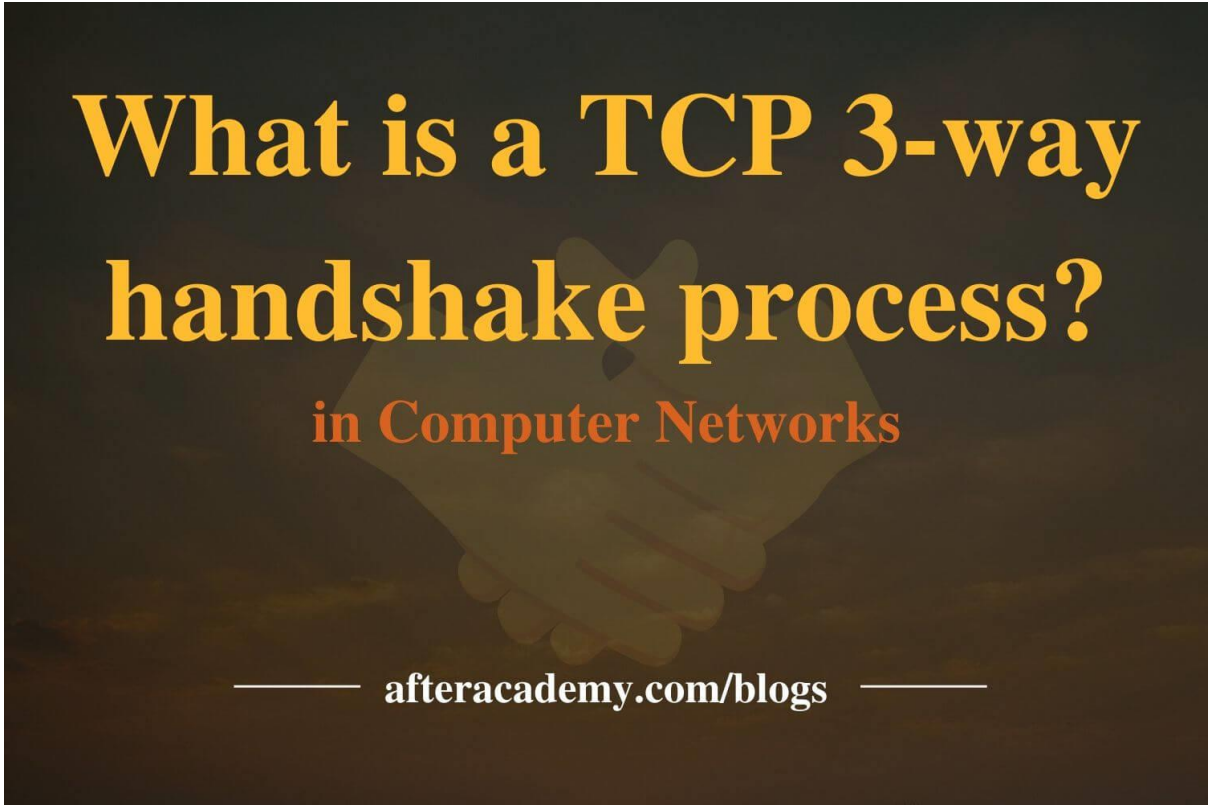


What is a TCP 3-way handshake process?

What is a TCP 3-way handshake process? in Computer Networks



———— afteracademy.com/blogs ————

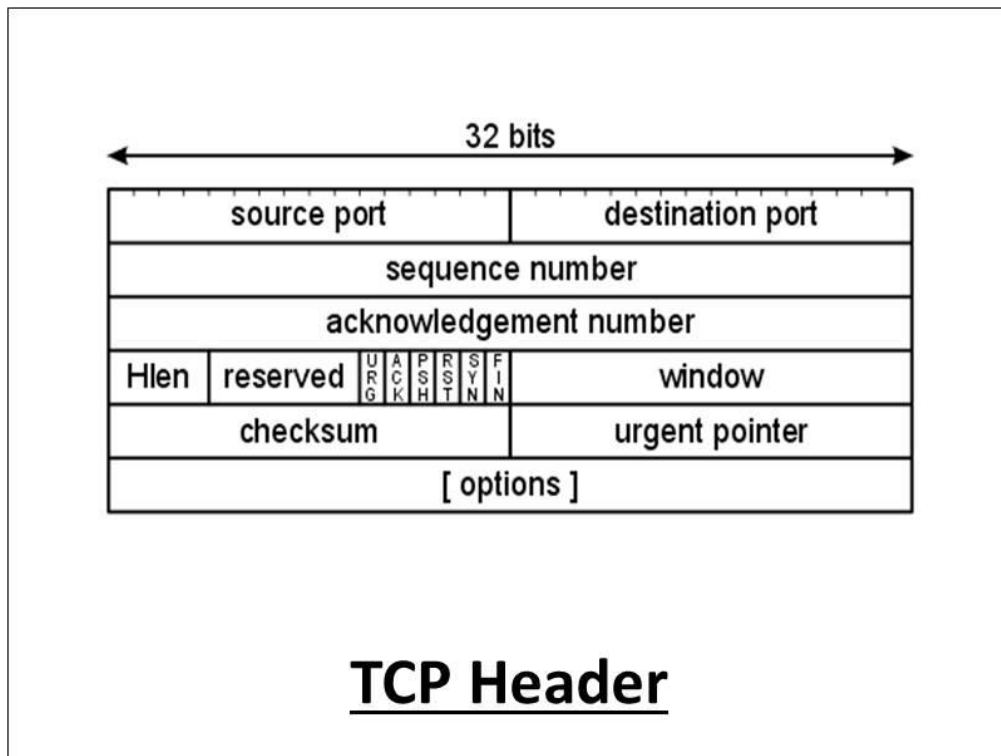
Nowadays we mainly use TCP(Transfer Control Protocol) for data transmission in a connection-oriented network. But have you ever wondered, why we prefer TCP over any other protocol for this purpose?

Actually, TCP provides us with a secure and reliable connection link between two devices. And, this is possible only due to the 3-way handshake process that takes place in the TCP during establishing and closing connections between two devices. As the name suggests, there are three steps for both establishing and closing the connection. So in this blog, we'll learn about the TCP 3-way handshake process and the different steps involved in it.

TCP 3-Way Handshake Process

The 3-Way Handshake process is the defined set of steps that takes place in the TCP for creating a secure and reliable communication link and also closing it. Actually, TCP uses the 3-way handshake process to establish a connection between two devices before transmitting the data. After the establishment of the connection, the data transfer takes place between the devices. After which the connection needs to be terminated, which is also done by using the 3-way handshake process. The secure and reliable connection is established to reserve the CPU, buffer, and bandwidth of the devices to communicate properly. Thus, it is a must to free these resources by terminating the connection after data transmission. Hence, the TCP 3-way handshake process can be used to establish and terminate connections in the network in a secure way.

Below is the pictorial representation of the TCP header.

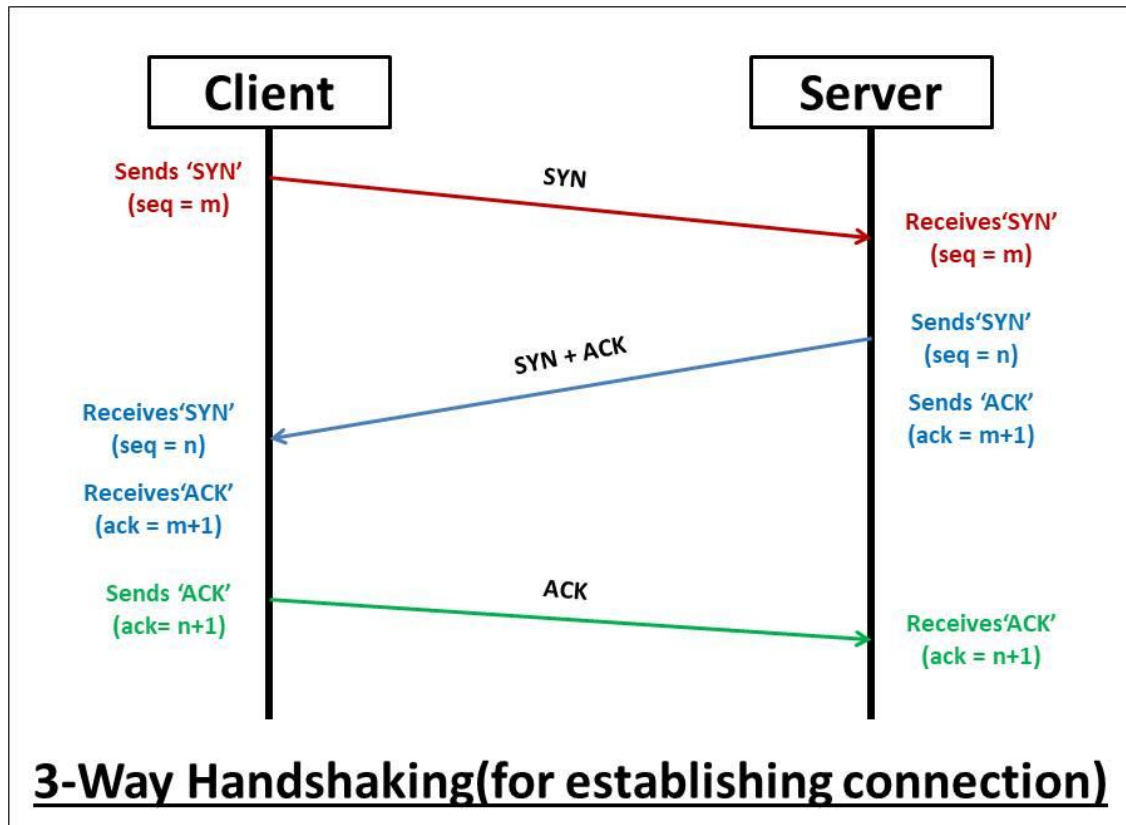


There are a few elements in the TCP header file which are used in the 3-way handshake process, they are:

1. **Sequence Number:** Sequence number is a random 32 bits(in the range of 0 to $(2^{32} - 1)$) number which is assigned to the first bit of the data. Generally, a sequence number is used only once in one connection. For other data transmission in the same connection, some other random sequence number can be used.
2. **Acknowledgement Number:** It is the next sequence number that the acknowledgement sending device expects from the sender. It is generally, 1 greater than the sequence number received from the sender.
3. **Window Size:** Window size is the buffer size. It is the capacity up to which data can be received in the buffer.

4. **Maximum Segment Size:** It is the maximum acceptable size of each data segment by the connected device. Above this size, the device will not be able to receive the data segments.
5. **SYN Flag:** SYN stands for synchronization. It can be described as a request for establishing a connection. If SYN is 1, it means that the device wants to establish a secure connection, else not.
6. **ACK Flag:** ACK stands for acknowledgement. It can be described as the response of SYN. If ACK is 1, the device has received the SYN message and acknowledges it, else not.
7. **FIN Flag:** FIN stands for Finished. After the data transmission has been completed, devices have to terminate the connection using the FIN flag. If FIN is 1, the device wants to terminate the connection, else not.

Below is the pictorial representation of the connection establishment using the 3-way handshake process.



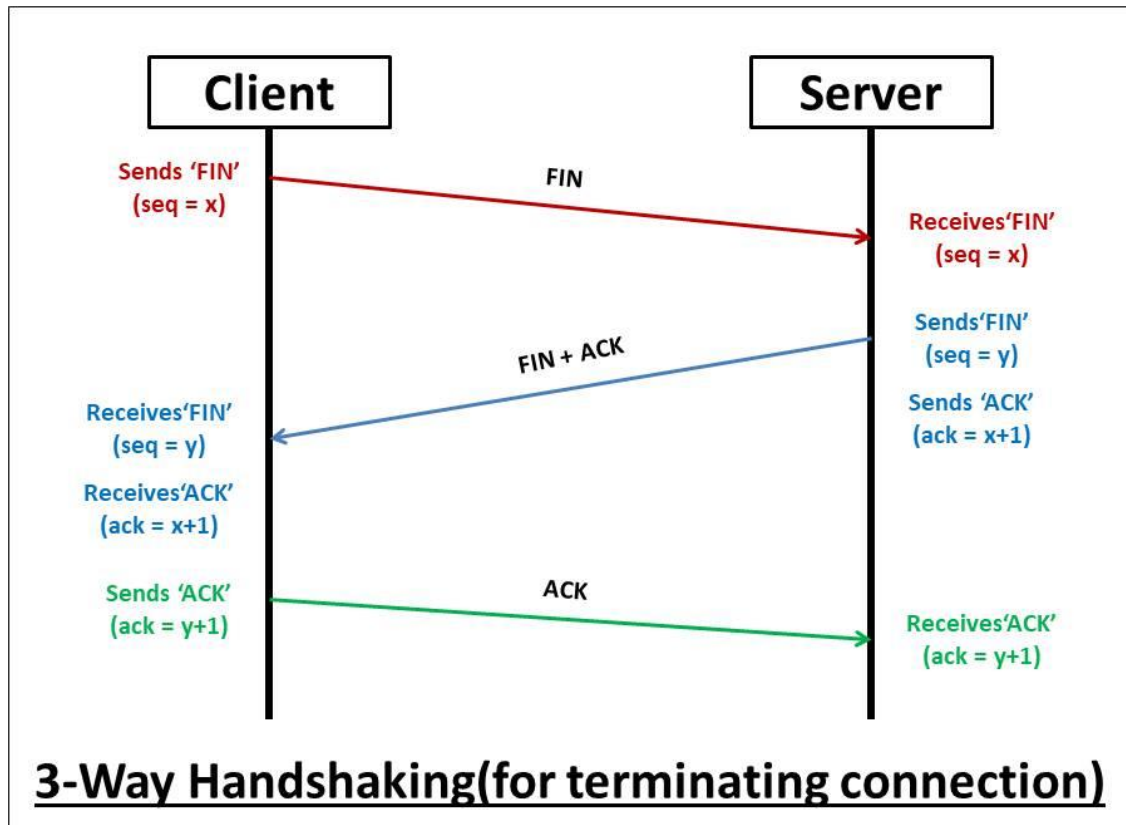
Following are the three steps involved in establishing the connection using the 3-way handshake process in TCP:

- 1. The client sends the SYN to the server:** When the client wants to connect to the server. It sets the 'SYN' flag as 1 and sends the message to the server. The message has also some additional information like the sequence number(any random 32 bits number), the ACK is set here to 0, the window size, and the maximum segment size. For Example, if the window size is 2000 bits, and the maximum segment size is 200 bits then a maximum of 10 data segments ($2000/200 = 10$) can be transmitted in the connection.
- 2. The server replies with the SYN and the ACK to the client:** After receiving the client's synchronization request, the server sends an acknowledge to the client by setting the ACK flag to '1'. The acknowledgement number of the ACK is one more than the received sequence number. For Example,

if the client has sent the SYN with sequence number = 1000, then the server will send the ACK with acknowledgement number = 10001. Also, the server sets the SYN flag to '1' and sends it to the client, if the server also wants to establish the connection. The sequence number used here for the SYN will be different from the client's SYN. The server also advertises its window size and maximum segment size to the client. After completion of this step, the connection is established from the client to the server-side.

3. **The client sends the ACK to the server:** After receiving the SYN from the server, the client sets the ACK flag to '1' and sends it with an acknowledgement number 1 greater than the server's SYN sequence number to the client. Here, the SYN flag is kept '0'. After completion of this step, the connection is now established from the server to the client-side also. After the connection is being established, the minimum of the sender's and receiver's maximum segment size is taken under consideration for data transmission.

Below is the pictorial representation of the connection termination using the 3-way handshake process.



Following are the three steps involved in terminating the connection using the 3-way handshake process in TCP:

1. **The client sends the FIN to the server:** When the client wants to terminate the connection. It sets the FIN flag as '1' and sends the message to the server with a random sequence number. Here, the ACK is set to 0.
2. **The server replies with the FIN and the ACK to the client:** After receiving the client's termination request, the server sends an acknowledge to the client by setting the ACK flag to '1'. The acknowledgement number of the ACK is one more than the received sequence number. For Example, if the client has sent the FIN with sequence number = 1000, then the server will send the ACK with acknowledgement number = 10001. Also, the server sets the FIN flag to '1' and sends it to the client, if the server also wants to terminate the connection. The sequence number used here for the FIN

will be different from the client's FIN. After completion of this step, the connection is terminated from the client to the server-side.

3. **The client sends the ACK to the server:** After receiving the FIN from the server, the client sets the ACK flag to '1' and sends it with an acknowledgement number 1 greater than the server's FIN sequence number to the client. Here, the FIN flag is kept '0'. After completion of this step, the connection is now terminated from the server to the client-side also.

This is all about the 3-way handshake process and its steps that are used in the TCP protocol. Hope you learned something new today.