         Delay Limits and Multiplexing Policies to be employed with Tunneling
              Compressing and Multiplexing Traffic Flows
                    draft-suznjevic-tsvwg-mtd-tcmtf-05

Abstract

   This document contains recommendations to be taken into account when
   using methods which optimize bandwidth utilization through Tunneling
   Compressing and Multiplexing (TCM) traffic flows over a network path.
   Different multiplexing policies and implementation issues which are
   service and link specific are discussed. Additionally, this document
   describes policies which can be used for detecting, classifying, and
   choosing the network flows suitable for optimization by using TCM.
   Finally, recommendations of maximum tolerable delays to be added by
   optimization techniques are reported. Recommendations are presented
   only for network services for which such bandwidth optimization
   techniques are applicable (i.e., services with low payload to header
   size ratio, which will also be denoted as "small-packet flows").

Status of This Memo

   This Internet-Draft is submitted to IETF in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on December 15, 2015.

Copyright Notice

Table of Contents

1. Introduction

   This document extends the draft [TCM] with a set of recommendations
   regarding the processes of traffic optimization, which may include
   compressing, multiplexing, and/or tunneling a number of packets.
   These recommendations are needed because these traffic optimization
   techniques, while saving bandwidth and reducing overhead, may cause
   network impairments if packets are delayed before being sent
   together. These techniques are also proposed at layer 2. For
   example, in [IEEE.802-11N.2009], a number of Protocol Data Units can
   be grouped and transmitted together.

   Network delay is one of the main factors which can degrade the
   Quality of Experience (QoE) of real-time network services RFC 6390

[RFC6390] [TGPP_TR26.944]. In order to prevent the perceived quality degradation of the services when using TCM, a policy defining a multiplexing period can be employed.

First, the document describes different multiplexing policies which can be employed for defining which native packets are multiplexed together. A policy combining a multiplexing period and a packet size limit is proposed in order to put an upper bound on the added delay.

Additionally, this document describes the policies that can be employed for detecting, classifying, and choosing the network flows suitable for TCM optimization.

Finally, values for maximum tolerable delays presented here from the base of the proposed multiplexing policy. The recommendations are presented for both real-time and non real-time network services in which TCM bandwidth optimization is applicable (i.e., services which have low payload-to-header-size ratio, which results in high protocol overhead, which will also be denoted as small-packet flows).

1.1. Requirements Language

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 RFC 2119 [RFC2119].

2. Terminology

This document uses a number of terms to refer to the roles played by participants in, and objects of, the TCM sessions.

TCM optimizer

The host where TCM optimization is deployed. If that hosts corresponds to the ingress of the tunnel where native packets are included it is called TCM-ingress optimizer (TCM-IO).

The host where TCM multiplexed packets are received and rebuilt to their native form is called TCM-egress optimizer (TCM-IO). It corresponds to the tunnel egress.

policy manager

A network entity which makes the decisions about TCM parameters: multiplexing period; flows to be multiplexed together, depending on their IP addresses, ports, etc. It is connected with a number of TCM optimizers, and orchestrates the optimization that takes place between them.

native packet

A packet sent by an application, belonging to a flow that can be optimized by means of TCM.

TCM-optimized packet

A packet including a number of multiplexed and header-compressed native ones, and also a tunneling header shared by all the packets, as detailed by TCM.

3. Considered services

The services considered suitable for being optimized by TCM are those that generate long-term flows of small packets, with a low payload to header size ratio. Some real-time and some non real-time services are suitable for optimization by means of TCM.

3.1. Real-time services

Under the term "real-time network services" we consider both conversational and streaming service classes as defined in [TGPP_TS]. Interactive and background services are considered non real-time. Fundamental requirements of real-time network services include conversational pattern (stringent and low delay) and preservation of the time relation (variation) between the information entities of the stream.

We identify the following real-time network services with low payload to header size ratio as candidates for the bandwidth optimization techniques presented in TCM:

o Voice over IP

o Online games

o Remote desktop services

While video services are considered real-time, they are not suitable for bandwidth optimization techniques proposed in [TCM], due to their high payload to header size ratio. Due to the same reason, we do not take into account cloud gaming services. In such gaming services all the calculations of the game state are deployed at the server and a real-time video stream is sent to the client. In these cases, TCM optimization is neither interesting nor applicable.

3.2. Non real-time services

   On the other hand, TCM can be applied for some non real-time services
   such as streaming audio, and instant messaging. These applications
   are suitable for TCM in terms of payload to header size ratio, but
   different studies have shown that acceptable delays for these
   services are up to several seconds [ITU-T_G.1010]. Also, some types
   of machine to machine (M2M) traffic (e.g., metering messages from
   various sensors) may have traffic properties suitable for TCM.
   Acceptable delays for these services can be go up to an hour
   [Liu_M2M]. We list limitations for these services as well, although
   in the practical application TCM should not introduce delays which
   would be noticeable in comparison with delays of such magnitude
   (i.e., seconds and more).

4. Multiplexing policies in TCM

   A multiplexing policy defines the decision process for determining
   which native packet goes in which multiplexed packet. The policies
   proposed for TCM are:

   o Fixed number of packets - once a fixed number of packets (N) has
      arrived, a multiplexed packet is created and sent.

   o Size limit - once a size limit is reached (e.g., next to the MTU
      of the underlying network), a multiplexed packet is sent.

   o Period - a multiplexed packet is sent every time period T.

   o Timeout - sends a multiplexed packet if a native one arrives and
      the time since the last multiplexed packet departure is above a
      defined timeout value.

   Only the two latter policies are able to control the additional delay
   introduced by multiplexing. In addition, different policies can be
   combined.

   In this document we focus on the combination of "size limit" and
   "period" policies, as shown in Figure 1. A multiplexed packet is
   sent at the end of each "period". However, if the size limit is
   reached, then a multiplexed packet is sent immediately, and the
   period is "reset". Thus, the added delay is for the worst case
   scenario equal to the defined period.

native traffic:

```
|<--P-->|<--P-->|<--P-->|<--P-->|<-t*->|<--P-->|
|       |       |       |       |      |       |
|#    # | #   # |       |    #  #|######| #    #|
-------------------------------------------------> t
```

multiplexed traffic:

```
|       |       |       |       |      |       |
|       |#      |#      |       |#     |##     |#
|       |#      |#      |       |#     |##     |#
-------------------------------------------------> t
```
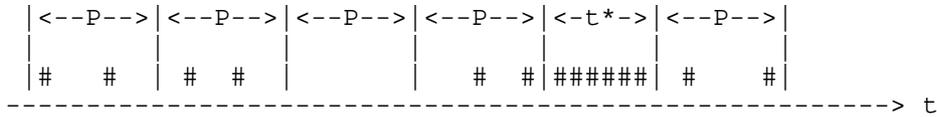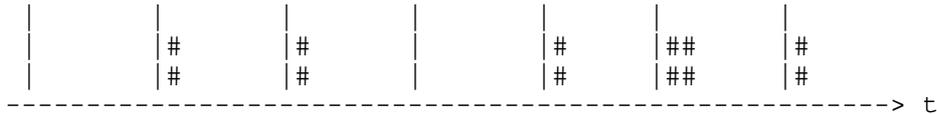
      * period reset (t<P) because size limit is reached

           Combined "period" and "size limit" policies

                             Figure 1

   It should be noted that the number of aggregated flows and their
   packet rate will have an influence on the average multiplexing delay
   added. If the number of flows is high, then the MTU size will be
   reached before the end of the period in most cases, so the additional
   delay will be smaller than the period. The recommendations presented
   in this document present the maximum period values to be used as a
   limit, in order to avoid delays which could impair the quality of the
   service.

5. Detecting, classifying, and choosing network flows to be optimized

   Three basic issues need to be solved in order to employ TCM
   optimization. First, the flows which are candidates for optimization
   need to be detected from the overall traffic mix. Secondly, the
   flows need to be classified into one of the defined categories so an
   adequate multiplexing period can be assigned. Finally, the decision
   whether a specific flow will be optimized or not using TCM needs to
   be made.

5.1. Optimization within an administrative domain

   Several scenarios can be considered for the use of TCM. If the
   optimization is deployed within an administrative domain, then all
   the data of the end hosts, the service class, etc., are known by the
   TCM optimizers.

   Two examples of this are 1) the end-to-end optimization and
   aggregation of a number of flows between two offices of the same

company and 2) the agreement between a network operator and a game
provider in order to multiplex all the packets generated in an
aggregation network with destination on a game server. In these
cases, the detection and classification of the desired flows will be
straightforward, since the TCM optimizer can simply inspect the
destination IP address and port, and apply the traffic category
according to the kind of service.

5.2. Optimization based on statistics

   If the optimization is not performed within an administrative domain,
   then the detection and classification of the flows, and the decision
   about multiplexing them, will have to be based on statistics of the
   traffic and heuristics. The intelligence of the flow identification
   method can be improved according to the statistics of already
   classified flows. E.g., if a number of small-packet flows sharing
   the same IP destination address are found, then this destination host
   can be classified as a frequent receiver of small-packet flows, and a
   tunnel including all the packets addressed to it can be set within a
   common network path.

   In addition, statistics can be enriched by the assignment of the
   traffic class, taking into account that some services, in addition to
   well-known ports, also have well-known IP addresses. E.g., the
   traffic travelling to the IP address of a popular online game server,
   can be associated with the proper traffic class; or the ports
   corresponding to certain services can also be identified and used in
   order to improve the classification.

   The detection of the flows candidates for TCM optimization should be
   done based on flow characteristics, primarily on the packet payload
   to header ratio and on the packet rate. As these properties cannot
   be established from statistics of just one packet, it is needed to
   gather a certain number of packets for each new flow arriving at the
   TCM optimizer, and to use some heuristics in order to decide whether
   to multiplex a certain flow or not.

   The classification method employed for the TCM needs to identify only
   the flows which are candidates for the TCM optimization. Therefore,
   the classification problem is significantly simplified by removal of
   peer to peer (P2P) downloading applications, video streaming, data
   transfer, and all other services which in general, utilize large
   packets. This is especially important as P2P applications are known
   to use various non assigned ports which may greatly ruin the
   performance of simple traffic classification methods. For the
   purposes of TCM optimization there is no need to identify a
   particular application, only the delay sensitive class in which that
   application falls. Also, the traffic classification methods employed

by TCM need to be able to assign flows to respective delay sensitive
classes in real time. Current network traffic classification methods
can be grouped into [Nguyen_TCSurvey]:

o Port based - through lookup of port numbers of endpoints in the
   Internet Assigned Numbers Authority (IANA)'s list of registered
   ports.

o Payload based - through stateful reconstruction of session and
   application information from each packet's content.

o Statistical - through comparison of the statistical properties of
   the traffic at the network layer.

While payload inspection does give the best results, and is often
used as ground truth in classification of network traffic, it is
demanding computation wise. Also, these techniques may be
interpreted as a violation of privacy. For the purposes of TCM we
recommend using a combination of port based classification (which can
yield very good results for games based on a client-server
architecture and remote desktop services), and inspection of
statistical properties of the flows. One such algorithm has been
employed for classification of different types of game flows and
showed good results [Han_GameClassification]. TCM should use
metadata information regarding the traffic class of particular flow
where such information is available as that significantly simplifies
the detection and classification problem.

The decision whether the flow should be optimized with TCM can be
made based on the following policies (configurations of the
multiplexing node):

o A static configuration - predefined rule set for each new
   occurring flow, so the TCM optimizer makes a decision.

o A policy manager which dynamically enforces the rule set.

o The TCM optimizer demands instructions for each new flow from the
   policy manager.

6. Delay recommendations

The three normally considered network impairments in the studies
related to subjective quality in real-time interactive games are:

o delay - can be reported as one-way-delay (OWD) [RFC2679] and two-
   way-delay (Round Trip Time) [RFC2681]. In this document, under
   the term latency, one way end-to-end delay is considered.

   o delay variation - which is a statistical variance of the data
      packet inter-arrival time, in other words the variation of the
      delay as defined in RFC 3393 [RFC3393].

   o packet loss - more important for certain applications, while other
      include very good algorithms for concealing it (e.g., some game
      genres).

   In this document we give recommendations for overall tolerable delays
   to be taken into account when optimizing network services by means of
   TCM. In an interactive service, the total delay is composed by the
   addition of delays as defined in 3GPP TR 26.944 [TGPP_TR26.944].

   o Transfer delay - from Host1 to Host2 at time T is defined by the
      statement: Host1 sent the first bit of a unit data to Host2 at
      wire-time T and that Host2 received the last bit of that packet at
      wire-time T+dT. Thus, it includes the transmission delay (the
      amount of time Host1 requires to push all of the packet's bits
      into the wire) and the propagation delay in the network (the
      amount of time it takes for the head of the packet to travel from
      Host1 to Host2).

   o Transaction delay - the sum of the time for a data packet to wait
      in queue and receive the service during the server transaction.
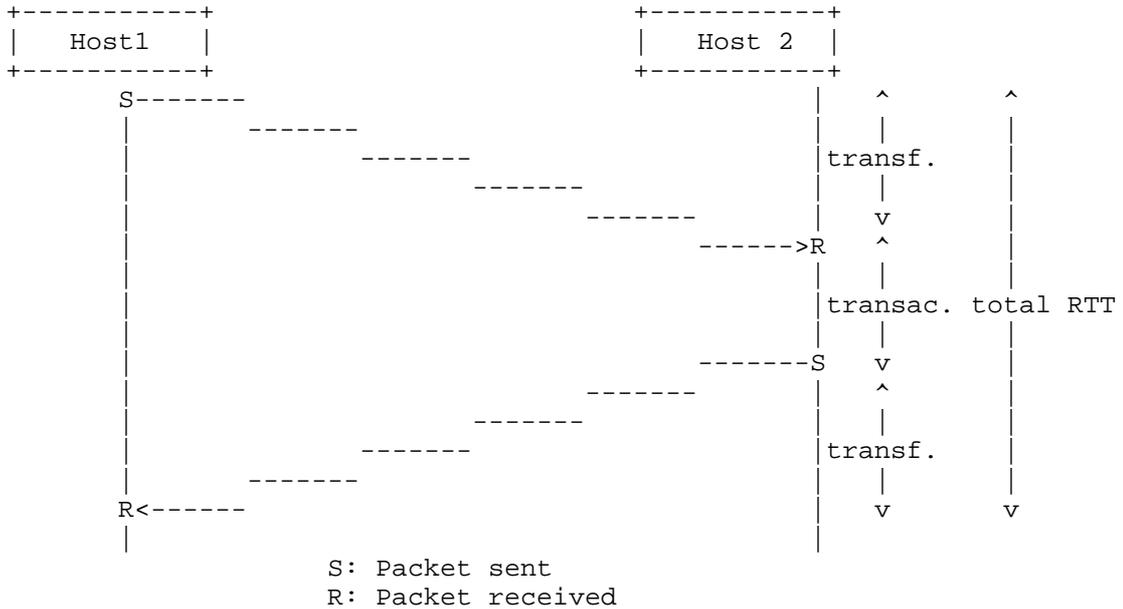
```
   +-----------+                        +-----------+
   |   Host1   |                        |  Host 2   |
   +-----------+                        +-----------+
        S-------                              |  ^        ^
        |      -------                        |  |        |
        |            -------                  |transf.    |
        |                  -------            |  |        |
        |                        -------      |  v        |
        |                              ------>R  ^        |
        |                                     |  |        |
        |                                     |transac. total RTT
        |                                     |  |        |
        |                              -------S  v        |
        |                        -------      |  ^        |
        |                  -------            |  |        |
        |            -------                  |transf.    |
        |      -------                        |  |        |
        R<------                              |  v        v
        |                                     |
                  S: Packet sent
                  R: Packet received


                          Figure 2
```

Figure 2 illustrates these delays. The labeled times (S and R)
designate the times in which the packet is sent and received,
respectively, by the network card interface.

The use of TCM requires the addition of TCM optimizers in the
scenario. A number of flows are multiplexed together before being
sent through the network. The packets are demultiplexed and rebuilt
before being forwarded to the application server. A scheme of TCM is
included in Figure 3:

```
 +--------+
 |client 1|___
 +--------+   \
              \                _  _
 +--------+    +--------+    ( `   )_           +--------+   +------+
 |client 2|--->| TCM-IO |--> (      )     `) --->| TCM-EO |-->|server|
 +--------+    +--------+   (_   (_ .  _) _)  +--------+   +------+
             /                  Internet
            /
 +--------+   /      <--------------TCM----------->
 |client n|_/
 +--------+
```
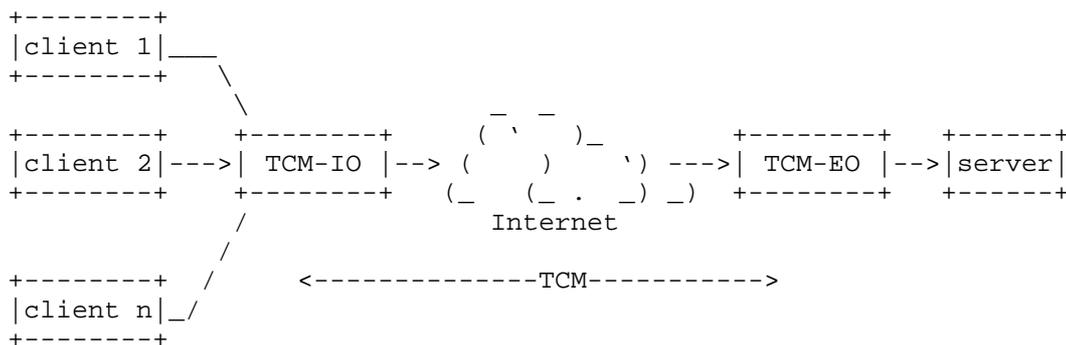
Figure 3

This technique groups packets in order to build a multiplexed one.
As previously stated, the focus of this document is on "multiplexing
period" policy for creating the multiplexed packet combined with size
limit policy. Multiplexing period is a time frame in which the TCM
optimizer waits for native packets to arrive in order to send them as
one multiplexed packet. If the multiplexed packet size limit is
reached before the multiplexing period has run out (i.e., if enough
native packets arrive to fill the limit), the multiplexed packet is
sent right away. In this way a certain amount of delay caused by the
TCM optimization is added in the communication. It is important to
emphasize that multiplexing delay can't exceed the multiplexing
period, and that it will only reach the value of multiplexing period
on a link with a low traffic load. Multiplexing delay can be
classified as caused by the middlebox presence as defined in RFC 6390
RFC 6390 [RFC6390]. The delay in the TCM-IO includes the time during
which the packets are retained until the bundled packet is sent, plus
processing time. In the TCM-EO however, the packets are not
retained, so only the processing time is considered.

Figure 4 shows the total delay, when a TCM optimizers are added. It
should be noted that multiplexing can be deployed independently in
both directions, or only in one of them.

```
     +---------+     +---------+       +---------+     +---------+
     | Host 1  |     | TCM-IO  |       | TCM-EO  |     | Host 2  |
     +---------+     +---------+       +---------+     +---------+
         S-------        |                 |             |   ^         ^
         |      -------  |                 |             |   |         |
         |           --R     ^             |             |   |         |
         |             |     |             |             |   |         |
         |             |    mux            |             |   |         |
         |             |     |             |             |   |         |
         |           | |     |             |             |   |         |
         |           S---- v |             |             | transf.     |
         |           |     -------         |             | (& mux)     |
         |           |          ------R     ^            |   |         |
         |           |               |    demux          |   |      total
         |           |               |     |             |   |       RTT
         |           |               S--- v |            |   v         |
         |           |               |    --------->R   ^            |
         |           |               |                 |   |         |
         |           |               |                 |transac.     |
         |           |               -------S           v            |
         |           |          --------       |         ^            |
         |           |     --------             |         |           |
         |       --------                       | transf. |           |
         |    -------                           |   |     |           |
         R<--------                             |   v     v           v
         |                                      |
                      S: Packet sent
                      R: Packet received
```
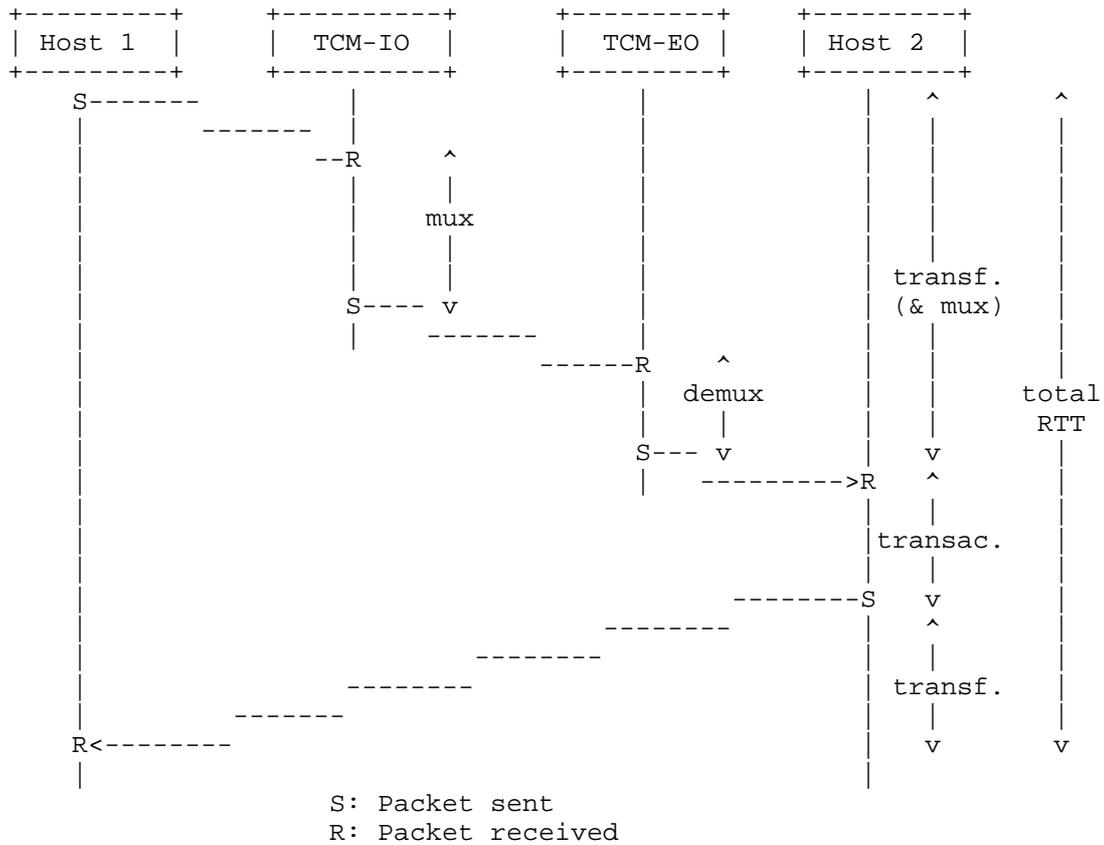
                            Figure 4

   With respect to efficiency in terms of use of the bandwidth, a
   tradeoff appears: the longer the multiplexing period, the higher the
   number of packets which can be grouped, thus obtaining better
   bandwidth savings. So in order to calculate the maximum multiplexing
   period, the rest of the delays have to be considered: if the sum of
   transaction, and transfer delays is under the maximum tolerable
   delay, then multiplexing will be possible without harming the user
   experience. The overall delay may be calculated according to the
   ITU-T Y.1541 recommendation [ITU-T_Y.1541]. Subtracting propagation,
   processing, and transmission delay from the tolerable delay for
   specific service results in the maximum value of the multiplexing
   period.

   Next, we will report the maximum tolerable latency for the previously
   listed real-time network services.

6.1. VoIP

   For conversational audio, the International Telecommunication Union
   recommends [ITU-T_G.114] less than 150 millisecond one-way end-to-end
   delay for high-quality real time traffic, but delays between 150 ms
   and 400 ms are acceptable. When considering conversational audio it
   should be noted that this delay limits include jitter buffers and
   codec processing. For streaming audio, delay constraints are much
   looser, the delay should be less than 10 s [ITU-T_G.1010]. Tunneling
   Multiplexed Compressed RTP (TCRTP) [RFC4170] already considers
   tunneling, compressing and multiplexing VoIP packets.

6.2. Online games

   Online games comprise game genres which have different latency
   requirements. This draft focuses on real-time online games and
   endorses the general game categorization proposed in
   [Claypool_Latency] in which online games have been divided into:

   o Omnipresent, with the threshold of acceptable latency (i.e.,
      latency in which performance is above 75% of the unimpaired
      performance) of 1000 ms. The most representative genre of
      omnipresent games are Real-Time Strategies.

   o Third Person Avatar, with the threshold of acceptable latency of
      500 ms. These games include Role Playing Games (RPG) and
      Massively Multiplayer Online Role-Playing Games (MMORPG).

   o First Person Avatar, in which threshold of acceptable latency is
      100 ms. The most popular subgenre of them are First Person
      Shooters, such as "Call of Duty" or "Halo" series.

   As remarked in [Bernier_Latency] and [Oliveira_online], different
   methods can be employed to combat delay in online games. The so-
   called "client-side prediction" has been largely used in First Person
   Shooters. It can be divided into "input prediction" and "dead
   reckoning," where input prediction hides the latency for the client-
   controlled actions while dead reckoning hides the latency of other
   participating players.

   The study [Claypool_Latency] evaluated players' performance in
   certain tasks, while increasing latency, and reported values at which
   the performance dropped below 75% of the performance under unimpaired
   network conditions. While measuring objective performance metrics,
   this method highly underestimates the impact of delays on players'
   QoE. Further studies accessing a particular game genre reported much
   lower latency thresholds for unimpaired gameplay.

Other approach some studies have taken is to perform "objective
measurements" [Kaiser_objective] a number of identical "bots", i.e.
virtual avatars controlled by Artificial Intelligence, are placed in
the same virtual scenario and a number of parties between them are
performed. If the number of parties if high enough, then the score
will be the same for all the bots. Then, different network
impairments (latency, jitter, packet loss) are added to one of the
bots, and another set of tests is performed. The performance
degradation of the network-impaired bot can then be statistically
characterized.

A survey using a large number of First Person Shooter games has been
carried out in [Dick_Analysis]. They state that latency about 80 ms
could be considered as acceptable, since the games have been rated as
"unimpaired". Besides service QoE, it has been shown that delay has
great impact on the user's decision to join a game, but significantly
less on the decision to leave the game [Henderson_QoS].

A study on Mean Opinion Score (MOS) evaluation, based on variation of
delay and jitter for MMORPGs, suggested that MOS drops below 4 for
delays greater than 120 ms [Ries_QoEMMORPG]. The MOS score of 5
indicates excellent quality, while MOS score of 1 indicates bad
quality. Another study focused on extracting the duration of play
sessions for MMORPGs from the network traffic traces showed that the
session durations start to decline sharply when round trip time is
between 150 ms and 200 ms [Chen_HowSensitive].

While original classification work [Claypool_Latency] states that
latency up to 1 second is tolerated by omnipresent games, other
studies argued that only latency up to 200 ms is tolerated by players
of RTS games [Cajada_RTS].

6.3. Remote desktop access

For the remote computer access services, the delays are dependent on
the task performed through the remote desktop. Tasks may include
operations with audio, video and data (e.g., reading, web browsing,
document creation). A QoE study indicates that for audio latency
below 225 ms and for data latency below 200 ms is tolerated
[Dusi_Thin].

6.4. Non real-time service

Traffic flows of several types of non real-time services can be
optimized using TCM. Under this category we include services for M2M
metering information, streaming audio, and instant messaging. M2M
metering services are suitable for TCM optimization not only due to
their very loose delay requirements, but also because of the one way

nature of the communication (i.e., most information travels from
sensors to the central server) [Liu_M2M]. The signalling information
related to M2M can also be optimized. Internet of Things application
layer protocols such as CoAP RFC 7252 [RFC7252], used in Constrained
RESTful Environments (CoRE)[RFC6690], work over UDP and send small
packets. The ACK_TIMEOUT period in CoAP is set to 2 seconds.
Instant messaging (despite "instant" in its name) has been
categorized as data service by the ITU-T, and it has been designated
with acceptable delays of up to a few seconds [ITU-T_G.1010].

6.5. Summary

We group all the results in the Table 1 indicating the maximum
allowed latency and proposed multiplexing periods. Proposed
multiplexing periods are guidelines, since the exact values are
dependant of the existing delay in the network. It should be noted
that reported tolerable latency is based on values of preferred
delays, and delays in which QoE estimation is not significantly
degraded. Multiplexing periods of about 1 second can be considered
as sufficient for non real-time services (e.g., streaming audio).

Service	Tolerable latency (OWD)	Mux. period
Voice communication	< 150ms	< 30ms
Omnipresent games	< 200ms	< 40ms
First person avatar games	< 80ms	< 15ms
Third person avatar games	< 120ms	< 25ms
Remote desktop	< 200ms	< 40ms
Instant messaging	< 5s	< 1s
M2M (metering)	< 1hour	< 1s

Table 1: Final recommendations

7. Acknowledgements

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

   No relevant security considerations have been identified

10. References

10.1. Normative References

   [IEEE.802-11N.2009]
            "Information technology - Telecommunications and
            information exchange between systems - Local and
            metropolitan area networks - Specific requirements - Part
            11: Wireless LAN Medium Access Control (MAC) and Physical
            Layer (PHY) specifications - Amendment 5: Enhancements for
            higher throughput", IEEE Standard 802.11n, Oct 2009,
            <http://standards.ieee.org/getieee802/
            download/802.11n-2009.pdf>.

   [ITU-T_G.1010]
            International Telecommunication Union-Telecommunication,
            "End-user multimedia QoS categories", SERIES G:
            TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND
            NETWORKS; Quality of service and performance , 2001.

   [ITU-T_G.114]
            ITU-T, "ITU-T Recommendation G.114 One-way transmission
            time", ITU G.114, 2003.

   [ITU-T_Y.1541]
            International Telecommunication Union-Telecommunication,
            "; Network performance objectives for IP-based services",
            SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET
            PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS; Internet
            protocol aspects - Quality of service and network
            performance , 2011.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
            Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way
            Delay Metric for IPPM", RFC 2679, September 1999.

   [RFC2681] Almes, G., Kalidindi, S., and M. Zekauskas, "A Round-trip
            Delay Metric for IPPM", RFC 2681, September 1999.

   [RFC3393] Demichelis, C., Chimento, S., and P. Zekauskas, "IP Packet
            Delay Variation Metric for IP Performance Metrics (IPPM)",
            RFC 3393, November 2002.

   [RFC4170] Thompson, B., Koren, T., and D. Wing, "Tunneling
              Multiplexed Compressed RTP (TCRTP)", RFC 6690, November
              2005.

   [RFC6390] Clark, A. and B. Claise, "Guidelines for Considering New
              Performance Metric Development", RFC 6390, October 2011.

   [RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link
              Format", RFC 6690, August 2012.

   [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained
              Application Protocol (CoAP)", RFC 7252, June 2014.

10.2. Informative References

   [Bernier_Latency]
              Bernier, Y., "Latency Compensating Methods in Client/
              Server In-Game Protocol Design and Optimization", Proc.
              Game Developers Conference, San Jose Vol. 98033. No. 425.,
              2001.

   [Cajada_RTS]
              Cajada, M., "VFC-RTS: Vector-Field Consistency para Real-
              Time-Strategy Multiplayer Games", Master of Science
              Disertation , 2012.

   [Chen_HowSensitive]
              Chen, K., Huang, P., and L. Chin-Luang, "How sensitive are
              online gamers to network quality?", Communications of the
              ACM 49, 2006.

   [Claypool_Latency]
              Claypool, M. and K. Claypool, "Latency and player actions
              in online games", Communications of the ACM 49, 2006.

   [Dick_Analysis]
              Dick, M., Wellnitz, O., and L. Wolf, "Analysis of factors
              affecting players' performance and perception in
              multiplayer games", Proceedings of 4th ACM SIGCOMM
              workshop on Network and system support for games, pp. 1 -
              7 , 2005.

   [Dusi_Thin]
              Dusi, M., Napolitano, S., Niccolini, S., and S. Longo, "A
              Closer Look at Thin-Client Connections: Statistical
              Application Identification for QoE Detection", IEEE
              Communications Magazine, pp. 195 - 202 , 2012.

   [Han_GameClassification]
             Han, Y-T. and H-S. Park, "Game Traffic Classification
             Using Statistical Characteristics at the Transport Layer",
             ETRI Journal pp. 22 - 32 32, 2010.

   [Henderson_QoS]
             Henderson, T. and S. Bhatti, "Networked games: a QoS-
             sensitive application for QoS-insensitive users?",
             Proceedings of the ACM SIGCOMM workshop on Revisiting IP
             QoS: What have we learned, why do we care?, pp. 141-147 ,
             2003.

   [Kaiser_objective]
             Kaiser, A., Maggiorini, D., Boussetta , K., and N. Achir,
             "On the Objective Evaluation of Real-Time Networked
             Games", Proc. IEEE Global Telecommunications Conference
             (GLOBECOM 2009) , 2009.

   [Liu_M2M] Liu, R., Wu, W., Zao, H., and D. Yang, "M2M-Oriented QoS
             Categorization in Cellular Network", Master of Science
             Disertation , 2012.

   [Nguyen_TCSurvey]
             Nguyen, T. and G. Armitage, "A Survey of Techniques for
             Internet Traffic Classification using Machine Learning",
             IEEE Communications Surveys and Tutorials pp. 56 - 76. 10,
             2008.

   [Oliveira_online]
             Oliveira, M. and T. Henderson, "What online gamers really
             think of the Internet?", Proceedings of the 2nd workshop
             on Network and system support for games (NetGames '03).
             ACM, New York, NY, USA pp. 185-193, 2003.

   [Ries_QoEMMORPG]
             Ries, M., Svoboda, P., and M. Rupp, "Empirical Study of
             Subjective Quality for Massive Multiplayer Games",
             Proceedings of the 15th International Conference on
             Systems, Signals and Image Processing, pp.181 - 184 ,
             2008.

   [TCM]   Saldana, J., Wing, D., Fernandez Navajas, J., Perumal, M.,
             and F. Pascual Blanco, "Tunneling Compressed Multiplexed
             Traffic Flows (TCM)", Internet-Draft Jul, 2012.

   [TGPP_TR26.944]
             3rd Generation Partnership Project;, "Technical
             Specification Group Services and System Aspects; End-to-
             end multimedia services performance metrics", 3GPP TR
             26.944 version 9.0.0 , 2012.

   [TGPP_TS] 3rd Generation Partnership Project, European
             Telecommunications Standards Institute, "Quality of
             Service (QoS) concept and architecture", 3GPP TS 23.107
             version 11.0.0 Release 11 , 2012.

Authors' Addresses

   Mirko Suznjevic
   University of Zagreb
   Faculty of Electrical Engineering and Computing, Unska 3
   Zagreb 10000
   Croatia

   Phone: +385 1 6129 755
   Email: mirko.suznjevic@fer.hr

   Jose Saldana
   University of Zaragoza
   Dpt. IEC Ada Byron Building
   Zaragoza 50018
   Spain

   Phone: +34 976 762 698
   Email: jsaldana@unizar.es