Contents

The TCP/IP Guide
9 TCP/IP Lower-Layer (Interface, Internet and Transport) Protocols (OSI Layers 2, 3 and 4)
9 TCP/IP Transport Layer Protocols
9 Transmission Control Protocol (TCP) and User Datagram Protocol (UDP)
9 TCP/IP Transmission Control Protocol (TCP)
9 TCP Message Formatting and Data Transfer

Searchable, convenient, complete TCP/IP information. The TCP/IP Guide

Prev. Topic TCP Message Formatting and Data Transfer

Pages Prev. Page 1 2 3 4 Next Page TCP Checksum Calculation and the TCP "Pseudo H

TCP Message (Segment) Format

TCP Segment Format

The format for TCP messages (segments) is described fully in Table 156 and Figure 216.

Table 156: TCP Segment Format

Field Name	Size (bytes)	Description
Source Port	2	***Source Port:*** The 16-bit port number of the process that originated the TCP segment on the source This will normally be an ephemeral (client) port number for a request sent by a client to a server, or known/registered (server) port number for a reply from a server to a client.
Destination Port	2	***Destination Port:*** The 16-bit port number of the process that is the ultimate intended recipient of th on the destination device. This will usually be a well-known/registered (server) port number for a cli or an ephemeral (client) port number for a server reply.
Sequence Number	4	***Sequence Number:*** For normal transmissions, the sequence number of the first byte of data in this In a connection request (*SYN*) message, this carries the initial sequence number (ISN) of the sourc first byte of data will be given the next sequence number after the contents of this field, as describe topic on sequence number synchronization.
Acknowledgment Number	4	***Acknowledgment Number:*** When the *ACK* bit is set, this segment is serving as an acknowledgme addition to other possible duties) and this field contains the sequence number the source is next ex destination to send. See the topic describing TCP data transfer for details.
Data Offset	1/2 (4 bits)	***Data Offset:*** Specifies the number of 32-bit words of data in the TCP header. In other words, this v four equals the number of bytes in the header, which must always be a multiple of four. It is called a offset" since it indicates by how many 32-bit words the start of the data is offset from the beginning segment.
Reserved	3/4 (6 bits)	***Reserved:*** 6 bits reserved for future use; sent as zero.

<table>
<tr><td>Control Bits</td><td>3/4
(6 bits)</td><td>Control Bits: As mentioned, TCP does not use a separate format for control messages. Instead, certain bits are set to indicate the communication of control information. The six bits are:
<table>
<tr><th>Subfield Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>URG</td><td>1/8
(1 bit)</td><td>Urgent Bit: When set to 1, indicates that the priority data transfer feature has been invoked for this segment, and that the Urgent Pointer field is valid.</td></tr>
<tr><td>ACK</td><td>1/8
(1 bit)</td><td>Acknowledgment Bit: When set to 1, indicates that this segment is carrying an acknowledgment, and the value of the Acknowledgment Number field is valid and carrying the next sequence expected from the destination of this segment.</td></tr>
<tr><td>PSH</td><td>1/8
(1 bit)</td><td>Push Bit: The sender of this segment is using the TCP push feature, requesting that the data in this segment be immediately pushed to the application on the receiving device.</td></tr>
<tr><td>RST</td><td>1/8
(1 bit)</td><td>Reset Bit: The sender has encountered a problem and wants to reset the connection.</td></tr>
<tr><td>SYN</td><td>1/8
(1bit)</td><td>Synchronize Bit: This segment is a request to synchronize sequence numbers and establish a connection; the Sequence Number field contains the initial sequence number (ISN) of the sender of the segment.</td></tr>
<tr><td>FIN</td><td>1/8
(1 bit)</td><td>Finish Bit: The sender of the segment is requesting that the connection be closed.</td></tr>
</table></td></tr>
<tr><td>Window</td><td>2</td><td>Window: Indicates the number of octets of data the sender of this segment is willing to accept from receiver at one time. This normally corresponds to the current size of the buffer allocated to accept connection. This field is, in other words, the current receive window size for the device sending this which is also the send window for the recipient of the segment. See the data transfer mechanics to details.</td></tr>
<tr><td>Checksum</td><td>2</td><td>Checksum: A 16-bit checksum for data integrity protection, computed over the entire TCP datagra special “pseudo header” of fields. It is used to protect the entire TCP segment against not just error transmission, but also errors in delivery. Optional alternate checksum methods are also supported.</td></tr>
<tr><td>Urgent Pointer</td><td>2</td><td>Urgent Pointer: Used in conjunction with the URG control bit for priority data transfer. This field co sequence number of the last byte of urgent data. See the priority data transfer topic for details.</td></tr>
<tr><td>Options</td><td>Variable</td><td>Options: TCP includes a generic mechanism for including one or more sets of optional data in a TCP segment. Each of the options can be either one byte in length or variable in length. The first byte is the Option-Kind subfield, and its value specifies the type of option, which in turn indicates whether the option is just a single byte or multiple bytes. Options that are many bytes consist of three fields:
<table>
<tr><th>Subfield Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>Option-Kind</td><td>1</td><td>Option-Kind: Specifies the option type.</td></tr>
<tr><td>Option-Length</td><td>1</td><td>Option-Length: The length of the entire option in bytes, including the Option-Kind and Option-Length fields.</td></tr>
<tr><td>Option-Data</td><td>Variable</td><td>Option-Data: The option data itself. In at least one oddball case, this field is omitted (making Option-Length equal to 2).</td></tr>
</table></td></tr>
<tr><td>Padding</td><td>Variable</td><td>Padding: If the Options field is not a multiple of 32 bits in length, enough zeroes are added to pad so it is a multiple of 32 bits.</td></tr>
<tr><td>Data</td><td>Variable</td><td>Data: The bytes of data being sent in the segment.</td></tr>
</table>

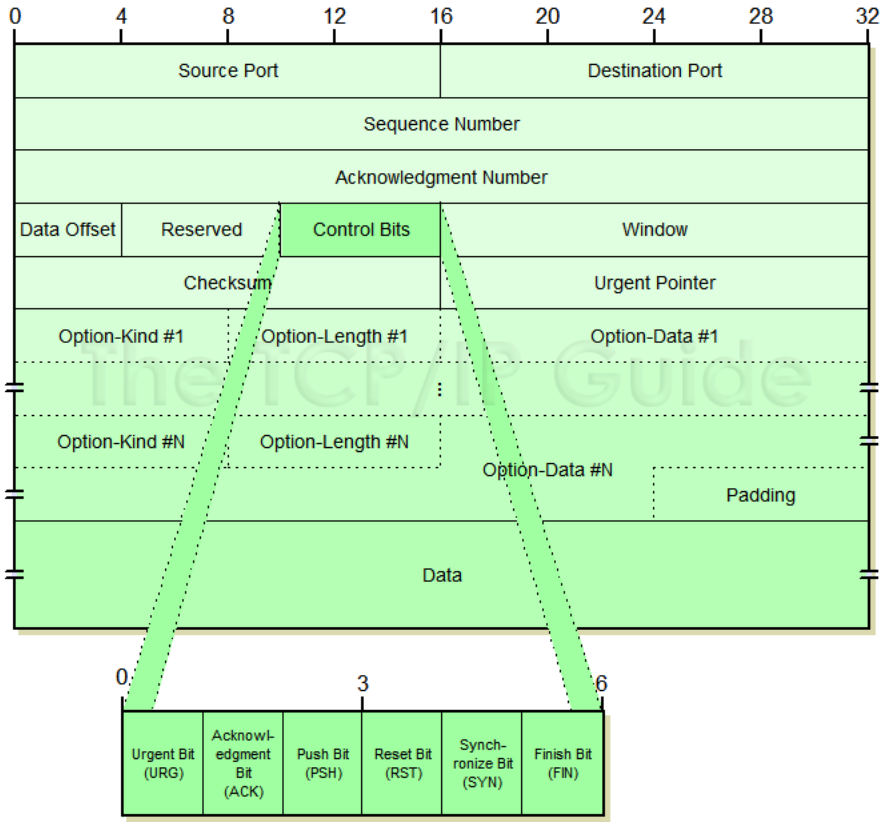

Figure 216: TCP Segment Format

Prev. Topic TCP Message Formatting and Data Transfer

Pages Prev. Page 1 2 3 4 Next Page TCP Checksum Calculation and the TCP "Pseudo H