TCP 3-Way Handshake Process

Last Updated : 27 Dec, 2024

The TCP 3-Way Handshake is a fundamental process that establishes a reliable connection between two devices over a TCP/IP network. It involves three steps: SYN (Synchronize), SYN-ACK (Synchronize-Acknowledge), and ACK (Acknowledge). During the handshake, the client and server exchange initial sequence numbers and confirm the connection establishment. In this article, we will discuss the TCP 3-Way Handshake Process.

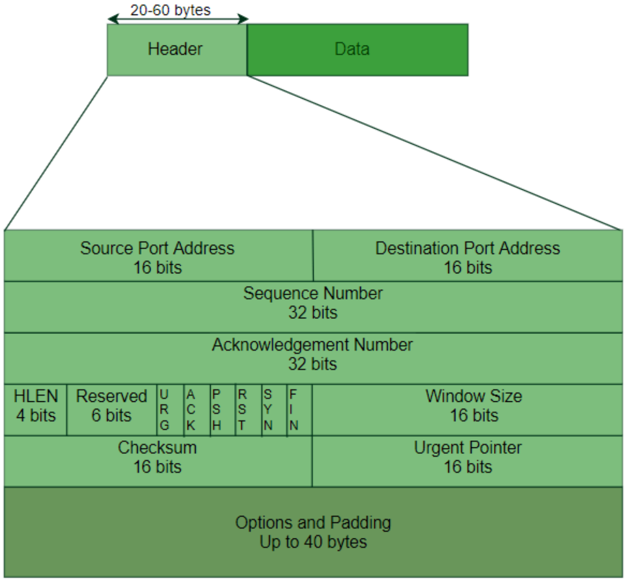
What is the TCP 3-Way Handshake?

The TCP 3-Way Handshake is a fundamental process used in the Transmission Control Protocol (TCP) to establish a reliable connection between a client and a server before data transmission begins. This handshake ensures that both parties are synchronized and ready for communication.

TCP Segment Structure

A TCP segment consists of data bytes to be sent and a header that is added to the data by TCP as shown:

can get a real feel of the exam. Plus, get our live mentorship classes with experts and attend **live doubt-solving sessions** to clear all your queries.



The header of a TCP segment can range from 20-60 bytes. 40 bytes are for options. If there are no options, a header is 20 bytes else it can be of upmost 60 bytes. Header fields:

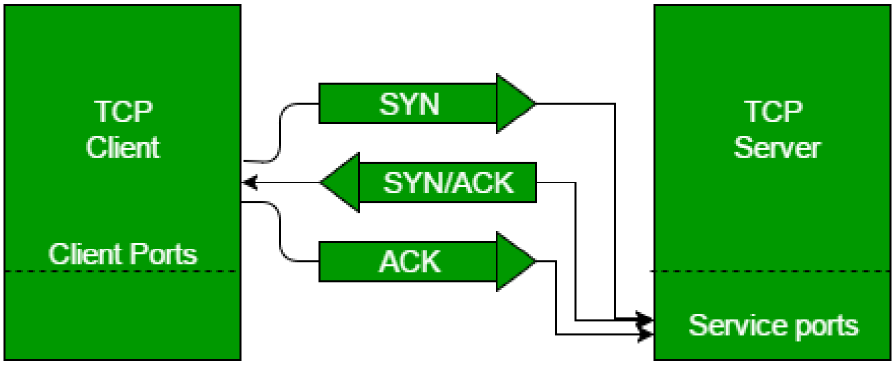
- **Source Port Address:** A 16-bit field that holds the port address of the application that is sending the data segment.
- **Destination Port Address:** A 16-bit field that holds the port address of the application in the host that is receiving the data segment.
- **Sequence Number:** A 32-bit field that holds the sequence number , i.e, the byte number of the first byte that is sent in that particular segment.

It is used to reassemble the message at the receiving end of the segments that are received out of order.

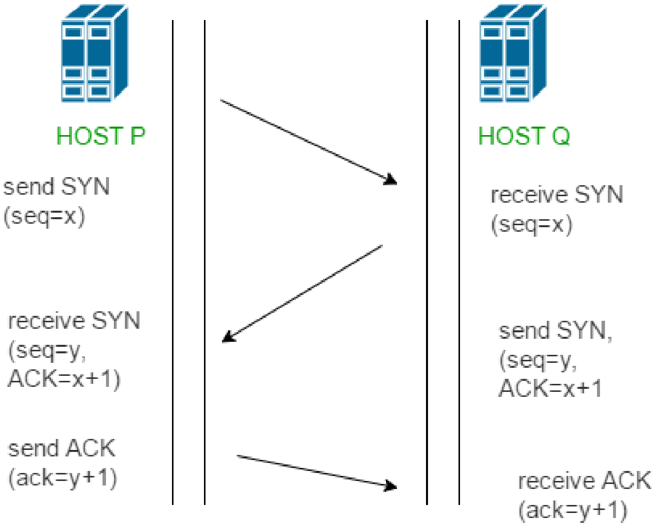
- **Acknowledgement Number:** A 32-bit field that holds the acknowledgement number, i.e, the byte number that the receiver expects to receive next. It is an acknowledgement for the previous bytes being received successfully.
- **Header Length (HLEN):** This is a 4-bit field that indicates the length of the TCP header by a number of 4-byte words in the header, i.e if the header is 20 bytes(min length of TCP header), then this field will hold 5 (because 5 x 4 = 20) and the maximum length: 60 bytes, then it'll hold the value 15(because 15 x 4 = 60). Hence, the value of this field is always between 5 and 15.
- **Control flags:** These are 6 1-bit control bits that control connection establishment, connection termination, connection abortion, flow control, mode of transfer etc. Their function is:
  - URG: Urgent pointer is valid
  - ACK: Acknowledgement number is valid(used in case of cumulative acknowledgement)
  - PSH: Request for push
  - RST: Reset the connection
  - SYN: Synchronize sequence numbers
  - FIN: Terminate the connection
- **Window size:** This field tells the window size of the sending TCP in bytes.
- **Checksum:** This field holds the checksum for error control . It is mandatory in TCP as opposed to UDP.
- **Urgent pointer:** This field (valid only if the URG control flag is set) is used to point to data that is urgently required that needs to reach the receiving process at the earliest. The value of this field is added to the sequence number to get the byte number of the last urgent byte.

TCP 3-way Handshake Process

The process of communication between devices over the internet happens according to the current **TCP/IP** suite model(stripped-out version of OSI reference model). The Application layer is a top pile of a stack of TCP/IP models from where network-referenced applications like web browsers on the client side establish a connection with the server. From the application layer, the information is transferred to the transport layer where our topic comes into the picture. The two important protocols of this layer are – TCP, and **UDP(User Datagram Protocol)** out of which TCP is prevalent(since it provides reliability for the connection established). However, you can find an application of UDP in querying the DNS server to get the binary equivalent of the Domain Name used for the website.



TCP provides reliable communication with something called **Positive Acknowledgement with Re-transmission(PAR)** . The Protocol Data Unit(PDU) of the transport layer is called a segment. Now a device using PAR resend the data unit until it receives an acknowledgement. If the data unit received at the receiver's end is damaged(It checks the data with checksum functionality of the transport layer that is used for Error Detection), the receiver discards the segment. So the sender has to resend the data unit for which positive acknowledgement is not received. You can realize from the above mechanism that three segments are exchanged between sender(client) and receiver(server) for a reliable TCP connection to get established. Let us delve into how this mechanism works



- **Step 1 (SYN):** In the first step, the client wants to establish a connection with a 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 segments with
- **Step 2 (SYN + ACK):** Server responds to the client request with SYN-ACK signal bits set. Acknowledgement(ACK) signifies the response of the segment it received and SYN signifies with what sequence number it is likely to start the segments with
- **Step 3 (ACK):** In the final part client acknowledges the response of the server and they both establish a reliable connection with which they will start the actual data transfer

Question For Practice

Question: Consider a TCP client and a TCP server running on two different machines. After completing the data transfer, the TCP client calls **close** to terminate the connection and a FIN segment is sent to the TCP server. Server-side TCP responds by sending an ACK which is received by the client-side TCP. As per the TCP connection state diagram(RFC 793), in which state does the client-side TCP connection wait for the FIN from the server-side TCP? **[GATE-CS-2017 (Set 1)]**

(A) LAST-ACK

(B) TIME-WAIT

(C) FIN-WAIT-1

(D) FIN-WAIT-2

Explanation : *(D) For detail solution visit the article. GATE PYQs*

Conclusion

The TCP 3-Way Handshake is a critical mechanism for establishing a secure connection between a client and a server over a TCP/IP network. It consists of three important steps: the client initiates the connection by sending an SYN packet, the server responds with a SYN-ACK message to acknowledge the client's request and synchronize sequence numbers, and the client sends an ACK packet to complete the connection. This handshake ensures that both sides are in sync and prepared for dependable data transmission, making it an essential mechanism for stable and secure communication in TCP/IP networks.

Frequently Asked Questions on TCP 3-Way Handshake Process – FAQs

What is the purpose of the SYN flag in the TCP three-way handshake?

The SYN (Synchronize Sequence Number) flag is used in the initial step of the handshake. It informs the server that the client wants to establish a connection and specifies the sequence number for subsequent segments ·

Why is a three-way handshake necessary in TCP/IP networks?