TCMTF: Tunneling, Compressing and Multiplexing Traffic Flows

Jose Saldana
Julián Fernández-Navajas
José Ruiz-Mas
University of Zaragoza, Spain

Dan Wing
Muthu A.M. Perumal
Michael Ramalho
Cisco Systems

Gonzalo Camarillo
Ericsson Research
INDEX

I. Is there a problem?
II. Is TCMTF a solution to the problem?
III. Is TSVWG the correct place to solve it?
INDEX

I. Is there a problem?
II. Is TCMTF a solution to the problem?
III. Is TSVWG the correct place to solve it?
Is there a problem?

Problem: Inefficiency of real-time flows
- High frequency implies:
  - Small payloads
  - IPv4/UDP/RTP headers: 40 bytes

$\eta = \frac{1460}{1500} = 97\%$

One IPv4/TCP packet 1500 bytes

One IPv4/UDP/RTP VoIP packet with two samples of 10 bytes

$\eta = \frac{20}{60} = 33\%$
Is there a problem?

Problem: Inefficiency of real-time flows

- High frequency implies:
  - Small payloads
  - IPv6/UDP/RTP headers: 60 bytes

One IPv6/TCP packet 1500 bytes
\( \eta = \frac{1440}{1500} = 96\% \)

One IPv6/UDP/RTP packet of VoIP with two samples of 10 bytes
\( \eta = \frac{20}{80} = 25\% \)
Is there a problem?

- Ten years ago: **Question**: Can we **improve efficiency** when a number of flows share the same path?
  - **Answer**: TCRTP (RFC 4170) **2005**: *Best current practice.*
    - **Audio/Video Transport** (avt) (concluded WG) of RAI Area: it was designed for RTP
Is there a problem?

TCRTP for IPv4

One IPv4/UDP/RTP VoIP packet with two samples of 10 bytes
\( \eta = \frac{20}{60} = 33\% \)

Five IPv4/UDP/RTP VoIP packets with two samples of 10 bytes
\( \eta = \frac{20}{60} = 33\% \)

One IPv4 TCMTF Packet multiplexing five two sample packets
\( \eta = \frac{100}{161} = 62\% \)

saving

40 to 6-8 bytes compression
Is there a problem?

TCRTP saves bandwidth, but what has happened since its publication in 2005?
1) Outbreak of wireless access networks*
2) Publication of ROHC (RFC 4995), 2007*:
Designed for robustness when dealing with high RTT, packet loss. Typical in wireless scenarios.
- Able to compress: **RTP/UDP/IP, UDP/IP, TCP/IP**
- Robust: it is able to maintain context synchronization
- Drawback: Implementation complexity

- May 2010: RFC 5856: ROHC over IPSec

*updated by RFC 5795 in 2010
Is there a problem?

3) New real-time services have increased their popularity (e.g. online games)
   - Some of them do not use RTP (bare UDP, or TCP)
   - They generate tiny packets
   - The users are very sensitive to delay
Is there a problem?

So... why not widen TCRTP’s scope in order to:

- Allow other traffics different from RTP
- Allow these new developed header compression techniques
INDEX

I. Is there a problem?
II. Is TCMTF a solution to the problem?
III. Is TSVWG the correct place to solve it?
Is TCMTF a solution to the problem?

TCMTF proposal:
Three layers
1. Tunneling
2. Multiplexing
3. Compressing
Is TCMTF a solution to the problem?

New options:

1) Different **traffics**
   - RTP
   - UDP
   - TCP
Is TCMTF a solution to the problem?

Backwards compatibility:

TCRTP is this “branch”
Is TCMTF a solution to the problem?

New options:
2) Different **header compression** algorithms.
   The most adequate one can be selected according to:
   - Kind of traffic
   - Scenario: loss, delay
   - Processing capacity
   - Etc.
Is TCMTF a solution to the problem?

New options:
3) Different **mux** algorithms
- Currently: PPPMux
- New developed ones

![Diagram of network protocols](image-url)
Is TCMTF a solution to the problem?

New options:
4) Different **tunneling** algorithms
   - Currently: L2TPv3
   - GRE
   - others

![Diagram showing different tunneling algorithms with layers for compression, multiplexing, and tunneling.](image-url)
Is TCMTF a solution to the problem?

Does it work?

First Person Shooter game (UDP)

One IPv4/UDP server-to-client packet of Counter Strike with 9 players
\[ \eta = \frac{160}{188} = 85\% \]

Four IPv4/UDP client-to-server packets of Counter Strike
\[ \eta = \frac{61}{89} = 68\% \]

One IPv4/TCM packet multiplexing four client-to-server Counter Strike packets
\[ \eta = \frac{244}{293} = 83\% \]

Massively Multiplayer Online Role Playing Game (TCP)

Six IPv4/TCP client-to-server packets of World of Warcraft. \( E[P] = 20 \text{bytes} \)
\[ \eta = \frac{20}{60} = 33\% \]

One IPv4/TCM packet multiplexing six client-to-server World of Warcraft packets
\[ \eta = \frac{120}{187} = 64\% \]
Is TCMTF a solution to the problem?

Does it work?: UDP First Person Shooter

First Person Shooters: Can a Smarter Network Save Bandwidth without Annoying the Players?, "IEEE Communications Magazine, vol. 49, no.11, pp. 190-198, November 2011
Is TCMTF a solution to the problem?

Does it work?: TCP MMORPG

I. Is there a problem?
II. Is TCMTF a solution to the problem?
III. Is TSVWG the correct place to solve it?
Is TSVWG the correct place to solve it?

- This is **cross-area work**. It relates to RAI, Transport, and Internet.
  - L2TPv3: **Internet** Area (RFC 3931, March 2005)
  - PPPMux: **Internet** Area (RFC 3153, August 2001)
  - ECRTP: **RAI** Area (RFC 3545, July 2003)
  - ROHC: **Transport Area**, although it can also compress RTP (RFC 5795, March 2010)

- RAI Area: It does not fit, because RTP is only a **particular case** of the solution.

- **Internet** or **Transport** Area?
Is TSVWG the correct place to solve it?

- **RFC 1122:**
  - Transport Layer: “The transport layer provides end-to-end communication services for applications”.
  - Internet Layer: “All Internet transport protocols use the Internet Protocol (IP) to carry data from source host to destination host. IP is a connectionless or datagram internetwork service, providing no end-to-end delivery guarantees”.

- TCMTF is an end-to-end solution, requiring some knowledge of the traffic to multiplex, and a synchronization of the context on both sides.
So, why not TSVWG?

Thank you
Additional slides
Is there a problem?

Ten years ago: Question: Can we **improve efficiency** when a number of flows share the same path?
- Does this **scenario** exist?
- Are the **added delays** reasonable?
Is there a problem?

Does this scenario exist?

- An enterprise with different offices
- A number of calls share a common path: they can also share the common header
Is there a problem?

Other non-RTP scenarios

- Proxies of a **game-provider** or access network
- Internet café
- Satellite link: Reducing pps: Compressing ACKs of different flows
- A group of users of a remote desktop system (webRTC)
Is there a problem?

Are the added delays reasonable?

Delay upper bound: inter-packet time
3) New real-time services have increased their popularity (e.g. online games)

- Some of them do not use RTP (bare UDP, or TCP)
- They generate tiny packets
- The users are very sensitive to delay
- They use wireless access networks
- Supporting infrastructures are critical. They MUST work 24/7.

- Over-provisioning?. Multiplexing tradeoff: in the rush hour, we can save bandwidth at the cost of adding small delays: flexibility
Is there a problem?
Is there a problem?

Total MMOG active subscriptions listed on the site

Is there a problem?

Does this scenario exist?

Quintum’s PacketSaver technology multiplexes small voice/fax-over-IP packets into larger packets to increase network efficiency, thereby reducing the total amount of packet "overhead" required to transmit voice and fax over IP networks.
Is there a problem?

1) Outbreak of **wireless access networks**
Is there a problem?

Is TCMTF a solution to the problem?

- As inter-packet time is not fixed, we would need a policy to select the packet to multiplex.
Is TCMTF a solution to the problem?

Does it work?: RTP VoIP

Is TCMTF a solution to the problem?

Bandwidth saving IPv4

- **IPv4 10 ms 5 players**
- **IPv4 10 ms 20 players**
- **IPv4 reached**
- **IPv4 theoretical**

Games:
- Quake 2
- Unreal Tournament
- Counter Strike 1
- Quake 3
- Enemy Territory
- Counter Strike 2
- Halo 2
- Quake 4
Is TCMTF a solution to the problem?

Bandwidth saving IPv6

- IPv6 10 ms 5 players
- IPv6 10 ms 20 players
- IPv6 reached
- IPv6 theoretical
Is TCMTF a solution to the problem?
Is TCMTF a solution to the problem?

Wolfenstein: Enemy Territory

Counter Strike II

Halo II

Quake IV