TCP Segment Header

We'll now study TCP headers. They're are far more complex than UDP headers and really are what allow for TCP to work properly!

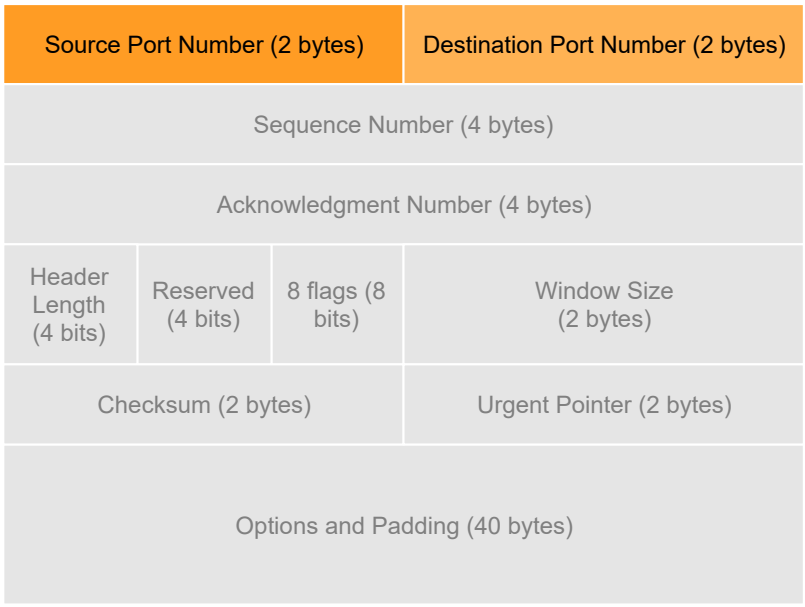
WE'LL COVER THE FOLLOWING

Introduction

TCP headers play a crucial role in the implementation of the protocol. In fact, TCP segments without actual data and with headers are completely valid. They’re actually used quite often!

The size of the headers range from **20 - 60 bytes**. Let’s discuss the header field by field.

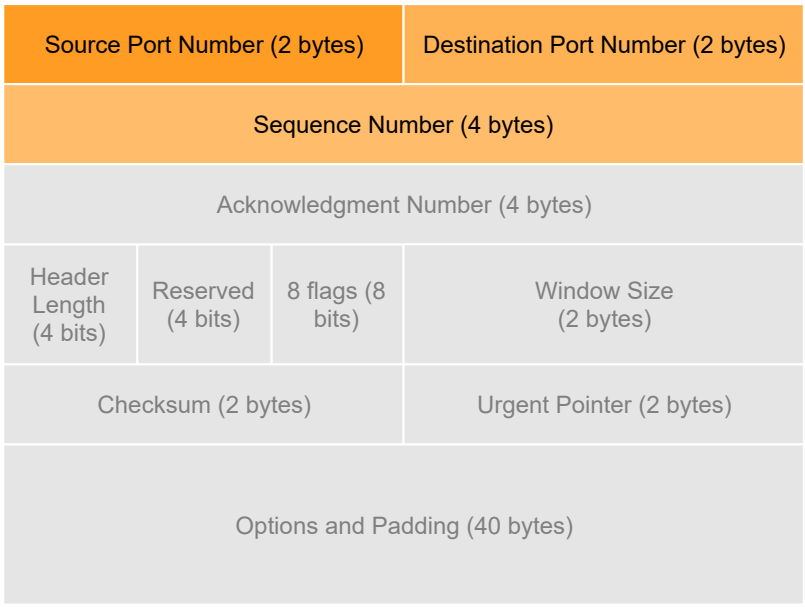
Source and Destination Ports



The source and destination ports are the first fields of the TCP header.

The source and destination port numbers are self-explanatory. They are exactly like the source and destination ports in UDP. Just for a refresher though, the source port is the port of the socket of the application that is **sending** the segment and the **destination port** is the port of the socket of the **receiving** application. The size of each field is **two bytes**.

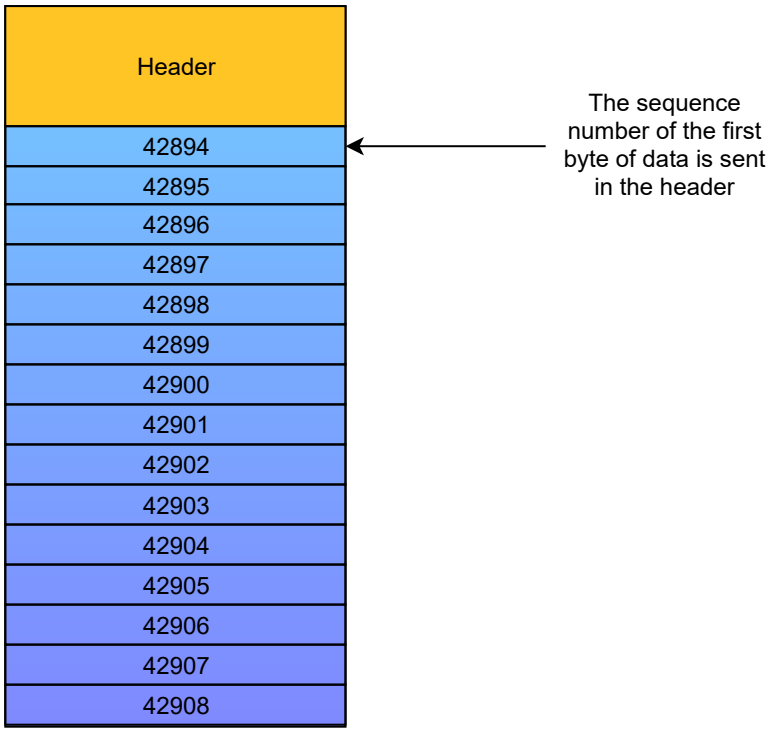
Sequence Number



The sequence number is the second field of the TCP header. It represents the first byte of data in the TCP segment.

Every byte of the TCP segment's data is labeled with a number called a **sequence number**. The sequence number field in the header has the

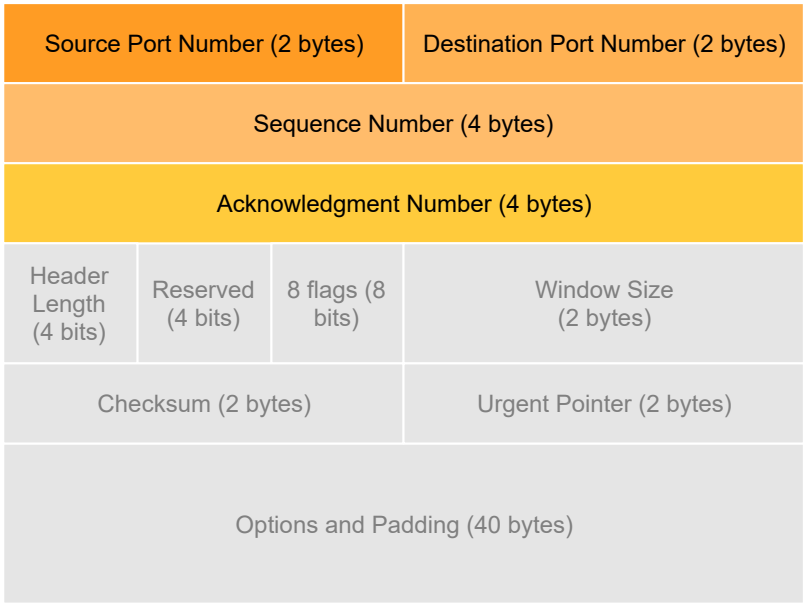
sequence number of the first byte of data in the segment.



Each byte of a TCP segment is labeled with a sequence number.

> **Note** The initial sequence number is a randomly generated number between 0 and $2^{32} - 1$.

Acknowledgement Number



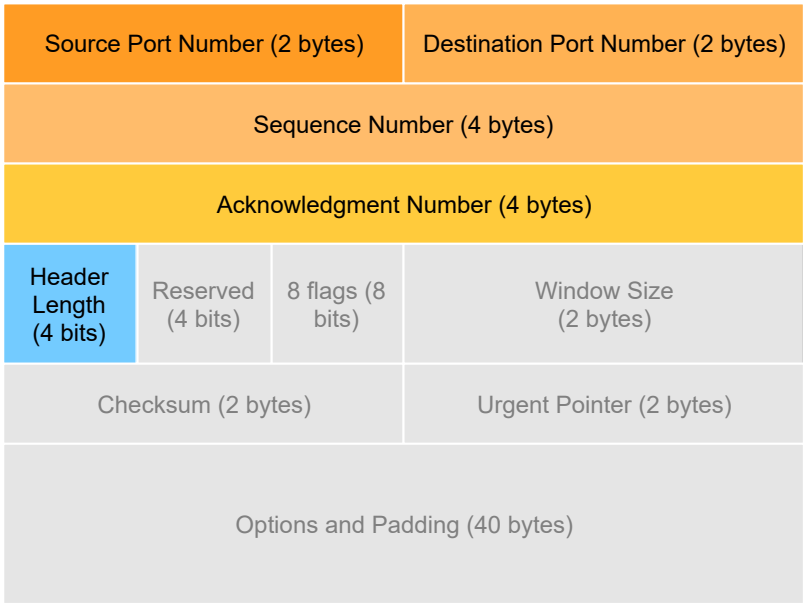
The third field in the TCP header is the acknlowledgement number

The **acknowledgment number** is a 4-byte field that represents the sequence number of the next expected segment that the sender will receive.

Example

So if a segment's sequence number was 42849 and its data field had 59 bytes of data, the sequence number of the next expected segment or the **acknowledgment number** would be 42908. This helps TCP to identify if a segment was missing or out of order.

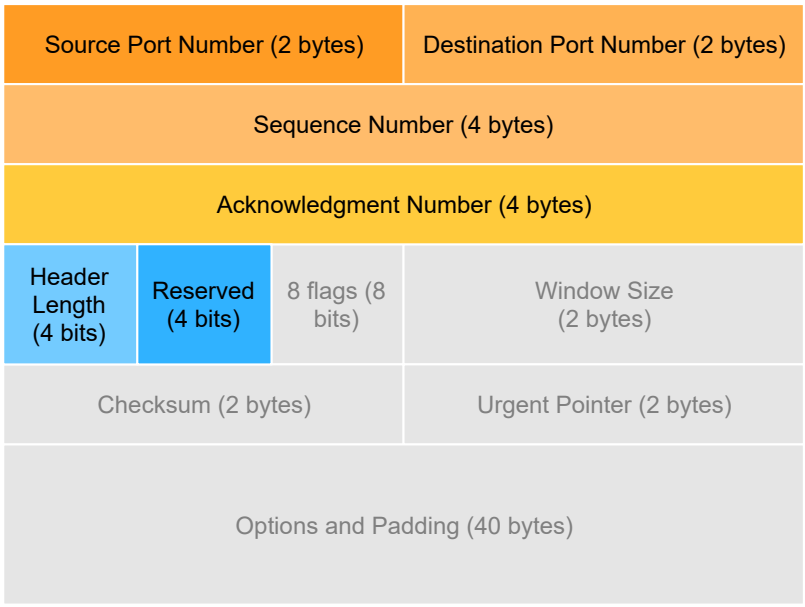
Header Length



The length of the TCP header is specified here. This helps the receiving end to identify where the header ends and the data starts from.

> **Note** The header length is represented by **4 bits**, i.e., the numbers $0000 \rightarrow 1111$ or $0 \rightarrow 15$ in decimal which is not enough to represent the potential **60 bytes** of the header. Hence, this number is **multiplied by 4** upon receiving. So 1111 would represent 60. In other words, the way the 4-bit header length field is used to represent a maximum header length of 60, is that this field represents the number of 4-byte words in the header

Reserved Field



The reserved bits serve as an offset and are left for potential future use

The header has a 4-bit field that is reserved and is always set to 0. This field aligns the total header size to be in multiples of 4 (as we saw was necessary for the header length to be processed).

Quick Quiz!

1 Given an initial sequence number of 255 and 50 bytes sent in a TCP segment, what will be the value of the sequence number field in the next TCP packet header?

We’ll continue dissecting the TCP headers in the next lesson!