Wireless Profiled TCP

Version 31 Mar 2001

Open Mobile Alliance
WAP-225-TCP-20010331-a

Continues the Technical Activities
Originated in the WAP Forum

© 2003 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.   [OMA-Template-SpecWAP-20030912]

Contents

1. Scope

Wireless Application Protocol (WAP^{TM}) is a result of continuous work to define an industry wide specification for developing applications that operate over wireless communication networks. The scope for the WAP Forum is to define a set of specifications to be used by service applications. The wireless market is growing very quickly and reaching new customers and providing new services. To enable operators and manufacturers to meet the challenges in advanced services, differentiation, and fast/flexible service creation, WAP defines a set of open, extensible protocols and content formats as a basis for interoperable implementations.

The WAP Architecture Specification [ARCH] defines a Transport Services Layer that provides datagram and connection-oriented services to the upper layer protocols; Wireless Profiled TCP (WP-TCP) provides the connection-oriented services. The inclusion of TCP has been motivated by the emergence of high-speed wireless networks (e.g. 2.5G and 3G). Some of the benefits provided by TCP include Large Data Transfer, End-to-End Security (using TLS) and convergence with IETF protocols. Wireless Profiled TCP is optimised for wireless environments and can interoperate with standard TCP [RFC0793] [RFC1122] implementations in the Internet.

The Wireless Profiled TCP Specification is independent of the IP version supported by the underlying bearers.

2. References

2.1 Normative References

[IOPProc]	"OMA Interoperability Policy and Process". Open Mobile Alliance™. OMA-IOP-Process-v1_0. URL:http//www.openmobilealliance.org/
[RFC0793]	"Transmission Control Protocol". Jon Postel. September 1981. URL:http://www.ietf.org/rfc/rfc0793.txt
[RFC1122]	"Requirements for Internet Hosts – Communication Layers". R. Braden. October 1989. URL:http://www.ietf.org/rfc/rfc1122.txt
[RFC1191]	"Path MTU Discovery". J. Mogul, S. Deering. November 1990. URL:http://www.ietf.org/rfc/rfc1191.txt
[RFC1323]	"TCP Extensions for High Performance". V. Jacobson, R. Braden, D. Borman. May 1992. URL:http://www.ietf.org/rfc/rfc1323.txt
[RFC1981]	"Path MTU Discovery for IP version 6". J. McCann, S. Deering, J. Mogul. August 1996. URL:http://www.ietf.org/rfc/rfc1981.txt
[RFC2018]	"TCP Selective Acknowledgement Options". M. Mathis, J. Madhavi, S. Floyd, A Romanow. October 1996. URL:http://www.ietf.org/rfc/rfc2018.txt
[RFC2119]	"Key words for use in RFCs to Indicate Requirement Levels". S. Bradner. March 1997. URL:http://www.ietf.org/rfc/rfc2119.txt
[RFC2414]	"Increasing TCP's Initial Window". M. Allman, S. Floyd, C. Patridge. October 1996. URL:http://www.ietf.org/rfc/rfc2414.txt
[RFC2481]	"A Proposal to add Explicit Congestion Notification (ECN) to IP". K. Ramakrishnan, S. Floyd. January 1999. URL:http://www.ietf.org/rfc/rfc2481.txt
[RFC2581]	"TCP Congestion Control". M. Allman, V. Paxson, W. Stevens. April 1999. URL:http://www.ietf.org/rfc/rfc2581.txt

2.2 Informative References

[WAPARCH]	"WAP Architecture". WAP Forum™. WAP-210-WAPArch. URL:http//www.wapforum.org/
[ISO7498]	"Information Technology – Open Systems Interconnection – Basic Reference Model: The Basic Model". ISO/IEC 7498-1: 1994.
[RFC1435]	"IESG Advice from Experience with Path MTU Discovery". S. Knowles. March 1993. URL:http://www.ietf.org/rfc/rfc1435.txt
[RFC2507]	"IP Header Compression". M. Degermark, B. Nordgren, S. Pink. February 1999. URL:http://www.ietf.org/rfc/rfc2507.txt
[RFC2509]	"IP Header Compression over PPP". M. Engan, S. Casner, C. Bormann. February 1999. URL:http://www.ietf.org/rfc/rfc2509.txt
[RFC2757]	"Long Thin Networks". G. Montenegro, S. Dawkins, M. Kojo, V. Magret, N. Vaidya. January 2000. URL:http://www.ietf.org/rfc/rfc2757.txt
[RFC2884]	"Performance Evaluation of Explicit Congestion Notification (ECN) in IP Networks". J. Salim, U. Ahmed. July 2000. URL:http://www.ietf.org/rfc/rfc2884.txt

3. Terminology and Conventions

3.1 Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except " Scope" and "Introduction", are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Client – a device (or application) that initiates a request for a connection with a server.

Device – a network entity that is capable of sending and receiving packets of information and has a unique device address. A device can act as both a client and a server within a given context or across multiple contexts. For example, a device can service a number of clients (as a server) while being a client to another server.

Origin Server – the server on which a given resource resides or is to be created. Often referred to as a web server or an HTTP server.

Proxy – an intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients.

Router – an intermediary mechanism that determines the path taken by IP packets.

Server – a device (or application) that passively waits for connection requests from one or more clients. A server may accept or reject a connection request from a client.

Terminal – a device providing the user with user agent capabilities, including the ability to request and receive information. Also called a mobile terminal or mobile station.

User – a user is a person who interacts with a user agent to view, hear, or otherwise use a resource.

User Agent – a user agent is any software or device that interprets WML, WMLScript, WTAI or other resources. This may include textual browsers, voice browsers, search engines, etc.

Web Server – the same as Origin Server.

3.3 Abbreviations

A-SAP	Application – Service Access Point
BDP	Bandwidth Delay Product
BER	Bit Error Rate
ECN	Explicit Congestion Notification
IETF	Internet Engineering Task Force
IP	Internet Protocol
MTA	Mail Transfer Agent
MTU	Maximum Transmission Unit
PILC	Performance Implications of Link Characteristics
RFC	Request For Comments
RTTM	Round Trip Time Measurement
SACK	Selective Acknowledgement
SAP	Service Access Point
SEC-SAP	Security Services – Service Access Point
S-SAP	Session Services – Service Access Point
TCP	Transmission Control Protocol
T-SAP	Transport Services – Service Access Point
TS-SAP	Transfer Services – Service Access Point
TLS	Transport Layer Security
URL	Uniform Resource Locator

WAP   Wireless Application Protocol
WP-TCP   Wireless Profiled TCP
WWW   World Wide Web

4. Introduction

4.1   The WAP Model

WAP allows for the use a proxy technology to connect between the wireless domain and the World Wide Web. The proxy may provide a variety of functions including:

- Protocol Gateway – to translate requests from a wireless protocol stack (e.g. the WAP 1.x stack) to the WWW stack.

- Content Encoders and Decoders.

- User Agent Profile Management.
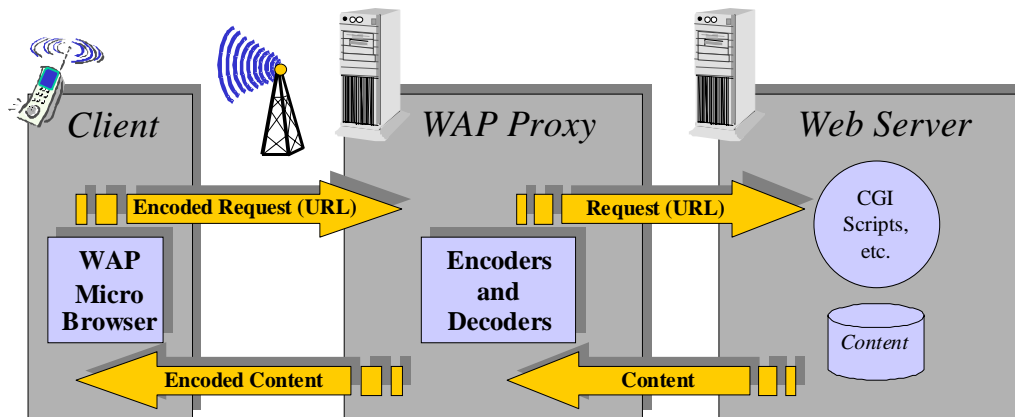
Figure 1: The WAP Model

Feature proxies like the WAP Proxy are widely used in the Internet: common examples include Web proxies and relay MTAs (Mail Transfer Agents). Employing such proxies results in a split-TCP connection with the proxy as the intermediary; the WAP Client establishes a TCP connection to the WAP Proxy, and a Proxy Application (on the WAP Proxy) would then establish a separate TCP connection from the WAP Proxy to the Origin Server. The Proxy Application will subsequently retrieve inbound data from either connection and send that data out through the other connection. The TCP connection between the WAP Client and the WAP Proxy employs the features and characteristics recommended for Wireless Profiled TCP; the proxy thus allows for the optimisation of TCP over the wireless network. The following figure shows a WAP stack employing the split TCP approach.
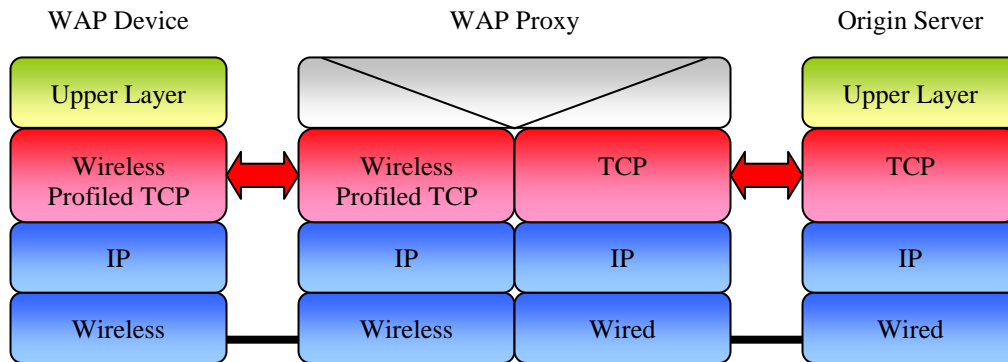
Figure 2: Wireless Profiled TCP with WAP Proxy

The split-TCP approach has a number of advantages: it provides a simple way to shield the problems associated with wireless links from the wireline Internet and vice versa. It also allows for the early deployment of various proposals to improve TCP performance over wireless links as these enhancements can be implemented on the mobile devices and the proxies. A detailed discussion of the advantages and disadvantages of the split-TCP approach is provided in [RFC2757].

Wireless Profiled TCP implementations can also be used for end-to-end connectivity (i.e. without any TCP connection information on intermediate nodes).
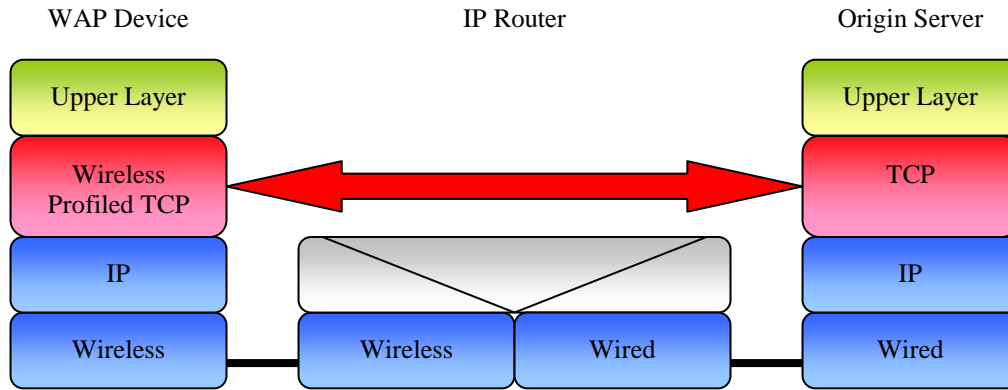
Figure 3: Wireless Profiled TCP Without WAP Proxy

Wireless Profiled TCP implementations must support both modes of operation; i.e. split and end-to-end TCP. The choice of the approach (i.e. split or end-to-end) to be employed for communication between a client and an Origin Server would be determined by factors such as the current provisioning, the application, the network access point etc.

4.2   Reference Model
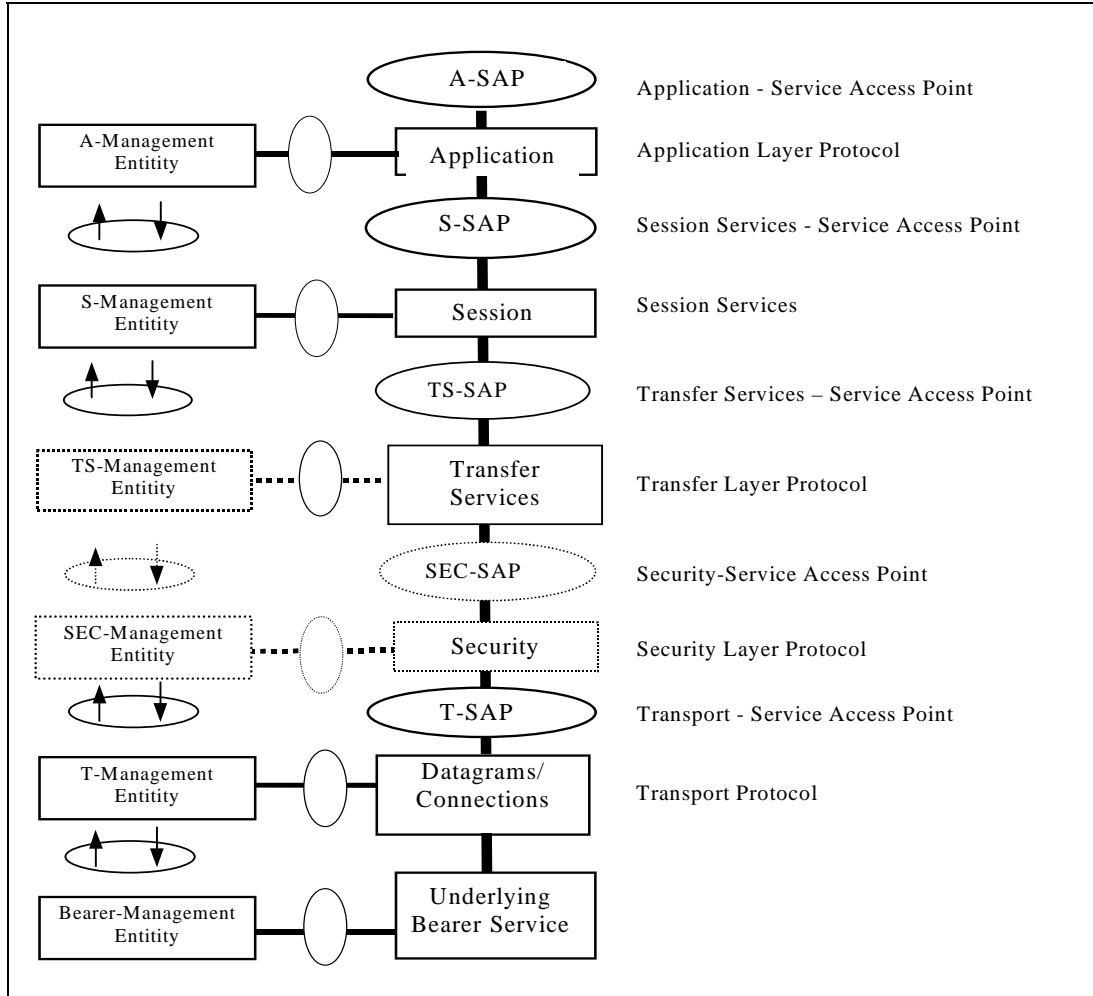
Figure 4: WAP Release 2 Reference Model

A model of layering the protocols in WAP is illustrated in Figure 4. WAP protocols and their functions are layered in a style resembling that of [ISO7498]. The Management Entities handle protocol initialisation, configuration and error conditions (such as loss of connectivity due to the mobile station roaming out of coverage) that are not handled by the protocol itself.

5. TCP Optimisations

The requirements for Host TCP implementations are described in [RFC0793][RFC1122], with additional requirements for congestion control and recovery specified in [RFC2581]; Wireless Profiled TCP implementations on devices and gateways MUST conform to these requirements.

This section describes additional requirements for Wireless Profiled TCP implementations on devices and gateways. It should be noted that implementations using header compression [RFC2507, RFC2509] might find that the compression factor is impacted by the presence of TCP options. For example both Windows Scale Option and Timestamps Option described below have a negative impact on header compression.

Wireless Profiled TCP implementations MUST support split and end-to-end modes of operation. When operating in either mode implementations MAY employ extensions beyond those listed in this document; such extensions MUST only be used if they can be negotiated in a manner consistent with other TCP extensions.

5.1 Motivation for Optimisations

This section is informative.

Cellular networks are characterized by high bit error rates, relatively long delays and variable bandwidth and delays. TCP performance in such environments degrades on account of the following reasons:

- Packet losses on account of corruption are treated as congestion losses and lead to reduction of the congestion window and slow recovery.
- TCP window sizes tend to stay small for long periods of time in high BER environments.
- The use of exponential back-off retransmission mechanisms increases the retransmission timeout resulting in long periods of silence or connection loss
- Independent timers in the link and transport layer may trigger redundant retransmissions.
- Periods of disconnection because of handoffs or the absence of coverage.

Research in optimising TCP has resulted in a number of mechanisms to improve performance. Some of these mechanisms are documented in Standards Track RFCs and have been accepted by the Internet community as useful and technically stable. The IETF PILC group has recommended the use of some of these mechanisms for TCP implementations in long thin networks [RFC2757].

5.2 Large Window Size

The minimum window size required to maximize TCP performance is computed by Bandwidth Delay Product (BDP), where Bandwidth is the available bandwidth and Delay is the round-trip time of the given path [RFC1323]. The maximum window size is the minimum of the send and receive socket buffer. The receive socket buffer generally determines the advertisement window on the receiver. The congestion window on the sender further limits the amount of data the sender can inject into the network depending on the congestion level on the path.

If the maximum window size is too small, relative to the available bandwidth of the network path, the TCP connection will not be able to fully utilize the available capacity. If the maximum window is too large for the network path to handle, the congestion window will eventually grow to the point where TCP will overwhelm the network with too many segments, some of which will be discarded before reaching the destination.

In real networks it is very difficult to pick the appropriate window size because of dynamic characteristics of bandwidth, delay, and congestion. However if the maximum window size is large enough to overwhelm the network (and the necessary buffer sizes are available), the TCP congestion control algorithms will find a congestion window size that is appropriate for the network path. Wireless Profiled TCP implementations SHOULD support large window sizes based on the BDP.

5.3   Window Scale Option

Wireless Profiled TCP implementations in WAP clients MAY support window sizes larger than or equal to 64 KB based on the BDP, while implementations in WAP servers and proxies SHOULD support those sizes. Implementations that allow such window sizes MUST support the Window Scale Option [RFC1323]. Implementations whose window sizes are less than 64 KB MAY use the Window Scale Option [RFC1323].

5.4   Round Trip Time Measurement

[RFC1323] recommends that TCP implementations supporting large windows of 64 KB and larger use the RTTM mechanism to obtain better round trip time estimates. Wireless Profiled TCP implementations may employ this mechanism; this would require implementation of the Timestamps Option. If window size is larger than or equal to 64 KB, implementations SHOULD support Timestamps Option for RTTM, and otherwise this option for RTTM MAY be supported.

5.5   Large Initial Window

The Slow Start algorithm requires that a sender MUST use an Initial Window (i.e. the initial size of the congestion window) of up to two segments. This recommendation also applies to the Restart Window (i.e. the congestion window size when restarting transmission after an idle period).

[RFC2414] describes a non-standard, experimental TCP extension that allows the use of an initial window of three or four segments with an upper limit of 4380 bytes. Wireless Profiled TCP implementations MAY support this extension.

5.6   Path MTU Discovery

Path MTU discovery allows a sender to determine the maximum end-to-end transmission unit for a given routing path. [RFC1191] and [RFC1981] describe the MTU discovery procedure for IPv4 and IPv6 respectively. This allows TCP senders to employ larger segment sizes (without causing fragmentation) instead of assuming the default MTU. Using larger segment sizes allows for a faster increase in the congestion window and a smaller ratio of header overhead to data. It should be noted that larger MTUs increase the probability of error in a given segment and also increase the packet transmission time. Wireless Profile TCP implementations SHOULD implement Path MTU Discovery.

Path MTU Discovery requires intermediate routers to support the generation of the necessary ICMP messages. [RFC1435] provides recommendations that may be relevant for some router implementations.

5.7   MTU Larger than Default IP MTU

Wireless Profiled TCP implementations that cannot support MTU Discovery MAY assume a path MTU larger than the default IPv4 or IPv6 values; however assuming an MTU larger than 1500 bytes is not recommended. The MTU value chosen must reflect the MSS value used in the MSS option in the SYN packets that initiated this connection.

5.8   Selective Acknowledgement

Selective Acknowledgement [RFC2018] is especially useful when there is a considerable probability of multiple segment losses per window; such losses are likely when using large windows or when there is a high possibility of burst errors and

congestion losses on the wireless link. The Fast Recovery algorithm is not known to be very efficient when recovering from multiple losses in a single window. Wireless Profiled TCP implementations MUST support SACK.

5.9 Explicit Congestion Notification

Explicit Congestion Notification [RFC2481] allows a TCP receiver to inform the sender of congestion in the network by setting the ECN-Echo flag; a receiver will set this flag on receiving an IP packet marked with the CE bit. The TCP sender can then reduce its congestion window. This proposal is still in an experimental/informational state and is believed to provide performance benefits [RFC2884]. Wireless Profiled TCP implementations MAY support ECN.

[RFC2481] also places requirements on intermediate routers (e.g. active queue management and setting of the CE bit in the IP header to indicate congestion). Thus the use of ECN on the TCP connections is dependent on the necessary support from the relevant IP routers.

5.10 TCP Optimisations Summary

The RFCs used in Wireless Profiled TCP are summarized in the following table.

Items	Qualifier	Support level
Large window size based on BDP		SHOULD
Window Scale Option [RFC1323]	Window size >= 64KB	MUST
	Window size < 64KB	MAY
Timestamps Option [RFC1323] for RTTM	Window size >= 64KB	SHOULD
	Window size < 64KB	MAY
Large Initial Window (cwnd<=2) [RFC2581]		MUST
Large Initial Window (cwnd>2) [RFC2414]		MAY
Selective Acknowledgement Option (SACK) [RFC2018]		MUST
Path MTU Discovery [RFC1191, RFC1981]		SHOULD
MTU larger than default IP MTU	Path MTU Discovery NOT Supported	MAY
Explicit Congestion Notification (ECN) [RFC2481]		MAY

6. Elements of Layer to Layer Communication

Wireless Profiled TCP interfaces MUST conform to the application layer interfaces documented in [RFC0793] and [RFC1122].

7. State Tables

The basic state transitions for TCP are documented in [RFC0793]. Various TCP extensions modify this basic behaviour; these modifications are documented in the corresponding RFC (for e.g. [RFC2581], [RFC1323]). Wireless Profiled TCP implementations MUST conform to the state transitions documented in [RFC0793] and the RFCs corresponding to the implemented optimisations.

Appendix A.   Implementation Notes   (Informative)

A.1 Explicit Congestion Notification

There have been reports of ECN-capable hosts being unable to establish TCP connections; these problems have been attributed to problems in TCP implementations at the peer and to firewalls dropping IP packets that have the ECN related bits in the IP header set. Additional information on this issue can be found at (http://www.aciri.org/tbit/). It is recommended that Wireless Profiled TCP be implemented such that ECN can be turned off if necessary.

Appendix B.   Static Conformance Requirements   (Normative)

This static conformance clause defines a minimum set of feature that can be implemented to ensure that the implementation will be able to inter-operate. A feature can be optional or mandatory. If a Wireless Profiled TCP layer implementation does not support an optional feature, transmission must occur without error, but may not be optimal.

The notation used in this appendix is specified in [IOPProc].

B.1. Wireless Profiled TCP Client

Item	Function	Reference	Status	Requirement
TCP-C-001	Transmission Control Protocol (TCP) [RFC0793], [RFC1122], [RFC2581]	Section 6, 7, 8	M	
TCP-C-002	Large Window Size	Section 6.2	O	
TCP-C-003	Window size >= 64KB	Section 6.2	O	TCP-C-004
TCP-C-004	Window Scale Option	Section 6.2	O	
TCP-C-005	Timestamps Option	Section 6.2	O	
TCP-C-006	Large Initial Window (cwnd<=2)	Section 6.3	M	
TCP-C-007	Large Initial Window (cwnd>2)	Section 6.3	O	
TCP-C-008	Path MTU Discovery	Section 6.4	O	
TCP-C-009	Selective Acknowledgement	Section 6.6	M	
TCP-C-010	Explicit Congestion Notification	Section 6.7	O	

B.2. Wireless Profiled TCP Server

Item	Function	Reference	Status	Requirement
TCP-S-001	Transmission Control Protocol (TCP) [RFC0793], [RFC1122], [RFC2581]	Section 6, 7, 8	M	
TCP-S-002	Large Window Size	Section 6.2	O	
TCP-S-003	Window size >= 64KB	Section 6.2	O	TCP-S-004
TCP-S-004	Window Scale Option	Section 6.2	O	
TCP-S-005	Timestamps Option	Section 6.2	O	
TCP-S-006	Large Initial Window (cwnd<=2)	Section 6.3	M	
TCP-S-007	Large Initial Window (cwnd>2)	Section 6.3	O	
TCP-S-008	Path MTU Discovery	Section 6.4	O	
TCP-S-009	Selective Acknowledgement	Section 6.6	M	
TCP-S-010	Explicit Congestion Notification	Section 6.7	O	

Appendix C.   Change History   (Informative)

C.1   Approved Version History

Reference	Date	Description
Class 0	31-March-2001	Initial Approved version.
Class 3	16 Dec 2003	Updated for OMA template & minor consistency comments
WAP-225-TCP-20010331-a	10-Jan-2005	Revert document numbering to WAP Forum format. Adjust history for 16-12-2003 as should be Class 3