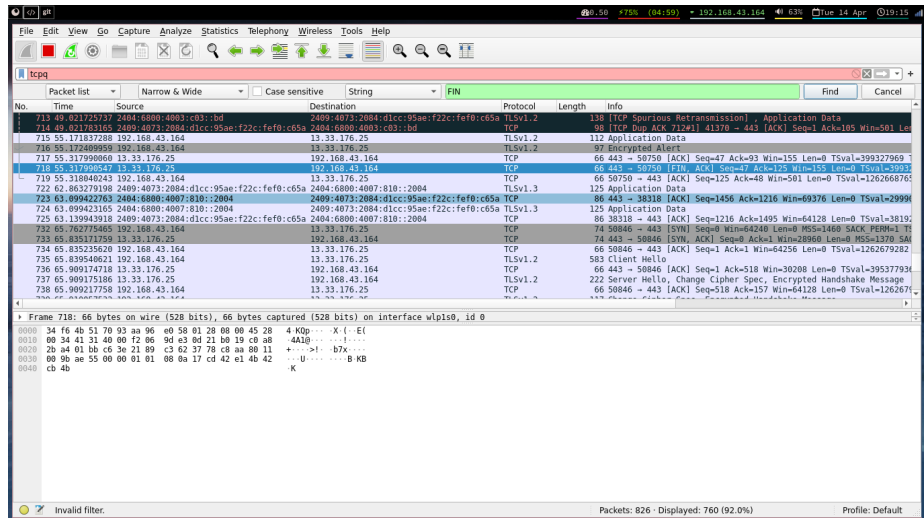


Figure 3: FIN packet