# TCP Packets - Baseline Analysis Exercise

**Q-1:** How many Total TCP packets for Window server?

**Answer:**

Total TCP Packets =16

3 for make a connection,6\*2 for send/receive messages,1 for disconnect.

**Q-2:** How many Total TCP packets for Linux server?

**Answer:**

Total TCP Packets =19

3 for make a connection,6\*2 for send/receive messages,4 for disconnect.

**Q-3:** Does the counting of total TCP packets differ for Windows and Linux server? If YES, explain why?

**Answer：**

**Although ctrl+C is used to end the process, Windows only uses one package to disconnect, but Linux has four packages to disconnect**

**Q-4:** What are the Minimum, Maximum and Common/Average Size (Length) of TCP packets for Windows server?

**Answer:**

Minimum Size of TCP Packets=54Bytes

Maximum Size of TCP Packets =66Bytes

Common Size of TCP Packets =58.5Bytes

**Q-5:** What are the Minimum, Maximum and Common/Average Size (Length) of TCP packets for Linux server?

**Answer:**

Minimum Size of TCP Packets=66Bytes

Maximum Size of TCP Packets =75Bytes

Common Size of TCP Packets =69.2857Bytes

**Q-6:** Did you find the TCP 3-Way Handshake packets for Windows/Linux server? If YES, Write the details of each packet: Set Flags, Sequence Number (Seq=), Acknowledgement Number (Ack=), and names of TCP Options fields.

**Answer:**

Name of Server (Window/Linux) =Linux

**1. Handshake Packet-1**

Set Flags=0x40[SYN]

Sequence Number (Seq) = 0 (relative sequence number)

Acknowledgement Number (Ack) = 0

TCP Options Size = (20 bytes)

TCP Options Field Names = tcp.options

**2. Handshake Packet-2**

Set Flags= 0x012 (SYN, ACK)

Sequence Number (Seq) = 0 (relative sequence number)

Acknowledgement Number (Ack) =1 (relative ack number)

TCP Options Size = 20Bytes

TCP Options Field Names =tcp.options

**3. Handshake Packet-3**

Set Flags=0x010 (ACK)

Sequence Number (Seq) =1 (relative sequence number)

Acknowledgement Number (Ack) = 1 (relative ack number)

TCP Options Size = 12Bytes

TCP Options Field Names =tcp.options

**Q-7:** Did you find the TCP Session Termination (FIN/RST) packets for Windows/Linux server? If YES, Write the details of each packet: Set Flags, Sequence Number (Seq=), Acknowledgement Number (Ack=), and names of TCP Options fields.

**Answer:**

Name of Server (Window/Linux) =Linux

**1. Termination Packet-1**

Set Flags=0x011 (FIN, ACK)

Sequence Number (Seq) =16 (relative sequence number)

Acknowledgement Number (Ack) =16 (relative ack number)

TCP Options Size = 12Bytes

TCP Options Field Names = tcp.options

**2. Termination Packet-2**

Set Flags= Flags: 0x010 (ACK)

Sequence Number (Seq) = Sequence Number: 16 (relative sequence number)

Acknowledgement Number (Ack) = Acknowledgment Number: 16 (relative ack number)

TCP Options Size = 12Bytes

TCP Options Field Names = tcp.options

**3.Termination Packet-3**

Set Flags=0x011 (FIN, ACK)

Sequence Number (Seq) =16 (relative sequence number)

Acknowledgement Number (Ack) = 17 (relative ack number)

TCP Options Size = 12Bytes

TCP Options Field Names =tcp.options

**4.Termination Packet-4**

Set Flags= 0x010 (ACK)

Sequence Number (Seq) =17 (relative sequence number)

Acknowledgement Number (Ack) =17 (relative sequence number)]

TCP Options Size = 12Bytes

TCP Options Field Names =tcp.options

Name of Server (Window/Linux) =Linux

**1. Termination Packet-1**

Set Flags=0x014 (RST, ACK)

Sequence Number (Seq) =16 (relative sequence number)

Acknowledgement Number (Ack) =16 (relative ack number)

TCP Options Size = 12Bytes

TCP Options Field Names = tcp.options

In Windows, I use CTRL+C to end communication, and only one packet is captured

**Q-8:** What other Flags/Combination of Flags (Set Flags) are found in collected TCP packets for Windows/Linux server? Explain about the combination of Flags [PSH, ACK].

**Answer:**

Set Flags for Windows/Linux server = Flags: 0x018 (PSH, ACK)

Explain about the combination of Flags [PSH, ACK]:

PSH: Tell the receiver to hand over the data to the upper layer immediately after receiving the data, rather than until the whole buffer is full.

ACK : The message segment includes a confirmation that the message segment has been successfully received.

**Q-9:** What are the different Sizes of TCP Options in all the TCP packets for Windows server? What is the common pattern (various fields) of TCP Options in most TCP packets and why?

**Answer:**

Different Sizes of TCP Options =0Bytes | 12Bytes

Common Size of TCP Options =0Bytes

Common Pattern of TCP Options =

Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted

**The 12 byte option field will be added only when the connection is established**

**Q-10:** What are the different Sizes of TCP Options in all the TCP packets for Linux server? What is the common pattern (various fields) of TCP Options in most TCP packets and why?

**Answer:**

Different Sizes of TCP Options =12Bytes | 20Bytes

Common Size of TCP Options = 12Bytes

Common Pattern of TCP Options =

Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps

Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale

**Only packets in the connection establishment process , the packet options size is 20.**

**Q-11:** Write the size of Data (in bytes) for all your client and server messages in the Windows server based experiment. Why these message packets are not continuous packets (unlike your continuous messages).

**Answer:**

1. Size of “Client says HELLO” Message =18
2. Size of “Server says HELLO” Message =18
3. Size of “Client TCP Packets” Message =19
4. Size of “Server TCP Packets” Message =19
5. Size of “Client says Finished” Message =21
6. Size of “Server says Finished” Message =21

Reason:

**After the sender sends the information, it will wait for the receiver to send an ack for confirmation**

**Q-12:** Write the size of Data (in bytes) for all your client and server messages in the Linux server based experiment. Why these message packets are not continuous packets (unlike your continuous messages).

**Answer:**

1. Size of “Client says HELLO” Message =18
2. Size of “Server says HELLO” Message =18
3. Size of “Client TCP Packets” Message =19
4. Size of “Server TCP Packets” Message =19
5. Size of “Client says Finished” Message =21
6. Size of “Server says Finished” Message =21

Reason:

After receiving the data packet, the receiver shall send a data packet for confirmation. Therefore, after sending each message packet, there will be a packet to confirm receipt.

**Q-13:** Did you find any retransmitted and/or duplicated (e.g. Dup, Ack) packets? If YES, explain briefly your analysis of these packets.

Total Retransmitted TCP Packets = 0

Total Packets with Duplicate Acknowledgement = 0

Analysis:not find

**Q-14:** Explain briefly about the source and destination port number for all your client and server message packets and analyse your result in the Windows server based experiment.

**Answer:**

1. Source and Destination Port Numbers of “Client says HELLO” Message =

Source Port: 33572 Destination Port: 1337

1. Source and Destination Port Numbers of “Server says HELLO” Message =

Source Port: 1337 Destination Port: 35572

1. Source and Destination Port Numbers of “Client TCP Packets” Message =

Source Port: 33572 Destination Port: 1337

1. Source and Destination Port Numbers of “Server TCP Packets” Message =

Source Port: 1337 Destination Port: 35572

1. Source and Destination Port Numbers of “Client says Finished” Message =

Source Port: 33572 Destination Port: 1337

1. Source and Destination Port Numbers of “Server says Finished” Message =

Source Port: 1337 Destination Port: 35572

Analysis:

**The server specifies the port 1337 that can be connected. The server will allocate a port to communicate with the 1337 port of the server**

**Q-15:** Explain briefly about the source and destination port number for all your client and server message packets and analyse your result in the Linux server based experiment.

**Answer:**

1. Source and Destination Port Numbers of “Client says HELLO” Message =

Source Port: 33900 Destination Port: 1337

1. Source and Destination Port Numbers of “Server says HELLO” Message =

Source Port: 1337 Destination Port: 33900

1. Source and Destination Port Numbers of “Client TCP Packets” Message =

Source Port: 33900 Destination Port: 1337

1. Source and Destination Port Numbers of “Server TCP Packets” Message =

Source Port: 1337 Destination Port: 33900

1. Source and Destination Port Numbers of “Client says Finished” Message =

Source Port: 33900 Destination Port: 1337

1. Source and Destination Port Numbers of “Server says Finished” Message =

Source Port: 1337 Destination Port: 33900

Analysis:

**The server specifies a port 1337 that can be connected. The server will allocate a port (33900 port selected in this communication) to communicate with the 1337 port of the server**