TCP, Transmission Control Protocol

RFC Sourcebook | Description | Glossary | RFCs | Publications | Obsolete RFCs

Description:

Protocol suite: TCP/IP.
Protocol type: Transport layer connection oriented byte stream protocol.
IP *Protocol:* 6.
MIME *subtype:*
SNMP MIBs: iso.org.dod.internet.experimental.ipv6TcpMIB (1.3.6.1.3.86).
iso.org.dod.internet.mgmt.mib-2.tcp (1.3.6.1.2.1.6).
iso.org.dod.internet.mgmt.mib-2.tcpMIB (1.3.6.1.2.1.49).
Working groups: pilc, Performance Implications of Link Characteristics.
tcpimpl, TCP Implementation.
tcpm, TCP Maintenance and Minor Extensions.
tsvwg, Transport Area Working Group.
IANA: TCP option numbers.
Links:

TCP is a transport layer protocol used by applications that require guaranteed delivery. It is a sliding window protocol that provides handling for both timeouts and retransmissions.

TCP establishes a full duplex virtual connection between two endpoints. Each endpoint is defined by an IP address and a TCP port number. The operation of TCP is implemented as a finite state machine.

The byte stream is transfered in segments. The window size determines the number of bytes of data that can be sent before an acknowledgement from the receiver is necessary.

MAC header	IP header	TCP header	Data :::

TCP header:

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="16">Source Port</td><td colspan="16">Destination Port</td></tr>
<tr><td colspan="32">Sequence Number</td></tr>
<tr><td colspan="32">Acknowledgment Number</td></tr>
<tr><td colspan="4">Data Offset</td><td colspan="3">reserved</td><td colspan="3">ECN</td><td colspan="6">Control Bits</td><td colspan="16">Window</td></tr>
<tr><td colspan="16">Checksum</td><td colspan="16">Urgent Pointer</td></tr>
<tr><td colspan="32">Options and padding :::</td></tr>
<tr><td colspan="32">Data :::</td></tr>
</table>

Source Port. 16 bits.

Destination Port. 16 bits.

Sequence Number. 32 bits.
The sequence number of the first data byte in this segment. If the SYN bit is set, the sequence number is the initial sequence number and the first data byte is initial sequence number + 1.

Acknowledgment Number. 32 bits.
If the ACK bit is set, this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established this is always sent.

Data Offset. 4 bits.
The number of 32-bit words in the TCP header. This indicates where the data begins. The length of the TCP header is always a multiple of 32 bits.

reserved. 3 bits.
Must be cleared to zero.

ECN, Explicit Congestion Notification. 3 bits.
Added in RFC 3168.

00	01	02
N	C	E

N, NS, Nonce Sum. 1 bit.
Added in RFC 3540. This is an optional field added to ECN intended to protect against accidental or malicious concealment of marked packets from the TCP sender.

C, CWR. 1 bit.

E, ECE, ECN-Echo. 1 bit.

Control Bits. 6 bits.

00	01	02	03	04	05
U	A	P	R	S	F

U, URG. 1 bit.
Urgent pointer valid flag.

A, ACK. 1 bit.
Acknowledgment number valid flag.

P, PSH. 1 bit.
Push flag.

R, RST. 1 bit.
Reset connection flag.

S, SYN. 1 bit.
Synchronize sequence numbers flag.

F, FIN. 1 bit.
End of data flag.

Window. 16 bits, unsigned.

The number of data bytes beginning with the one indicated in the acknowledgment field which the sender of this segment is willing to accept.

Checksum. 16 bits.
This is computed as the 16-bit one's complement of the one's complement sum of a pseudo header of information from the IP header, the TCP header, and the data, padded as needed with zero bytes at the end to make a multiple of two bytes. The pseudo header contains the following fields:

<table>
<tr><th>00</th><th>01</th><th>02</th><th>03</th><th>04</th><th>05</th><th>06</th><th>07</th><th>08</th><th>09</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>20</th><th>21</th><th>22</th><th>23</th><th>24</th><th>25</th><th>26</th><th>27</th><th>28</th><th>29</th><th>30</th><th>31</th></tr>
<tr><td colspan="32">Source IP address</td></tr>
<tr><td colspan="32">Destination IP address</td></tr>
<tr><td colspan="8">0</td><td colspan="8">IP Protocol</td><td colspan="16">Total length</td></tr>
</table>

Urgent Pointer. 16 bits, unsigned.
If the *URG* bit is set, this field points to the sequence number of the last byte in a sequence of urgent data.

Options. 0 to 40 bytes.
Options occupy space at the end of the TCP header. All options are included in the checksum. An option may begin on any byte boundary. The TCP header must be padded with zeros to make the header length a multiple of 32 bits.

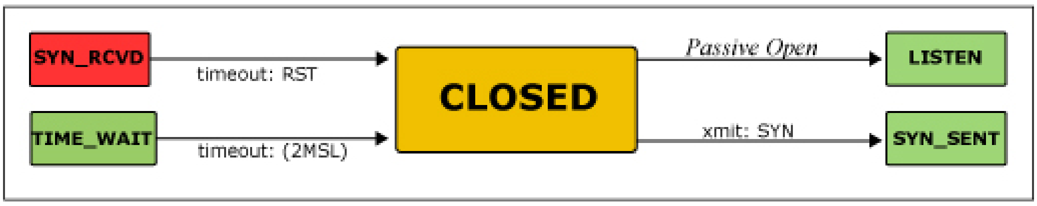
Kind	Length	Description	References
0	1	End of option list.	RFC 793
1	1	No operation.	RFC 793
2	4	MSS, Maximum Segment Size.	RFC 793
3	3	WSOPT, Window scale factor.	RFC 1323
4	2	SACK permitted.	RFC 2018
5	Variable.	SACK.	RFC 2018, RFC 2883
6	6	Echo. (obsolete).	RFC 1072
7	6	Echo reply. (obsolete).	RFC 1072
8	10	TSOPT, Timestamp.	RFC 1323
9	2	Partial Order Connection permitted.	RFC 1693
10	3	Partial Order service profile.	RFC 1693
11	6	CC, Connection Count.	RFC 1644
12	6	CC.NEW	RFC 1644
13	6	CC.ECHO	RFC 1644
14	3	Alternate checksum request.	RFC 1146
15	Variable.	Alternate checksum data.	RFC 1146
16		Skeeter.	
17		Bubba.	
18	3	Trailer Checksum Option.	
19	18	MD5 signature.	RFC 2385
20		SCPS Capabilities.	
21		Selective Negative Acknowledgements.	
22		Record Boundaries.	
23		Corruption experienced.	
24		SNAP.	

25			
26		TCP Compression Filter.	
27	8	Quick-Start Response.	RFC 4782
28	4	User Timeout.	RFC 5482
29		TCP-AO, TCP Authentication Option.	RFC 5925
30		MPTCP	
31 - 252			
253		RFC3692-style Experiment 1.	RFC 4727
254		RFC3692-style Experiment 2.	RFC 4727
255			

Data. Variable length.

TCP State machine:

State	Description
CLOSE-WAIT	Waits for a connection termination request from the remote host.
CLOSED	Represents no connection state at all.
CLOSING	Waits for a connection termination request acknowledgment from the remote host.
ESTABLISHED	Represents an open connection, data received can be delivered to the user. The normal state for the data transfer phase of the connection.
FIN-WAIT-1	Waits for a connection termination request from the remote host or an acknowledgment of the connection termination request previously sent.
FIN-WAIT-2	Waits for a connection termination request from the remote host.
LAST-ACK	Waits for an acknowledgment of the connection termination request previously sent to the remote host (which includes an acknowledgment of its connection termination request).
LISTEN	Waits for a connection request from any remote TCP and port.
SYN-RECEIVED	Waits for a confirming connection request acknowledgment after having both received and sent a connection request.
SYN-SENT	Waits for a matching connection request after having sent a connection request.
TIME-WAIT	Waits for enough time to pass to be sure the remote host received the acknowledgment of its connection termination request.



The CLOSED state is the entry point to the TCP state machine.

Glossary:

ABC, Appropriate Byte Counting.
Congestion control algorithm. A modification to the algorithm for increasing TCP's congestion window (cwnd) that

improves both performance and security. Rather than increasing a TCP's congestion window based on the number of acknowledgments (ACKs) that arrive at the data sender, the congestion window is increased based on the number of bytes acknowledged by the arriving ACKs. The algorithm improves performance by mitigating the impact of delayed ACKs on the growth of cwnd. At the same time, the algorithm provides cwnd growth in direct relation to the probed capacity of a network path, therefore providing a more measured response to ACKs that cover only small amounts of data (less than a full segment size) than ACK counting. This more appropriate cwnd growth can improve both performance and can prevent inappropriate cwnd growth in response to a misbehaving receiver. On the other hand, in some cases the modified cwnd growth algorithm causes larger bursts of segments to be sent into the network. In some cases this can lead to a non-negligible increase in the drop rate and reduced performance.

active open.

AIMD, Additive Increase, Multiplicative Decrease.
Congestion control algorithm. (RFC 2914) In the absence of congestion, the TCP sender increases its congestion window by at most one packet per roundtrip time. In response to a congestion indication, the TCP sender decreases its congestion window by half. More precisely, the new congestion window is half of the minimum of the congestion window and the receiver's advertised window.

Congestion Avoidance.
Congestion control algorithm.

Connection.
A logical communication path identified by a pair of endpoints.

cwnd, congestion window.
TCP state variable. This variable limits the amount of data a TCP can send. At any given time, a TCP MUST NOT send data with a sequence number higher than the sum of the highest acknowledged sequence number and the minimum of *cwnd* and *rwnd*.

TCP uses two algorithms for increasing the congestion window. During steady-state, TCP uses the Congestion Avoidance algorithm to linearly increase the value of cwnd. At the beginning of a transfer, after a retransmission timeout or after a long idle period (in some implementations), TCP uses the Slow Start algorithm to increase cwnd exponentially. Slow Start bases the cwnd increase on the number of incoming acknowledgments. During congestion avoidance RFC 2581 allows more latitude in increasing cwnd, but traditionally implementations have based the increase on the number of arriving ACKs.

CWV, Congestion Window Validation. Algorithm.
This algorithm limits the amount of unused *cwnd* a TCP connection can accumulate. *ABC* can be used in conjunction with *CWV* to obtain an accurate measure of the network path.

Eifel. Algorithm.
(RFC 3522) This algorithm allows a TCP sender to detect a posteriori whether it has entered loss recovery unnecessarily. It requires that the TCP Timestamp option is enabled for a connection. Eifel makes use of the fact that the TCP Timestamp option eliminates the retransmission ambiguity in TCP. Based on the timestamp of the first acceptable ACK that arrives during loss recovery, it decides whether loss recovery was entered unnecessarily. The Eifel detection algorithm provides a basis for future TCP enhancements. This includes response algorithms to back out of loss recovery by restoring a TCP sender's congestion control state.

Fast Recovery. Congestion control algorithm.
A sender invokes the Fast Recovery after Fast Retransmit. This algorithm allows the sender to transmit at half its previous rate (regulating the growth of its window based on congestion avoidance), rather than having to begin a Slow Start. This also saves time.

Fast Retransmit. Congestion control algorithm.

(RFC 2757) When a TCP sender receives several duplicate ACKs, fast retransmit allows it to infer that a segment was lost. The sender retransmits what it considers to be this lost segment without waiting for the full timeout, thus saving time.

flight size.
The amount of data that has been sent but not yet acknowledged.

full sized segment.
A segment that contains the maximum number of data bytes permitted.

IW, Initial Window.
The size of the sender's congestion window after the three-way handshake is completed.

LFN, Long Fat Network.
A communications path with a large bandwidth * delay product.

LW, Loss Window.
The size of the congestion window after a TCP sender detects loss using its retransmission timer.

MSL, Maximum Segment Lifetime.
The maximum time in seconds that a segment may be held before being discarded.

MSS, Maximum Segment Size.
When IPv4 is used as the network protocol, the MSS is calculated as the maximum size of an IPv4 datagram minus 40 bytes.

When IPv6 is used as the network protcol, the MSS is calculated as the maximum packet size minus 60 bytes. An MSS of 65535 should be interpreted as infinity.

passive open.

PAWS, Protect Against Wrapped Sequences.
A mechanism to reject old duplicate segments that might corrupt an open TCP connection. PAWS uses the same TCP timestamp option as the RTTM mechanism and assumes that every received TCP segment (including data and ACK segments) contains a timestamp whose values are monotone non-decreasing in time. The basic idea is that a segment can be discarded as an old duplicate if it is received with a timestamp less than some timestamp recently received on this connection.

RMSS, Receiver Maximum Segment Size.
The size of the largest segment the receiver is willing to accept. This is the value specified in the MSS option sent by the receiver during connection startup. Or, if the MSS option is not used, 536 bytes. The size does not include the TCP headers and options.

RTT, Round trip time.

RTTM, Round-Trip Time Measurement.
A technique for measuring the RTT by use of timestamps. The data segments are timestamped using the TSOPT option. The resulting ACK packets contain timestamps from the receiver. The resulting RTT can then be determined by the difference in the timestamps.

RW, Restart Window.
The size of the congestion window after a TCP restarts transmission after an idle period.

rwmd, Receiver Window. TCP state variable.
The most recently advertised receiver window.

SACK, Selective Acknowledgement. Algorithm.
This technique allows the data receiver to inform the sender about all segments that have arrived successfully, so the sender need retransmit only the segments that have actually been lost. This extension uses two TCP options. The first is an enabling option, SACK permitted, which may be sent in a SYN segment to indicate that the SACK option can be used once the connection is established. The other is the SACK option itself, which may be sent over an established connection once permission has been given.

segment.
A TCP data or acknowledgment packet.

Slow Start. Congestion control algorithm.
This algorithm is used to gradually increase the size of the TCP congestion window. It operates by observing that the rate at which new packets should be injected into the network is the rate at which the acknowledgments are returned by the other end.

SMSS, Sender Maximum Segment Size.
The size of the largest segment that the sender can transmit. This value can be based on the maximum transmission unit of the network, the path MTU discovery algorithm, RMSS, or other factors. The size does not include the TCP headers and options.

SWS, Silly Window Syndrome.

TFRC, TCP Friendly Rate Control. Algorithm.
A congestion control mechanism for unicast flows operating in a best effort Internet environment. It is reasonably fair when competing for bandwidth with TCP flows, but has a much lower variation of throughput over time compared with TCP, making it more suitable for applications such as telephony or streaming media where a relatively smooth sending rate is of importance. TFRC is designed for applications that use a fixed packet size and vary their sending rate in packets per second in response to congestion.

Van Jacobson's algorithm.

RFCs:

[IEN 2] Comments on Internet Protocol and TCP.

[IEN 12] Issues in Reliable Host-to-Host Protocols.

[IEN 45] TCP Checksum Function Design.

[IEN 74] Sequence Number Arithmetic.

[IEN 92] Protocol Options.

[IEN 98] TCP Implementation Status.

[IEN 114] PROTOCOL OPTIONS.

[IEN 150] TCP JSYS CALLING SEQUENCES.

[IEN 167] HP3000 TCP DESIGN DOCUMENT.

[RFC 721] Out-of-Band Control Signals in a Host-to-Host Protocol.

[RFC 761] DOD STANDARD TRANSMISSION CONTROL PROTOCOL.

- Obsoletes:
  IEN 5, IEN 21, IEN 27, IEN 40, IEN 44, IEN 55, IEN 81, IEN 112, IEN 124.

[RFC 793] Transmission Control Protocol.

- STD: 7.
- Defines the TCP protocol.
- Updated by:
  RFC 6528.

[RFC 801] NCP/TCP TRANSITION PLAN.

[RFC 813] WINDOW AND ACKNOWLEDGEMENT STRATEGY IN TCP.

[RFC 816] FAULT ISOLATION AND RECOVERY.

[RFC 832] Who Talks TCP?

[RFC 833] Who Talks TCP?

[RFC 834] Who Talks TCP?

[RFC 835] Who Talks TCP?

[RFC 836] Who Talks TCP?

[RFC 837] Who Talks TCP?

[RFC 838] Who Talks TCP?

[RFC 839] Who Talks TCP?

[RFC 842] Who Talks TCP? - Survey of 1 February 83.

[RFC 843] Who Talks TCP? -- Survey of 8 February 1983.

[RFC 845] Who Talks TCP? -- Survey of 15 February 1983.

[RFC 846] Who Talks TCP? -- Survey of 22 February 1983.

[RFC 872] TCP-ON-A-LAN.

[RFC 879] The TCP Maximum Segment Size and Related Topics.

[RFC 889] Internet Delay Experiments.

[RFC 896] Congestion Control in IP/TCP Internetworks.

[RFC 939] Executive Summary of the NRC Report on Transport Protocols for Department of Defense Data Networks.

[RFC 942] TRANSPORT PROTOCOLS FOR DEPARTMENT OF DEFENSE DATA NETWORKS.

[RFC 962] TCP-4 Prime.

[RFC 964] SOME PROBLEMS WITH THE SPECIFICATION OF THE MILITARY STANDARD TRANSMISSION CONTROL PROTOCOL.

[RFC 1025] TCP AND IP BAKE OFF.

[RFC 1106] TCP Big Window and Nak Options.

[RFC 1110] A Problem with the TCP Big Window Option.

[RFC 1122] Requirements for Internet Hosts -- Communication Layers.

- STD: 3.
- Updated by:
  RFC 1349, RFC 4379.

[RFC 1144] Compressing TCP/IP Headers for Low-Speed Serial Links.

[RFC 1146] TCP Alternate Checksum Options.

- Defines TCP option 14 (Alternate checksum request), 15 (Alternate checksum data).
- Obsoletes:
  RFC 1145.

[RFC 1156] Management Information Base for Network Management of TCP/IP-based internets.

- Obsoletes:
  RFC 1066.

[RFC 1180] A TCP/IP Tutorial.

[RFC 1191] Path MTU Discovery.

- Obsoletes:
  RFC 1063.

[RFC 1263] TCP EXTENSIONS CONSIDERED HARMFUL.

[RFC 1323] TCP Extensions for High Performance.

- Defines TCP options 3 (Window Scale) and 8 (Timestamp).
- Obsoletes:
  RFC 1072, RFC 1185.

[RFC 1337] TIME-WAIT Assassination Hazards in TCP.

[RFC 1347] TCP and UDP with Bigger Addresses (TUBA), A Simple Proposal for Internet Addressing and Routing.

[RFC 1379] Extending TCP for Transactions -- Concepts.

[RFC 1475] TP/IX: The Next Internet.

- Defines ICMP message 31 (Conversion error).
- Defines IP version 7.

[RFC 1644] T/TCP -- TCP Extensions for Transactions Functional Specification.

- Category: Experimental.
- Defines TCP options 11 (CC), 12 (CC.NEW), 13 (CC.ECHO).

[RFC 1693] An Extension to TCP : Partial Order Service.

- Category: Experimental.
- Defines TCP options 9 (POC permitted) and 10 (POC service profile).

[RFC 1705] Six Virtual Inches to the Left: The Problem with IPng.

- Category: Informational.

[RFC 1791] TCP And UDP Over IPX Networks With Fixed Path MTU.

- Category: Experimental.

[RFC 1812] Requirements for IP Version 4 Routers.

- Category: Standards Track.
- Obsoletes:
  RFC 1009, RFC 1716.

[RFC 1858] Security Considerations for IP Fragment Filtering.

- Category: Informational.

[RFC 1859] ISO Transport Class 2 Non-use of Explicit Flow Control over TCP RFC1006 extension.

- Category: Informational.

[RFC 1981] Path MTU Discovery for IP version 6.

- Category: Standards Track.

[RFC 2018] TCP Selective Acknowledgment Options.

- Category: Standards Track
- Defines TCP options 4 (SACK permitted) and 5 (SACK).

[RFC 2126] ISO Transport Service on top of TCP (ITOT).

- Category: Standards Track.

[RFC 2140] TCP Control Block Interdependence.

- Category: Informational.

[RFC 2385] Protection of BGP Sessions via the TCP MD5 Signature Option.

- Category: Standards Track.
- Defines TCP option 19 (MD5 Signature).

[RFC 2415] Simulation Studies of Increased Initial TCP Window Size.

- Catetgory: Informational.

[RFC 2416] When TCP Starts Up With Four Packets Into Only Three Buffers.

- Category: Informational.

[RFC 2460] Internet Protocol, Version 6 (IPv6) Specification.

- Category: Standards Track.
- Obsoletes:
  RFC 1883.

[RFC 2488] Enhancing TCP Over Satellite Channels using Standard Mechanisms.

- BCP: 28.

[RFC 2507] IP Header Compression.

- Category: Standards Track.

[RFC 2525] Known TCP Implementation Problems.

- Category: Informational.

[RFC 2581] TCP Congestion Control.

- Category: Standards Track.
- Obsoletes:
  RFC 2001.

[RFC 2675] IPv6 Jumbograms.

- Category: Standards Track.
- Obsoletes:
  RFC 2147.

[RFC 2757] Long Thin Networks.

- Category: Informational.

[RFC 2760] Ongoing TCP Research Related to Satellites.

- Category: Informational.

[RFC 2780] IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers.

- BCP: 37.
- Updated by:
  RFC 4443, RFC 5237.

[RFC 2861] TCP Congestion Window Validation.

- Category: Experimental.
- Defines the TCP congestion control algorithm CWV.

[RFC 2873] TCP Processing of the IPv4 Precedence Field.

- Category: Standards Track.

[RFC 2883] An Extension to the Selective Acknowledgement (SACK) Option for TCP.

- Category: Standards Track.

[RFC 2884] Performance Evaluation of Explicit Congestion Notification (ECN) in IP Networks.

- Category: Informational.

[RFC 2914] Congestion Control Principles.

- BCP: 41.

[RFC 2923] TCP Problems with Path MTU Discovery.

- Category: Informational.

[RFC 2988] Computing TCP's Retransmission Timer.

- Category: Standards Track.

[RFC 2990] Next Steps for the IP QoS Architecture.

- Category: Informational.

[RFC 3042] Enhancing TCP's Loss Recovery Using Limited Transmit.

- Category: Standards Track.

[RFC 3081] Mapping the BEEP Core onto TCP.

- Category: Standards Track.

[RFC 3128] Protection Against a Variant of the Tiny Fragment Attack.

- Category: Informational.
- Updates:
  RFC 1858.

[RFC 3135] Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations.

- BCP: 50.

[RFC 3148] A Framework for Defining Empirical Bulk Transfer Capacity Metrics.

- Category: Informational.

[RFC 3150] End-to-end Performance Implications of Slow Links.

- BCP: 48.

[RFC 3155] End-to-end Performance Implications of Links with Errors.

- BCP: 50.

[RFC 3168] The Addition of Explicit Congestion Notification (ECN) to IP.

- Category: Standards Track.
- Updated by:
  RFC 6040.
- Obsoletes:
  RFC 2481.

- Updates:
  RFC 793, RFC 2401, RFC 2474.

[RFC 3360] Inappropriate TCP Resets Considered Harmful.

- BCP: 60.

[RFC 3390] Increasing TCP's Initial Window.

- Category: Standards Track.
- Obsoletes:
  RFC 2414.
- Updates:
  RFC 2581.

[RFC 3430] Simple Network Management Protocol (SNMP) over Transmission Control Protocol (TCP) Transport Mapping.

- Category: Experimental.
- Defines the use of TCP as a transport protocol for SNMP.

[RFC 3449] TCP Performance Implications of Network Path Asymmetry.

- BCP: 69.

[RFC 3465] TCP Congestion Control with Appropriate Byte Counting (ABC).

- Category: Experimental.
- Defines the TCP congestion control algorithm ABC.

[RFC 3481] TCP over Second (2.5G) and Third (3G) Generation Wireless Networks.

- BCP: 71.

[RFC 3517] A Conservative Selective Acknowledgment (SACK)-based Loss Recovery Algorithm for TCP.

- Category: Standards Track.

[RFC 3522] The Eifel Detection Algorithm for TCP.

- Category: Experimental.

[RFC 3540] Robust Explicit Congestion Notification (ECN) Signaling with Nonces.

- Category: Experimental.
- Defines ECN bit NS (Nonce Sum).

[RFC 3562] Key Management Considerations for the TCP MD5 Signature Option.

- Category: Informational.

[RFC 3649] HighSpeed TCP for Large Congestion Windows.

- Category: Experimental.

[RFC 3708] Using TCP Duplicate Selective Acknowledgement (DSACKs) and Stream Control Transmission

Protocol (SCTP) Duplicate Transmission Sequence Numbers (TSNs) to Detect Spurious Retransmissions.

- Category: Experimental.

[**RFC 3742**] Limited Slow-Start for TCP with Large Congestion Windows.

- Category: Experimental.

[**RFC 4015**] The Eifel Response Algorithm for TCP.

- Category: Standards Track.

[**RFC 4022**] Management Information Base for the Transmission Control Protocol (TCP).

- Category: Standards Track.
- Defines SNMP MIB iso.org.dod.internet.mgmt.mib-2.tcp (1.3.6.1.2.1.6).
- Defines SNMP MIB iso.org.dod.internet.mgmt.mib-2.tcpMIB (1.3.6.1.2.1.49).
- Obsoletes:
  RFC 2012, RFC 2452.

[**RFC 4138**] Forward RTO-Recovery (F-RTO): An Algorithm for Detecting Spurious Retransmission Timeouts with TCP and the Stream Control Transmission Protocol (SCTP).

- Category: Experimental.

[**RFC 4278**] Standards Maturity Variance Regarding the TCP MD5 Signature Option (RFC 2385) and the BGP-4 Specification.

- Category: Informational.

[**RFC 4413**] TCP/IP Field Behavior.

- Category: Informational.

[**RFC 5348**] TCP Friendly Rate Control (TFRC): Protocol Specification.

- Defines the TFRC algorithm..
- Obsoletes:
  RFC 3448.
- Updates:
  RFC 4342.

[**RFC 5382**] NAT Behavioral Requirements for TCP.

- BCP: 142.

[**RFC 5461**] TCP's Reaction to Soft Errors.

- Category: Informational.

[**RFC 6069**] Making TCP More Robust to Long Connectivity Disruptions (TCP-LCD).

- Category: Experimental.

[**RFC 6191**] Reducing the TIME-WAIT State Using TCP Timestamps.

- BCP: 159.

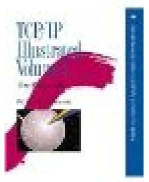
[RFC 6528] Defending against Sequence Number Attacks.

- Category: Standards Track.
- Specifies an algorithm for the generation of TCP Initial Sequence Numbers.
- Obsoletes:
  RFC 1948.
- Updates:
  RFC 793.

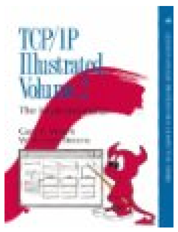
[RFC 6582] The NewReno Modification to TCP's Fast Recovery Algorithm.

- Category: Standards Track.
- Obsoletes:
  RFC 3782.

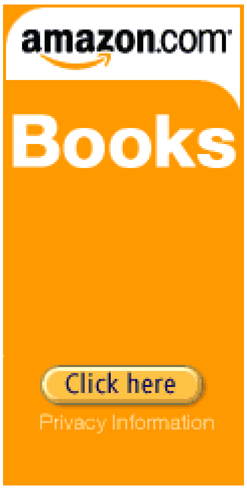
Publications:



TCP/IP Illustrated, Vol. 1
W. Richard Stevens...
Best Price $2.86
or Buy New
Buy from amazon.com
Privacy Information



TCP/IP Illustrated
Gary R. Wright, W....
Best Price $8.20
or Buy New $69.26
Buy from amazon.com
Privacy Information



Obsolete RFCs:

[IEN 112] TRANSMISSION CONTROL PROTOCOL.

- Obsoleted by:
  IEN 124, RFC 761.
- Obsoletes:
  IEN 5, IEN 21, IEN 27, IEN 40, IEN 44, IEN 55, IEN 81.

[IEN 124] DOD STANDARD TRANSMISSION CONTROL PROTOCOL.

- Obsoleted by:
  RFC 761.
- Obsoletes:
  IEN 5, IEN 21, IEN 27, IEN 40, IEN 44, IEN 55, IEN 81, IEN 112.

[RFC 1063] IP MTU Discovery Options.

- Obsoleted by:
  RFC 1191.

[RFC 1066] Management Information Base for Network Management of TCP/IP-based internets.

- Obsoleted by:
  RFC 1156.

[RFC 1072] TCP Extensions for Long-Delay Paths.

- Defines TCP options 3 (Window Scale), 4 (SACK permitted), 5 (SACK), 6 (Echo request), 7 (Echo reply).
- Obsoleted by:
  RFC 1323, RFC 2018.

[RFC 1145] TCP Alternate Checksum Options.

- Defines TCP option 14 (Alternate checksum request), 15 (Alternate checksum data).
- Obsoleted by:
  RFC 1146.

[RFC 1158] Management Information Base for Network Management of TCP/IP-based internets: MIB-II.

- Obsoleted by:
  RFC 1213.

[RFC 1185] TCP Extension for High-Speed Paths.

- Obsoleted by:
  RFC 1323.

[RFC 1948] Defending Against Sequence Number Attacks.

- Category: Informational.
- Obsoleted by:
  RFC 6528.

[RFC 2001] TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms.

- Category: Standards Track.
- Obsoleted by:
  RFC 2581.

[RFC 2012] SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2.

- Category: Standards Track.
- Defines SNMP MIB iso.org.dod.internet.mgmt.mib-2.tcpMIB (1.3.6.1.2.1.49).
- Obsoleted by:
  RFC 4022.
- Updates:
  RFC 1213.

[RFC 2147] TCP and UDP over IPv6 Jumbograms.

- Category: Standards Track.
- Obsoleted by:
  RFC 2675.
- Updates:
  RFC 1883.

[RFC 2414] Increasing TCP's Initial Window.

- Category: Experimental.
- Obsoleted by:
  RFC 3390.

[RFC 2452] IP Version 6 Management Information Base for the Transmission Control Protocol.

- Category: Standards Track.
- Defines SNMP MIB iso.org.dod.internet.experimental.ipv6TcpMIB (1.3.6.1.3.86).
- Obsoleted by:
  RFC 4022.

[RFC 2481] A Proposal to add Explicit Congestion Notification (ECN) to IP.

- Category: Experimental.
- Obsoleted by:
  RFC 3168.

[RFC 2582] The NewReno Modification to TCP's Fast Recovery Algorithm.

- Category: Experimental.
- Obsoleted by:
  RFC 3782.

[RFC 3448] TCP Friendly Rate Control (TFRC): Protocol Specification.

- Category: Standards Track.
- Defines the TFRC algorithm.
- Obsoleted by:
  RFC 5348.

[RFC 3782] The NewReno Modification to TCP's Fast Recovery Algorithm.

- Category: Standards Track.
- Obsoletes:
  RFC 2582.
- Obsoleted by:
  RFC 6582.

RFC Sourcebook Description Glossary RFCs Publications Obsolete RFCs

© 1998 - 2012 Network Sorcery, Inc. All rights reserved.