      TCP over Second (2.5G) and Third (3G) Generation Wireless Networks

Status of this Memo

Copyright Notice

Abstract

   This document describes a profile for optimizing TCP to adapt so that
   it handles paths including second (2.5G) and third (3G) generation
   wireless networks. It describes the relevant characteristics of 2.5G
   and 3G networks, and specific features of example deployments of such
   networks. It then recommends TCP algorithm choices for nodes known
   to be starting or ending on such paths, and it also discusses open
   issues. The configuration options recommended in this document are
   commonly found in modern TCP stacks, and are widely available
   standards-track mechanisms that the community considers safe for use
   on the general Internet.

Table of Contents

1. Introduction

   The second generation cellular systems are commonly referred to as
   2G. The 2G phase began in the 1990s when digital voice encoding had
   replaced analog systems (1G). 2G systems are based on various radio
   technologies including frequency-, code- and time- division multiple
   access. Examples of 2G systems include GSM (Europe), PDC (Japan),
   and IS-95 (USA). Data links provided by 2G systems are mostly
   circuit-switched and have transmission speeds of 10-20 kbps uplink
   and downlink. Demand for higher data rates, instant availability and
   data volume-based charging, as well as lack of radio spectrum
   allocated for 2G led to the introduction of 2.5G (for example, GPRS
   and PDC-P) and 3G (for example, Wideband CDMA and cdma2000) systems.

   Radio technology for both Wideband CDMA (W-CDMA) (adopted, for
   example, in Europe, Japan, etc) and cdma2000 (adopted, for example,
   in US, South Korea, etc) is based on code division multiple access
   allowing for higher data rates and more efficient spectrum
   utilization than 2G systems. 3G systems provide both packet-switched
   and circuit-switched connectivity in order to address the quality of
   service requirements of conversational, interactive, streaming, and
   bulk transfer applications. The transition to 3G is expected to be a
   gradual process. Initially, 3G will be deployed to introduce high
   capacity and high speed access in densely populated areas. Mobile
   users with multimode terminals will be able to utilize existing
   coverage of 2.5G systems on the rest of territory.

   Much development and deployment activity has centered around 2.5G and
   3G technologies. Along with objectives like increased capacity for
   voice channels, a primary motivation for these is data communication,
   and, in particular, Internet access. Accordingly, key issues are TCP
   performance and the several techniques which can be applied to
   optimize it over different wireless environments [19].

   This document proposes a profile of such techniques, (particularly
   effective for use with 2.5G and 3G wireless networks). The
   configuration options in this document are commonly found in modern
   TCP stacks, and are widely available IETF standards-track mechanisms
   that the community has judged to be safe on the general Internet
   (that is, even in predominantly non-wireless scenarios).
   Furthermore, this document makes one set of recommendations that
   covers both 2.5G and 3G networks. Since both generations of wireless
   technologies exhibit similar challenges to TCP performance (see
   Section 2), one common set is warranted.

   Two example applications of the recommendations in this document are:

   o The WAP Forum [25] (part of the Open Mobile Alliance [26] as of
      June 2002) is an industry association that has developed standards
      for wireless information and telephony services on digital mobile
      phones. In order to address WAP functionality for higher speed
      networks such as 2.5G and 3G networks, and to aim at convergence
      with Internet standards, the WAP Forum thoroughly revised its
      specifications. The resultant version 2.0 [31] adopts TCP as its
      transport protocol, and recommends TCP optimization mechanisms
      closely aligned with those described in this document.

   o I-mode [33] is a wireless Internet service deployed on handsets in
      Japan. The newer version of i-mode runs on FOMA [34], an
      implementation of W-CDMA. I-mode over FOMA deploys the profile of
      TCP described in this document.

   This document is structured as follows: Section 2 reviews the link
   layer characteristics of 2.5G/3G networks; Section 3 gives a brief
   overview of some representative 2.5G/3G technologies like W-CDMA,
   cdma2000 and GPRS; Section 4 recommends mechanisms and configuration
   options for TCP implementations used in 2.5G/3G networks, including a
   summary in chart form at the end of the section; finally, Section 5
   discusses some open issues.

2. 2.5G and 3G Link Characteristics

   Link layer characteristics of 2.5G/3G networks have significant
   effects on TCP performance. In this section we present various
   aspects of link characteristics unique to the 2.5G/3G networks.

2.1 Latency

   The latency of 2.5G/3G links is high mostly due to the extensive
   processing required at the physical layer of those networks, e.g.,
   for FEC and interleaving, and due to transmission delays in the radio
   access network [58] (including link-level retransmissions). A
   typical RTT varies between a few hundred milliseconds and one second.
   The associated radio channels suffer from difficult propagation
   environments. Hence, powerful but complex physical layer techniques
   need to be applied to provide high capacity in a wide coverage area
   in a resource efficient way. Hopefully, rapid improvements in all
   areas of wireless networks ranging from radio layer techniques over
   signal processing to system architecture will ultimately also lead to
   reduced delays in 3G wireless systems.

2.2 Data Rates

   The main incentives for transition from 2G to 2.5G to 3G are the
   increase in voice capacity and in data rates for the users. 2.5G
   systems have data rates of 10-20 kbps in uplink and 10-40 kbps in
   downlink. Initial 3G systems are expected to have bit rates around
   64 kbps in uplink and 384 kbps in downlink. Considering the
   resulting bandwidth-delay product (BDP) of around 1-5 KB for 2.5G and
   8-50 KB for 3G, 2.5G links can be considered LTNs (Long Thin Networks
   [19]), and 3G links approach LFNs (Long Fat Networks [2], as
   exemplified by some satellite networks [48]). Accordingly,
   interested readers might find related and potentially relevant issues
   discussed in RFC 2488 [49]. For good TCP performance both LFNs and
   LTNs require maintaining a large enough window of outstanding data.
   For LFNs, utilizing the available network bandwidth is of particular
   concern. LTNs need a sufficiently large window for efficient loss
   recovery. In particular, the fast retransmit algorithm cannot be
   triggered if the window is less than four segments. This leads to a
   lengthy recovery through retransmission timeouts. The Limited
   Transmit algorithm RFC 3042 [10] helps avoid the deleterious effects
   of timeouts on connections with small windows. Nevertheless, making
   full use of the SACK RFC 2018 [3] information for loss recovery in
   both LFNs and LTNs may require twice the window otherwise sufficient
   to utilize the available bandwidth.

   This document recommends only standard mechanisms suitable both for
   LTNs and LFNs, and to any network in general. However, experimental
   mechanisms suggested in Section 5 can be targeted either for LTNs
   [19] or LFNs [48].

   Data rates are dynamic due to effects from other users and from
   mobility. Arriving and departing users can reduce or increase the
   available bandwidth in a cell. Increasing the distance from the base
   station decreases the link bandwidth due to reduced link quality.
   Finally, by simply moving into another cell the user can experience a
   sudden change in available bandwidth. For example, if upon changing
   cells a connection experiences a sudden increase in available
   bandwidth, it can underutilize it, because during congestion
   avoidance TCP increases the sending rate slowly. Changing from a
   fast to a slow cell normally is handled well by TCP due to the self-
   clocking property. However, a sudden increase in RTT in this case
   can cause a spurious TCP timeout as described in Section 2.7. In
   addition, a large TCP window used in the fast cell can create
   congestion resulting in overbuffering in the slow cell.

2.3 Asymmetry

   2.5G/3G systems may run asymmetric uplink and downlink data rates.
   The uplink data rate is limited by battery power consumption and
   complexity limitations of mobile terminals. However, the asymmetry
   does not exceed 3-6 times, and can be tolerated by TCP without the
   need for techniques like ACK congestion control or ACK filtering
   [50]. Accordingly, this document does not include recommendations
   meant for such highly asymmetric networks.

2.4 Delay Spikes

   A delay spike is a sudden increase in the latency of the
   communication path. 2.5G/3G links are likely to experience delay
   spikes exceeding the typical RTT by several times due to the
   following reasons.

   1. A long delay spike can occur during link layer recovery from a
      link outage due to temporal loss of radio coverage, for example,
      while driving into a tunnel or within an elevator.

   2. During a handover the mobile terminal and the new base station
      must exchange messages and perform some other time-consuming
      actions before data can be transmitted in a new cell.

   3. Many wide area wireless networks provide seamless mobility by
      internally re-routing packets from the old to the new base station
      which may cause extra delay.

   4. Blocking by high-priority traffic may occur when an arriving
      circuit-switched call or higher priority data temporarily preempts
      the radio channel. This happens because most current terminals
      are not able to handle a voice call and a data connection
      simultaneously and suspend the data connection in this case.

   5. Additionally, a scheduler in the radio network can suspend a low-
      priority data transfer to give the radio channel to higher
      priority users.

   Delay spikes can cause spurious TCP timeouts, unnecessary
   retransmissions and a multiplicative decrease in the congestion
   window size.

2.5 Packet Loss Due to Corruption

   Even in the face of a high probability of physical layer frame
   errors, 2.5G/3G systems have a low rate of packet losses thanks to
   link-level retransmissions. Justification for link layer ARQ is
   discussed in [23], [22], [44]. In general, link layer ARQ and FEC
   can provide a packet service with a negligibly small probability of
   undetected errors (failures of the link CRC), and a low level of loss
   (non-delivery) for the upper layer traffic, e.g., IP. The loss rate
   of IP packets is low due to the ARQ, but the recovery at the link
   layer appears as delay jitter to the higher layers lengthening the
   computed RTO value.

2.6 Intersystem Handovers

   In the initial phase of deployment, 3G systems will be used as a 'hot
   spot' technology in high population areas, while 2.5G systems will
   provide lower speed data service elsewhere. This creates an
   environment where a mobile user can roam between 2.5G and 3G networks
   while keeping ongoing TCP connections. The inter-system handover is
   likely to trigger a high delay spike (Section 2.4), and can result in
   data loss. Additional problems arise because of context transfer,
   which is out of scope of this document, but is being addressed
   elsewhere in the IETF in activities addressing seamless mobility
   [51].

   Intersystem handovers can adversely affect ongoing TCP connections
   since features may only be negotiated at connection establishment and
   cannot be changed later. After an intersystem handover, the network
   characteristics may be radically different, and, in fact, may be
   negatively affected by the initial configuration. This point argues
   against premature optimization by the TCP implementation.

2.7 Bandwidth Oscillation

   Given the limited RF spectrum, satisfying the high data rate needs of
   2.5G/3G wireless systems requires dynamic resource sharing among
   concurrent data users. Various scheduling mechanisms can be deployed
   in order to maximize resource utilization. If multiple users wish to
   transfer large amounts of data at the same time, the scheduler may
   have to repeatedly allocate and de-allocate resources for each user.
   We refer to periodic allocation and release of high-speed channels as
   Bandwidth Oscillation. Bandwidth Oscillation effects such as
   spurious retransmissions were identified elsewhere (e.g., [30]) as
   factors that degrade throughput. There are research studies [52],
   [54], which show that in some cases Bandwidth Oscillation can be the
   single most important factor in reducing throughput. For fixed TCP
   parameters the achievable throughput depends on the pattern of

   resource allocation. When the frequency of resource allocation and
   de-allocation is sufficiently high, there is no throughput
   degradation. However, increasing the frequency of resource
   allocation/de-allocation may come at the expense of increased
   signaling, and, therefore, may not be desirable. Standards for 3G
   wireless technologies provide mechanisms that can be used to combat
   the adverse effects of Bandwidth Oscillation. It is the consensus of
   the PILC Working Group that the best approach for avoiding adverse
   effects of Bandwidth Oscillation is proper wireless sub-network
   design [23].

3. Example 2.5G and 3G Deployments

   This section provides further details on a few example 2.5G/3G
   technologies. The objective is not completeness, but merely to
   discuss some representative technologies and the issues that may
   arise with TCP performance. Other documents discuss the underlying
   technologies in more detail. For example, ARQ and FEC are discussed
   in [23], while further justification for link layer ARQ is discussed
   in [22], [44].

3.1 2.5G Technologies: GPRS, HSCSD and CDMA2000 1XRTT

   High Speed Circuit-Switched Data (HSCSD) and General Packet Radio
   Service (GPRS) are extensions of GSM providing high data rates for a
   user. Both extensions were developed first by ETSI and later by
   3GPP. In GSM, a user is assigned one timeslot downlink and one
   uplink. HSCSD allocates multiple timeslots to a user creating a fast
   circuit-switched link. GPRS is based on packet-switched technology
   that allows efficient sharing of radio resources among users and
   always-on capability. Several terminals can share timeslots. A GPRS
   network uses an updated base station subsystem of GSM as the access
   network; the GPRS core network includes Serving GPRS Support Nodes
   (SGSN) and Gateway GPRS Support Nodes (GGSN). The RLC protocol
   operating between a base station controller and a terminal provides
   ARQ capability over the radio link. The Logical Link Control (LLC)
   protocol between the SGSN and the terminal also has an ARQ capability
   utilized during handovers.

3.2 A 3G Technology: W-CDMA

   The International Telecommunication Union (ITU) has selected Wideband
   Code Division Multiple Access (W-CDMA) as one of the global telecom
   systems for the IMT-2000 3G mobile communications standard. W-CDMA
   specifications are created in the 3rd Generation Partnership Project
   (3GPP).

The link layer characteristics of the 3G network which have the
largest effect on TCP performance over the link are error controlling
schemes such as layer two ARQ (L2 ARQ) and FEC (forward error
correction).

W-CDMA uses RLC (Radio Link Control) [20], a Selective Repeat and
sliding window ARQ. RLC uses protocol data units (PDUs) with a 16
bit RLC header. The size of the PDUs may vary. Typically, 336 bit
PDUs are implemented [34]. This is the unit for link layer
retransmission. The IP packet is fragmented into PDUs for
transmission by RLC. (For more fragmentation discussion, see Section
4.4.)

In W-CDMA, one to twelve PDUs (RLC frames) constitute one FEC frame,
the actual size of which depends on link conditions and bandwidth
allocation. The FEC frame is the unit of interleaving. This
accumulation of PDUs for FEC adds part of the latency mentioned in
Section 2.1.

For reliable transfer, RLC has an acknowledged mode for PDU
retransmission. RLC uses checkpoint ARQ [20] with "status report"
type acknowledgments; the poll bit in the header explicitly solicits
the peer for a status report containing the sequence number that the
peer acknowledges. The use of the poll bit is controlled by timers
and by the size of available buffer space in RLC. Also, when the
peer detects a gap between sequence numbers in received frames, it
can issue a status report to invoke retransmission. RLC preserves
the order of packet delivery.

The maximum number of retransmissions is a configurable RLC parameter
that is specified by RRC [39] (Radio Resource Controller) through RLC
connection initialization. The RRC can set the maximum number of
retransmissions (up to a maximum of 40). Therefore, RLC can be
described as an ARQ that can be configured for either HIGH-
PERSISTENCE or LOW-PERSISTENCE, not PERFECT-PERSISTENCE, according to
the terminology in [22].

Since the RRC manages RLC connection state, Bandwidth Oscillation
(Section 2.7) can be eliminated by the RRC's keeping RF resource on
an RLC connection with data in its queue. This avoids resource de-
allocation in the middle of transferring data.

In summary, the link layer ARQ and FEC can provide a packet service
with a negligibly small probability of undetected error (failure of
the link CRC), and a low level of loss (non-delivery) for the upper
layer traffic, i.e., IP. Retransmission of PDUs by ARQ introduces
latency and delay jitter to the IP flow. This is why the transport
layer sees the underlying W-CDMA network as a network with a

   relatively large BDP (Bandwidth-Delay Product) of up to 50 KB for the
   384 kbps radio bearer.

3.3 A 3G Technology: CDMA2000 1X-EV

   One of the Terrestrial Radio Interface standards for 3G wireless
   systems, proposed under the International Mobile Telecommunications-
   2000 umbrella, is cdma2000 [55]. It employs Multi-Carrier Code
   Division Multiple Access (CDMA) technology with a single-carrier RF
   bandwidth of 1.25 MHz. cdma2000 evolved from IS-95 [56], a 2G
   standard based on CDMA technology. The first phase of cdma2000
   utilizes a single carrier and is designed to double the voice
   capacity of existing CDMA (IS-95) networks and to support always-on
   data transmission speeds of up to 316.8 kbps. As mentioned above,
   these enhanced capabilities are delivered by cdma2000 1XRTT. 3G
   speeds of 2 Mbps are offered by cdma2000 1X-EV. At the physical
   layer, the standard allows transmission in 5,10,20,40 or 80 ms time
   frames. Various orthogonal (Walsh) codes are used for channel
   identification and to achieve higher data rates.

   Radio Link Protocol Type 3 (RLP) [57] is used with a cdma2000 Traffic
   Channel to support CDMA data services. RLP provides an octet stream
   transport service and is unaware of higher layer framing. There are
   several RLP frame formats. RLP frame formats with higher payload
   were designed for higher data rates. Depending on the channel speed,
   one or more RLP frames can be transmitted in a single physical layer
   frame.

   RLP can substantially decrease the error rate exhibited by CDMA
   traffic channels [53]. When transferring data, RLP is a pure NAK-
   based finite selective repeat protocol. The receiver does not
   acknowledge successfully received data frames. If one or more RLP
   data frames are missing, the receiving RLP makes several attempts
   (called NAK rounds) to recover them by sending one or more NAK
   control frames to the transmitter. Each NAK frame must be sent in a
   separate physical layer frame. When RLP supplies the last NAK
   control frame of a particular NAK round, a retransmission timer is
   set. If the missing frame is not received when the timer expires,
   RLP may try another NAK round. RLP may not recover all missing
   frames. If after all RLP rounds, a frame is still missing, RLP
   supplies data with a missing frame to the higher layer protocols.

4. TCP over 2.5G and 3G

   What follows is a set of recommendations for configuration parameters
   for protocol stacks which will be used to support TCP connections
   over 2.5G and 3G wireless networks. Some of these recommendations
   imply special configuration:

   o at the data receiver (frequently a stack at or near the wireless
      device),

   o at the data sender (frequently a host in the Internet or possibly
      a gateway or proxy at the edge of a wireless network), or

   o at both.

   These configuration options are commonly available IETF standards-
   track mechanisms considered safe on the general Internet. System
   administrators are cautioned, however, that increasing the MTU size
   (Section 4.4) and disabling RFC 1144 header compression (Section 4.9)
   could affect host efficiency, and that changing such parameters
   should be done with care.

4.1 Appropriate Window Size (Sender & Receiver)

   TCP over 2.5G/3G should support appropriate window sizes based on the
   Bandwidth Delay Product (BDP) of the end-to-end path (see Section
   2.2). The TCP specification [14] limits the receiver window size to
   64 KB. If the end-to-end BDP is expected to be larger than 64 KB,
   the window scale option [2] can be used to overcome that limitation.
   Many operating systems by default use small TCP receive and send
   buffers around 16KB. Therefore, even for a BDP below 64 KB, the
   default buffer size setting should be increased at the sender and at
   the receiver to allow a large enough window.

4.2 Increased Initial Window (Sender)

   TCP controls its transmit rate using the congestion window mechanism.
   The traditional initial window value of one segment, coupled with the
   delayed ACK mechanism [17] implies unnecessary idle times in the
   initial phase of the connection, including the delayed ACK timeout
   (typically 200 ms, but potentially as much as 500 ms) [4]. Senders
   can avoid this by using a larger initial window of up to four
   segments (not to exceed roughly 4 KB) [4]. Experiments with
   increased initial windows and related measurements have shown (1)
   that it is safe to deploy this mechanism (i.e., it does not lead to
   congestion collapse), and (2) that it is especially effective for the
   transmission of a few TCP segments' worth of data (which is the
   behavior commonly seen in such applications as Internet-enabled
   mobile wireless devices). For large data transfers, on the other
   hand, the effect of this mechanism is negligible.

   TCP over 2.5G/3G SHOULD set the initial CWND (congestion window)
   according to Equation 1 in [4]:

                   min (4*MSS, max (2*MSS, 4380 bytes))

   This increases the permitted initial window from one to between two
   and four segments (not to exceed approximately 4 KB).

4.3 Limited Transmit (Sender)

   RFC 3042 [10], Limited Transmit, extends Fast Retransmit/Fast
   Recovery for TCP connections with small congestion windows that are
   not likely to generate the three duplicate acknowledgements required
   to trigger Fast Retransmit [1]. If a sender has previously unsent
   data queued for transmission, the limited transmit mechanism calls
   for sending a new data segment in response to each of the first two
   duplicate acknowledgments that arrive at the sender. This mechanism
   is effective when the congestion window size is small or if a large
   number of segments in a window are lost. This may avoid some
   retransmissions due to TCP timeouts. In particular, some studies
   [10] have shown that over half of a busy server's retransmissions
   were due to RTO expiration (as opposed to Fast Retransmit), and that
   roughly 25% of those could have been avoided using Limited Transmit.
   Similar to the discussion in Section 4.2, this mechanism is useful
   for small amounts of data to be transmitted. TCP over 2.5G/3G
   implementations SHOULD implement Limited Transmit.

4.4 IP MTU Larger than Default

   The maximum size of an IP datagram supported by a link layer is the
   MTU (Maximum Transfer Unit). The link layer may, in turn, fragment
   IP datagrams into PDUs. For example, on links with high error rates,
   a smaller link PDU size increases the chance of successful
   transmission. With layer two ARQ and transparent link layer
   fragmentation, the network layer can enjoy a larger MTU even in a
   relatively high BER (Bit Error Rate) condition. Without these
   features in the link, a smaller MTU is suggested.

   TCP over 2.5G/3G should allow freedom for designers to choose MTU
   values ranging from small values (such as 576 bytes) to a large value
   that is supported by the type of link in use (such as 1500 bytes for
   IP packets on Ethernet). Given that the window is counted in units
   of segments, a larger MTU allows TCP to increase the congestion
   window faster [5]. Hence, designers are generally encouraged to
   choose larger values. These may exceed the default IP MTU values of
   576 bytes for IPv4 RFC 1191 [6] and 1280 bytes for IPv6 [18]. While
   this recommendation is applicable to 3G networks, operation over 2.5G
   networks should exercise caution as per the recommendations in RFC
   3150 [5].

4.5 Path MTU Discovery (Sender & Intermediate Routers)

   Path MTU discovery allows a sender to determine the maximum end-to-
   end transmission unit (without IP fragmentation) for a given routing
   path. RFC 1191 [6] and RFC 1981 [8] describe the MTU discovery
   procedure for IPv4 and IPv6, respectively. This allows TCP senders
   to employ larger segment sizes (without causing IP layer
   fragmentation) instead of assuming the small default MTU. TCP over
   2.5G/3G implementations should implement Path MTU Discovery. Path
   MTU Discovery requires intermediate routers to support the generation
   of the necessary ICMP messages. RFC 1435 [7] provides
   recommendations that may be relevant for some router implementations.

4.6 Selective Acknowledgments (Sender & Receiver)

   The selective acknowledgment option (SACK), RFC 2018 [3], is
   effective when multiple TCP segments are lost in a single TCP window
   [24]. In particular, if the end-to-end path has a large BDP and a
   high packet loss rate, the probability of multiple segment losses in
   a single window of data increases. In such cases, SACK provides
   robustness beyond TCP-Tahoe and TCP-Reno [21]. TCP over 2.5G/3G
   SHOULD support SACK.

   In the absence of SACK feature, the TCP should use NewReno RFC 2582
   [15].

4.7 Explicit Congestion Notification (Sender, Receiver & Intermediate
    Routers)

   Explicit Congestion Notification, RFC 3168 [9], allows a TCP receiver
   to inform the sender of congestion in the network by setting the
   ECN-Echo flag upon receiving an IP packet marked with the CE bit(s).
   The TCP sender will then reduce its congestion window. Thus, the use
   of ECN is believed to provide performance benefits [32], [43]. RFC
   3168 [9] also places requirements on intermediate routers (e.g.,
   active queue management and setting of the CE bit(s) in the IP header
   to indicate congestion). Therefore, the potential improvement in
   performance can only be achieved when ECN capable routers are
   deployed along the path. TCP over 2.5G/3G SHOULD support ECN.

4.8 TCP Timestamps Option (Sender & Receiver)

   Traditionally, TCPs collect one RTT sample per window of data [14],
   [17]. This can lead to an underestimation of the RTT, and spurious
   timeouts on paths in which the packet transmission delay dominates
   the RTT. This holds despite a conservative retransmit timer such as
   the one specified in RFC 2988 [11]. TCP connections with large
   windows may benefit from more frequent RTT samples provided with

timestamps by adapting quicker to changing network conditions [2].
However, there is some empirical evidence that for TCPs with an RFC
2988 timer [11], timestamps provide little or no benefits on backbone
Internet paths [59].   Using the TCP Timestamps option has the
advantage that retransmitted segments can be used for RTT
measurement, which is otherwise forbidden by Karn's algorithm [17],
[11].   Furthermore, the TCP Timestamps option is the basis for
detecting spurious retransmits using the Eifel algorithm [30].

A 2.5/3G link (layer) is dedicated to a single host. It therefore
only experiences a low degree of statistical multiplexing between
different flows. Also, the packet transmission and queuing delays of
a 2.5/3G link often dominate the path's RTT. This already results in
large RTT variations as packets fill the queue while a TCP sender
probes for more bandwidth, or as packets drain from the queue while a
TCP sender reduces its load in response to a packet loss. In
addition, the delay spikes across a 2.5/3G link (see Section 2.4) may
often exceed the end-to-end RTT. The thus resulting large variations
in the path's RTT may often cause spurious timeouts.

When running TCP in such an environment, it is therefore advantageous
to sample the path's RTT more often than only once per RTT. This
allows the TCP sender to track changes in the RTT more closely. In
particular, a TCP sender can react more quickly to sudden increases
of the RTT by sooner updating the RTO to a more conservative value.
The TCP Timestamps option [2] provides this capability, allowing the
TCP sender to sample the RTT from every segment that is acknowledged.
Using timestamps in the mentioned scenario leads to a more
conservative TCP retransmission timer and reduces the risk of
triggering spurious timeouts [45], [52], [54], [60].

There are two problematic issues with using timestamps:

o 12 bytes of overhead are introduced by carrying the TCP Timestamps
   option and padding in the TCP header. For a small MTU size, it
   can present a considerable overhead. For example, for an MTU of
   296 bytes the added overhead is 4%. For an MTU of 1500 bytes, the
   added overhead is only 0.8%.

o Current TCP header compression schemes are limited in their
   handling of the TCP options field. For RFC 2507 [13], any change
   in the options field (caused by timestamps or SACK, for example)
   renders the entire field uncompressible (leaving the TCP/IP header
   itself compressible, however). Even worse, for RFC 1144 [40] such
   a change in the options field effectively disables TCP/IP header
   compression altogether. This is the case when a connection uses
   the TCP Timestamps option. That option field is used both in the
   data and the ACK path, and its value typically changes from one

packet to the next. The IETF is currently specifying a robust
TCP/IP header compression scheme with better support for TCP
options [29].

The original definition of the timestamps option [2] specifies that
duplicate segments below cumulative ACK do not update the cached
timestamp value at the receiver. This may lead to overestimating of
RTT for retransmitted segments. A possible solution [47] allows the
receiver to use a more recent timestamp from a duplicate segment.
However, this suggestion allows for spoofing attacks against the TCP
receiver. Therefore, careful consideration is needed in
implementing this solution.

Recommendation: TCP SHOULD use the TCP Timestamps option. It allows
for better RTT estimation and reduces the risk of spurious timeouts.

4.9 Disabling RFC 1144 TCP/IP Header Compression (Wireless Host)

It is well known (and has been shown with experimental data) that RFC
1144 [40] TCP header compression does not perform well in the
presence of packet losses [43], [52]. If a wireless link error is
not recovered, it will cause TCP segment loss between the compressor
and decompressor, and then RFC 1144 header compression does not allow
TCP to take advantage of Fast Retransmit Fast Recovery mechanism.
The RFC 1144 header compression algorithm does not transmit the
entire TCP/IP headers, but only the changes in the headers of
consecutive segments. Therefore, loss of a single TCP segment on the
link causes the transmitting and receiving TCP sequence numbers to
fall out of synchronization.   Hence, when a TCP segment is lost
after the compressor, the decompressor will generate false TCP
headers. Consequently, the TCP receiver will discard all remaining
packets in the current window because of a checksum error. This
continues until the compressor receives the first retransmission
which is forwarded uncompressed to synchronize the decompressor [40].

As previously recommended in RFC 3150 [5], RFC 1144 header
compression SHOULD NOT be enabled unless the packet loss probability
between the compressor and decompressor is very low. Actually,
enabling the Timestamps Option effectively accomplishes the same
thing (see Section 4.8). Other header compression schemes like RFC
2507 [13] and Robust Header Compression [12] are meant to address
deficiencies in RFC 1144 header compression. At the time of this
writing, the IETF was working on multiple extensions to Robust Header
Compression (negotiating Robust Header Compression over PPP,
compressing TCP options, etc) [16].

4.10 Summary

```
   Items                            Comments
   -------------------------------------------------------------
   Appropriate Window Size          (sender & receiver)
                                    based on end-to-end BDP

   Window Scale Option              (sender & receiver)
   [RFC1323]                        Window size > 64KB

   Increased Initial Window         (sender)
   [RFC3390]                        CWND = min (4*MSS,
                                    max (2*MSS, 4380 bytes))

   Limited Transmit                 (sender)
   [RFC3042]

   IP MTU larger than               more applicable to 3G
   Default

   Path MTU Discovery               (sender & intermediate routers)
   [RFC1191,RFC1981]

   Selective Acknowledgment
   option (SACK)
   [RFC2018]                        (sender & receiver)

   Explicit Congestion
   Notification(ECN)
   [RFC3168]                        (sender, receiver &
                                    intermediate routers)

   Timestamps Option                (sender & receiver)
   [RFC1323, R.T.Braden's ID]

   Disabling RFC1144
   TCP/IP Header Compression
   [RFC1144]                        (wireless host)
```

5. Open Issues

   This section outlines additional mechanisms and parameter settings
   that may increase end-to-end performance when running TCP across
   2.5G/3G networks. Note, that apart from the discussion of the RTO's
   initial value, those mechanisms and parameter settings are not part
   of any standards track RFC at the time of this writing. Therefore,
   they cannot be recommended for the Internet in general.

Other mechanisms for increasing TCP performance include enhanced TCP/
IP header compression schemes [29], active queue management RFC 2309
[28], link layer retransmission schemes [23], and caching packets
during transient link outages to retransmit them locally when the
link is restored to operation [23].

Shortcomings of existing TCP/IP header compression schemes (RFC 1144
[40], RFC 2507 [13]) are that they do not compress headers of
handshaking packets (SYNs and FINs), and that they lack proper
handling of TCP option fields (e.g., SACK or timestamps) (see Section
4.8).   Although RFC 3095 [12] does not yet address this issue, the
IETF is developing improved TCP/IP header compression schemes,
including better handling of TCP options such as timestamps and
selective acknowledgements. Especially, if many short-lived TCP
connections run across the link, the compression of the handshaking
packets may greatly improve the overall header compression ratio.

Implementing active queue management is attractive for a number of
reasons as outlined in RFC 2309 [28]. One important benefit for
2.5G/ 3G networks, is that it minimizes the amount of potentially
stale data that may be queued in the network ("clicking from page to
page" before the download of the previous page is complete).
Avoiding the transmission of stale data across the 2.5G/3G radio link
saves transmission (battery) power, and increases the ratio of useful
data over total data transmitted. Another important benefit of
active queue management for 2.5G/3G networks, is that it reduces the
risk of a spurious timeout for the first data segment as outlined
below.

Since 2.5G/3G networks are commonly characterized by high delays,
avoiding unecessary round-trip times is particularly attractive.
This is specially beneficial for short-lived, transactional (request/
response-style) TCP sessions that typically result from browsing the
Web from a smart phone. However, existing solutions such as T/TCP
RFC 1644 [27], have not been adopted due to known security concerns
[38].

Spurious timeouts, packet re-ordering, and packet duplication may
reduce TCP's performance. Thus, making TCP more robust against those
events is desirable. Solutions to this problem have been proposed
[30], [35], [41], and standardization work within the IETF is ongoing
at the time of writing. Those solutions include reverting congestion
control state after such an event has been detected, and adapting the
retransmission timer and duplicate acknowledgement threshold. The
deployment of such solutions may be particularly beneficial when
running TCP across wireless networks because wireless access links
may often be subject to handovers and resource preemption, or the
mobile transmitter may traverse through a radio coverage hole. Such

disrupting events may easily trigger a spurious timeout despite a conservative retransmission timer. Also, the mobility mechanisms of some wireless networks may cause packet duplication.

The algorithm for computing TCP's retransmission timer is specified in RFC 2988 [11]. The standard specifies that the initial setting of the retransmission timeout value (RTO) should not be less than 3 seconds. This value might be too low when running TCP across 2.5G/3G networks. In addition to its high latencies, those networks may be run at bit rates of as low as about 10 kb/s which results in large packet transmission delays. In this case, the RTT for the first data segment may easily exceed the initial TCP retransmission timer setting of 3 seconds. This would then cause a spurious timeout for that segment. Hence, in such situations it may be advisable to set TCP's initial RTO to a value larger than 3 seconds. Furthermore, due to the potentially large packet transmission delays, a TCP sender might choose to refrain from initializing its RTO from the RTT measured for the SYN, but instead take the RTT measured for the first data segment.

Some of the recommendations in RFC 2988 [11] are optional, and are not followed by all TCP implementations. Specifically, some TCP stacks allow a minimum RTO less than the recommended value of 1 second (section 2.4 of [11]), and some implementations do not implement the recommended restart of the RTO timer when an ACK is received (section 5.3 of [11]). Some experiments [52], [54], have shown that in the face of bandwidth oscillation, using the recommended minimum RTO value of 1 sec (along with the also recommended initial RTO of 3 sec) reduces the number of spurious retransmissions as compared to using small minimum RTO values of 200 or 400 ms. Furthermore, TCP stacks that restart the retransmission timer when an ACK is received experience far less spurious retransmissions than implementations that do not restart the RTO timer when an ACK is received. Therefore, at the time of this writing, it seems preferable for TCP implementations used in 3G wireless data transmission to comply with all recommendations of RFC 2988.

6. Security Considerations

   In 2.5G/3G wireless networks, data is transmitted as ciphertext over the air and as cleartext between the Radio Access Network (RAN) and the core network. IP security RFC 2401 [37] or TLS RFC 2246 [36] can be deployed by user devices for end-to-end security.

7. IANA Considerations

   This specification requires no IANA actions.

8. Acknowledgements

   The authors would like to acknowledge contributions to the text from
   the following individuals:

      Max Hata, NTT DoCoMo, Inc. (hata@mml.yrp.nttdocomo.co.jp)

      Masahiro Hara, Fujitsu, Inc. (mhara@FLAB.FUJITSU.CO.JP)

      Joby James, Motorola, Inc. (joby@MIEL.MOT.COM)

      William Gilliam, Hewlett-Packard Company (wag@cup.hp.com)

      Alan Hameed, Fujitsu FNC, Inc. (Alan.Hameed@fnc.fujitsu.com)

      Rodrigo Garces, Mobility Network Systems
                         (rodrigo.garces@mobilitynetworks.com)

      Peter Ford, Microsoft (peterf@Exchange.Microsoft.com)

      Fergus Wills, Openwave (fergus.wills@openwave.com)

      Michael Meyer (Michael.Meyer@eed.ericsson.se)

   The authors gratefully acknowledge the valuable advice from the
   following individuals:

      Gorry Fairhurst (gorry@erg.abdn.ac.uk)

      Mark Allman (mallman@grc.nasa.gov)

      Aaron Falk (falk@ISI.EDU)

9. Normative References

   [1] Allman, M., Paxson, V. and W. Stevens, "TCP Congestion Control",
        RFC 2581, April 1999.

   [2] Jacobson, V., Braden, R. and D. Borman, "TCP Extensions for High
        Performance", RFC 1323, May 1992.

   [3] Mathis, M., Mahdavi, J., Floyd, S. and R. Romanow, "TCP
        Selective Acknowledgment Options", RFC 2018, October 1996.

   [4] Allman, M., Floyd, S. and C. Partridge, "Increasing TCP's
        Initial Window", RFC 3390, October 2002.

   [5] Dawkins, S., Montenegro, G., Kojo, M. and V. Magret, "End-to-end
        Performance Implications of Slow Links", BCP 48, RFC 3150, July
        2001.

   [6] Mogul, J. and S. Deering, "Path MTU Discovery", RFC 1191,
        November 1990.

   [7] Knowles, S., "IESG Advice from Experience with Path MTU
        Discovery", RFC 1435, March 1993.

   [8] McCann, J., Deering, S. and J. Mogul, "Path MTU Discovery for IP
        version 6", RFC 1981, August 1996.

   [9] Ramakrishnan, K., Floyd, S. and D. Black, "The Addition of
        Explicit Congestion Notification (ECN) to IP", RFC 3168,
        September 2001.

  [10] Allman, M., Balakrishnan, H. and S. Floyd, "Enhancing TCP's Loss
        Recovery Using Limited Transmit", RFC 3042, January 2001.

  [11] Paxson, V. and M. Allman, "Computing TCP's Retransmission
        Timer", RFC 2988, November 2000.

  [12] Bormann, C., Burmeister, C., Degermark, M., Fukushima, H.,
        Hannu, H., Jonsson, L-E., Hakenberg, R., Koren, T., Le, K., Liu,
        Z., Martensson, A., Miyazaki, A., Svanbro, K., Wiebke, T.,
        Yoshimura, T. and H. Zheng, "RObust Header Compression (ROHC):
        Framework and four profiles: RTP, UDP, ESP, and uncompressed",
        RFC 3095, July 2001.

  [13] Degermark, M., Nordgren, B. and S. Pink, "IP Header
        Compression", RFC 2507, February 1999.

  [14] Postel, J., "Transmission Control Protocol - DARPA Internet
        Program Protocol Specification", STD 7, RFC 793, September 1981.

  [15] Floyd, S. and T. Henderson, "The NewReno Modification to TCP's
        Fast Recovery Algorithm", RFC 2582, April 1999.

  [16] Bormann, C., "Robust Header Compression (ROHC) over PPP", RFC
        3241, April 2002.

  [17] Braden, R., "Requirements for Internet Hosts - Communication
        Layers", STD 3, RFC 1122, October 1989.

  [18] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6)
        Specification", RFC 2460, December 1998.

10. Informative References

   [19]   Montenegro, G., Dawkins, S., Kojo, M., Magret, V. and N.
          Vaidya, "Long Thin Networks", RFC 2757, January 2000.

   [20]   Third Generation Partnership Project, "RLC Protocol
          Specification (3G TS 25.322:)", 1999.

   [21]   Fall, K. and S. Floyd, "Simulation-based Comparisons of Tahoe,
          Reno, and SACK TCP", Computer Communication Review, 26(3) , July
          1996.

   [22]   Fairhurst, G. and L. Wood, "Advice to link designers on link
          Automatic Repeat reQuest (ARQ)", BCP 62, RFC 3366, August 2002.

   [23]   Karn, P., "Advice for Internet Subnetwork Designers", Work in
          Progress.

   [24]   Dawkins, S., Montenegro, G., Magret, V., Vaidya, N. and M.
          Kojo, "End-to-end Performance Implications of Links with
          Errors", BCP 50, RFC 3135, August 2001.

   [25]   Wireless Application Protocol, "WAP Specifications", 2002,
          <http://www.wapforum.org>.

   [26]   Open Mobile Alliance, "Open Mobile Alliance", 2002,
          <http://www.openmobilealliance.org/>.

   [27]   Braden, R., "T/TCP -- TCP Extensions for Transactions", RFC
          1644, July 1994.

   [28]   Braden, R., Clark, D., Crowcroft, J., Davie, B., Deering, S.,
          Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge,
          C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J.
          and L. Zhang, "Recommendations on Queue Management and
          Congestion Avoidance in the Internet", RFC 2309, April 1998.

   [29]   IETF, "Robust Header Compression", 2001,
          <http://www.ietf.org/html.charters/rohc-charter.html>.

   [30]   Ludwig, R. and R. H. Katz, "The Eifel Algorithm: Making TCP
          Robust Against Spurious Retransmissions", ACM Computer
          Communication Review 30(1), January 2000.

   [31]   Wireless Application Protocol, "WAP Wireless Profiled TCP",
          WAP-225-TCP-20010331-a, April 2001,
          <http://www.wapforum.com/what/technical.htm>.

[32]   Hadi Salim, J. and U. Ahmed, "Performance Evaluation of Explicit
       Congestion Notification (ECN) in IP Networks", RFC 2884, July
       2000.

[33]   NTT DoCoMo Technical Journal, "Special Issue on i-mode Service",
       October 1999.

[34]   NTT DoCoMo Technical Journal, "Special Article on IMT-2000
       Services", September 2001.

[35]   Floyd, S., Mahdavi, J., Mathis, M. and M. Podolsky, "An
       Extension to the Selective Acknowledgement (SACK) Option for
       TCP", RFC 2883, July 2000.

[36]   Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC
       2246, January 1999.

[37]   Kent, S. and R. Atkinson, "Security Architecture for the
       Internet Protocol", RFC 2401, November 1998.

[38]   de Vivo, M., O. de Vivo, G., Koeneke, R. and G. Isern, "Internet
       Vulnerabilities Related to TCP/IP and T/TCP", ACM Computer
       Communication Review 29(1), January 1999.

[39]   Third Generation Partnership Project, "RRC Protocol
       Specification (3GPP TS 25.331:)", September 2001.

[40]   Jacobson, V., "Compressing TCP/IP Headers for Low-Speed Serial
       Links", RFC 1144, February 1990.

[41]   Blanton, E. and M. Allman, "On Making TCP More Robust to Packet
       Reordering", ACM Computer Communication Review 32(1), January
       2002, <http://roland.grc.nasa.gov/~mallman/papers/tcp-reorder-
       ccr.ps>.

[42]   Karn, P. and C. Partridge, "Improving Round-Trip Time Estimates
       in Reliable Transport Protocols", ACM SIGCOMM 87, 1987.

[43]   Ludwig, R., Rathonyi, B., Konrad, A. and A. Joseph, "Multi-layer
       tracing of TCP over a reliable wireless link", ACM SIGMETRICS
       99, May 1999.

[44]   Ludwig, R., Konrad, A., Joseph, A. and R. Katz, "Optimizing the
       End-to-End Performance of Reliable Flows over Wireless Links",
       Kluwer/ACM Wireless Networks Journal Vol. 8, Nos. 2/3, pp. 289-
       299, March-May 2002.

[45] Gurtov, A., "Making TCP Robust Against Delay Spikes", University
      of Helsinki, Department of Computer Science, Series of
      Publications C, C-2001-53, Nov 2001,
      <http://www.cs.helsinki.fi/u/gurtov/papers/report01.html>.

[46] Stevens, W., "TCP/IP Illustrated, Volume 1; The Protocols",
      Addison Wesley, 1995.

[47] Braden, R., "TCP Extensions for High Performance: An Update",
      Work in Progress.

[48] Allman, M., Dawkins, S., Glover, D., Griner, J., Tran, D.,
      Henderson, T., Heidemann, J., Touch, J., Kruse, H., Ostermann,
      S., Scott, K. and J. Semke, "Ongoing TCP Research Related to
      Satellites", RFC 2760, February 2000.

[49] Allman, M., Glover, D. and L. Sanchez, "Enhancing TCP Over
      Satellite Channels using Standard Mechanisms", BCP 28, RFC 2488,
      January 1999.

[50] Balakrishnan, H., Padmanabhan, V., Fairhurst, G. and M.
      Sooriyabandara, "TCP Performance Implications of Network
      Asymmetry", RFC 3449, December 2002.

[51] Kempf, J., "Problem Description: Reasons For Performing Context
      Transfers Between Nodes in an IP Access Network", RFC 3374,
      September 2002.

[52] Khafizov, F. and M. Yavuz, "Running TCP over IS-2000", Proc. of
      IEEE ICC, 2002.

[53] Khafizov, F. and M. Yavuz, "Analytical Model of RLP in IS-2000
      CDMA Networks", Proc. of IEEE Vehicular Technology Conference,
      September 2002.

[54] Yavuz, M. and F. Khafizov, "TCP over Wireless Links with
      Variable Bandwidth", Proc. of IEEE Vehicular Technology
      Conference, September 2002.

[55] TIA/EIA/cdma2000, "Mobile Station - Base Station Compatibility
      Standard for Dual-Mode Wideband Spread Spectrum Cellular
      Systems", Washington: Telecommunication Industry Association,
      1999.

[56] TIA/EIA/IS-95 Rev A, "Mobile Station - Base Station
      Compatibility Standard for Dual-Mode Wideband Spread Spectrum
      Cellular Systems", Washington: Telecommunication Industry
      Association, 1995.

   [57] TIA/EIA/IS-707-A-2.10, "Data Service Options for Spread Spectrum
         Systems: Radio Link Protocol Type 3", January 2000.

   [58] Dahlman, E., Beming, P., Knutsson, J., Ovesjo, F., Persson, M.
         and C. Roobol, "WCDMA - The Radio Interface for Future Mobile
         Multimedia Communications", IEEE Trans. on Vehicular Technology,
         vol. 47, no. 4, pp. 1105-1118, November 1998.

   [59] Allman, M. and V. Paxson, "On Estimating End-to-End Network Path
         Properties", ACM SIGCOMM 99, September 1999.

   [60] Gurtov, A. and R. Ludwig, "Responding to Spurious Timeouts in
         TCP", IEEE INFOCOM'03, March 2003.

11. Authors' Addresses

    Hiroshi Inamura
    NTT DoCoMo, Inc.
    3-5 Hikarinooka
    Yokosuka Shi, Kanagawa Ken 239-8536
    Japan

    EMail: inamura@mml.yrp.nttdocomo.co.jp
    URI: http://www.nttdocomo.co.jp/

    Gabriel Montenegro
    Sun Microsystems Laboratories, Europe
    Avenue de l'Europe
    ZIRST de Montbonnot
    38334 Saint Ismier CEDEX
    France

    EMail: gab@sun.com

    Reiner Ludwig
    Ericsson Research
    Ericsson Allee 1
    52134 Herzogenrath
    Germany

    EMail: Reiner.Ludwig@Ericsson.com

    Andrei Gurtov
    Sonera
    P.O. Box 970, FIN-00051
    Helsinki,
    Finland

    EMail: andrei.gurtov@sonera.com
    URI: http://www.cs.helsinki.fi/u/gurtov/

    Farid Khafizov
    Nortel Networks
    2201 Lakeside Blvd
    Richardson, TX 75082,
    USA

    EMail: faridk@nortelnetworks.com

12. Full Copyright Statement

   Copyright (C) The Internet Society (2003). All Rights Reserved.

   This document and translations of it may be copied and furnished to
   others, and derivative works that comment on or otherwise explain it
   or assist in its implementation may be prepared, copied, published
   and distributed, in whole or in part, without restriction of any
   kind, provided that the above copyright notice and this paragraph are
   included on all such copies and derivative works. However, this
   document itself may not be modified in any way, such as by removing
   the copyright notice or references to the Internet Society or other
   Internet organizations, except as needed for the purpose of
   developing Internet standards in which case the procedures for
   copyrights defined in the Internet Standards process must be
   followed, or as required to translate it into languages other than
   English.

   The limited permissions granted above are perpetual and will not be
   revoked by the Internet Society or its successors or assigns.

   This document and the information contained herein is provided on an
   "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
   TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
   BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
   HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
   MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

   Funding for the RFC Editor function is currently provided by the
   Internet Society.