



HAL
open science

An End-2-End tunnel header compression solution for nested mobile networks

Priyanka Rawat, Jean-Marie Bonnin, Laurent Toutain, Ana Carolina Minaburo Villar

► **To cite this version:**

Priyanka Rawat, Jean-Marie Bonnin, Laurent Toutain, Ana Carolina Minaburo Villar. An End-2-End tunnel header compression solution for nested mobile networks. ICLAN'2007: International Conference on the Latest Advances in Networks, Dec 2007, Paris, France. hal-02369576

HAL Id: hal-02369576

<https://hal.science/hal-02369576>

Submitted on 19 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An End-2-End Tunnel Header Compression Solution for Nested Mobile Networks

Priyanka Rawat, Jean-Marie Bonnin, Ana Minaburo, and Laurent Toutain

GET/ENST Bretagne

355576 Cesson Sévigné Cedex, France

{priyanka.rawat, jm.bonnin, anacarolina.minaburo, laurent.toutain}@enst-bretagne.fr

Abstract

Various IP tunneling methods are widely used in network security (VPN), IPv4-to-IPv6 transition and most recently to provide IP mobile node with mobility support. However, these IP tunneling methods use multiple levels of encapsulation using several IP and transport headers in each packet. That introduces high protocol header overheads leading to bad performance in wireless links where bandwidth remains a scarce resource.

We propose the use of TuCP, a novel tunneling compression protocol, in order to drastically reduce the tunneling overhead especially in the case when several nested tunnels are used. TuCP provides an end-2-end tunnel header compression solution usable in mobile IP scenario even in the presence of several nested tunnels. Such nested mobility scenarios could occur when a mobile node or a mobile network attaches itself to another mobile network. We present some preliminary results on the performance evaluation of the TuCP compression for different types of flows.

1 Introduction

The concept of IP tunneling evolved around early 90s and since then, it has been widely used to provide solutions for network security problems. The tunneling concept has been exploited further and various IP tunneling methods are being used for transition from IPv4 to IPv6 and NAT traversal using UDP [12] encapsulation for IPv6 packets, for example using L2TP [16] tunnel. Nowadays, IP in IP encapsulation [14] is being used to provide mobility support in mobile networks such as wireless LAN, Personal Area Network (PAN), and Car Area Network (CAN). This mobile communication environment involves multiple levels of mobile networks. A mobile network uses a bi-directional IP/IP tunnel between the Mobile Router (MR) and the Home Agent (HA). However, when NEMO Ba-

sic [6] Support protocol is used, a nested mobile network uses multiple levels of bi-directional tunnels [1] [7]. This nested tunneling leads to an undesirable overhead.

Route Optimization (RO) [5] solution is used for mobile nodes, NEMO, and in network security to overcome the problems of increased length of packet route and header overhead. However, even if RO is usable in MobileIP context, it is a very complicated solution and is still a research topic in the case of network mobility, especially for large scale deployment. Moreover, in the case of mobile networks, some CNs (Correspondent Nodes) may not agree to employ RO for specific flows because of security concerns. So, we have some flows with RO and some without RO. Thus, we will continue to use various tunneling mechanisms to pass through a network using another network protocol or another addressing space.

Several IP tunneling methods are in use, the simplest being IP in IP tunneling, which is used in IP mobility protocol and for security in conjunction with IPSec [15]. Some tunneling methods could also be used in order to build an overlay network for transition purposes (passing through an IPv4 cloud to reach IPv6 Internet). As we connect to an ISP and since we often have to traverse a NAT, these methods tend to use a transport protocol such as UDP or L2TP. The latter, allows to extend a PPP [13] connection through the Internet to the Network Access Server of the ISP. In this case, we have many supplementary headers in each packet and at least IP/UDP/L2TP/PPP/... headers. This leads to bad performance in bandwidth constrained networks.

A solution to this problem is use of header compression mechanisms to reduce the tunnel header overheads. Several header compression algorithms have been studied that propose to reduce the protocol header size. ROHC [2] is the most commonly used header compression protocol. It intends to compress IP and following headers depending on the ROHC profile used when packets travel over a link (level 2). The link could be a PPP connection over a serial link or L2TP session. It could also be a SMDCP (Subnetwork Dependent Convergence Protocol) connection in UMTS. Some

ROHC profiles such as IP-only, IP/UDP, IP/UDP/RTP have been defined and some new profiles such as IP/TCP are still under definition. ROHC is known to be able to reduce the header size and performs well over wireless links where the packet loss rate is high. The IP/UDP/RTP profile of ROHC compresses the overhead of 40 bytes for IPv4 or 60 bytes for IPv6 into 2-3 bytes. Thus, use of ROHC in tunneling makes tunneling mechanisms more efficient due to the reduced overhead.

However, the existing header compression protocols such as ROHC focus on the compression of the inner headers (i.e., IP, UDP, and RTP [4]) contained in the IP packet carried into tunnel and do not deal with the compression of the outer headers used by the tunneling mechanism (e.g., UDP or UDP/L2TP/PPP). It should be noted that the outermost header, namely the IP header, could not be compressed since it is needed by the intermediate routers to route the packet to the tunnel endpoint. Moreover, present header compression mechanisms do not deal with the case of nested tunnels even if supplementary headers used for inner tunnels are useless for the outermost tunnel packet routing purpose. In our previous work [11], we proposed the use of TuCP (Tunneling Compression) in conjunction with ROHC to reduce the tunnel overhead in NEMO networks. In this paper, we show how the solution of ROHC and TuCP compression can be used in nested tunneling scenarios such as those found in nested mobile networks.

The paper is organized as follows. Following the introduction, section 2 present an overview of nested tunneling in nested mobile networks. Section 3 describes the TuCP protocol. Section 4 explains our TuCP approach to provide end-2-end tunnel header compression in nested tunnels. Further, we give our preliminary results on the performance evaluation of TuCP compression for different types of flows in section 5 and conclude with a section on conclusion and future work.

2 Nested Tunneling in Nested Mobile Networks

In the near future, application of mobile communication networks such as wireless LAN, Personal Area Network (PAN), and Car Area Network (CAN) will make it possible to have a permanent Internet access in public transportation systems such as bus, train, ship, and airplane while on the move. NEMO and IPv6 [3], MIPv6 [8], and FMIPv6 [9] protocols have been chosen as the network level of the CALM [17] architecture which is intended to be the architecture of networked car in the near future. NEMO protocol which has been proposed and standardized at IETF, provides mobility management support for mobile networks. NEMO works through the use of a Home Agent (HA) and Mobile Router (MR). A mobile network is connected to

the Internet via MR. The MR has a HA, and it maintains a bi-directional tunnel with its HA. The HA is placed in the home network for the MR and this home network controls the mobile network. The MR-HA bi-directional tunnel preserves session continuity while the MR moves around and all the traffic is via the MR-HA tunnel.

However, a mobile network may be nested. A nested mobile network consists of multiple levels of mobility. A mobile network is called nested when it attaches to another mobile network to multiple level. NEMO basic support protocol supports nested mobile networks. However, in the case of nested mobile networks, for each level of nesting, a bi-directional tunnel is established between each pair of MR/HA. Packets are routed through several HAs and they will be encapsulated and tunneled several times to reach the destination. Thus, the application of NEMO protocol on nested mobile network increases the tunnel within a tunnel overhead. Figure 1 shows the multiple levels of tunnels in a nested NEMO network in a moving train carrying passengers. In the nested case of Figure 1, the Correspondent Node (CN) sends a packet to a Mobile Node (laptop) in the passenger's PAN (Personal Area Network) traveling within the train's VAN (Vehicular Area Network). The passenger's PAN is the mobile network 1 connected to the train's VAN, mobile network 2 via MR2. The train's VAN is connected to the Internet via MR1. The packet from the CN will pass through the tunnel between each MR and its HA(HA_MR). Thus, the tunneling from CN to MN (Mobile Node) is $CN \Rightarrow HA_MR2 \Rightarrow HA_MR1 \Rightarrow MR1 \Rightarrow MR2 \Rightarrow MN$. We do not show the reverse path from MN to CN in the figure.

3 TuCP

IP tunneling is the encapsulation of a packet within another packet, both of which supporting the same or different protocols. Figure 4 shows packet encapsulations in level-1 and level-2 tunneling in a nested NEMO network. IP tunneling consists of inner and outer encapsulation. The tunneled protocol gives the inner encapsulation and the tunneling protocol represents the outer encapsulation. The existing header compression protocols such as ROHC can be used inside of a tunnel to reduce the size of the IP, UDP, and RTP headers of the IP packet carried in the tunnel. However, the tunnel itself has overhead due to its IP header and the tunnel header. It is complex to use ROHC to compress both the inner and outer encapsulation together because in that case we will need to modify the existing header compression mechanism to take into account tunneling. Also, we do not want to compress the outermost IP header of the tunnel because it is used by routers to forward the packet to the tunnel end-point. Moreover, the existing header compression mechanisms do not deal with the case of nested

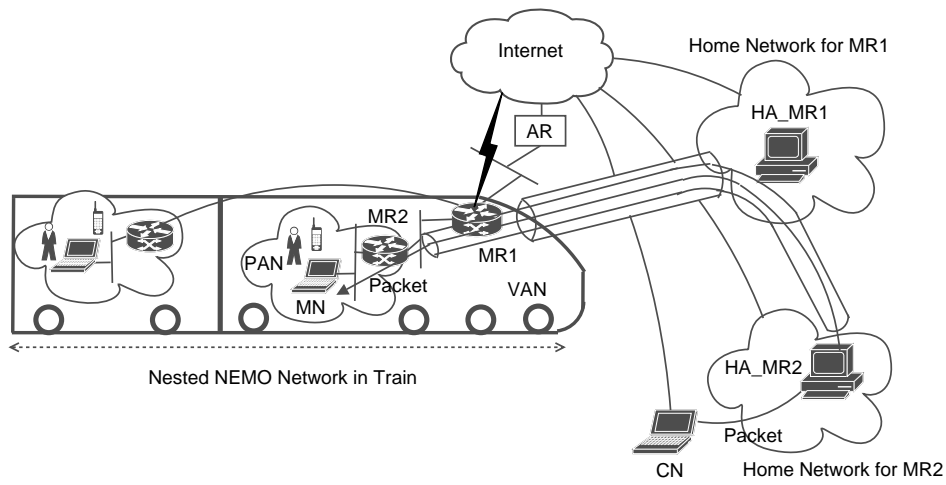


Figure 1. Nested Tunnels in a Nested NEMO Network

tunnels as found in nested mobile networks.

TuCP addresses the above issues. TuCP provides an end-to-end tunnel header compression for nested tunnels in nested mobile networks. TuCP can be used to compress the tunnel header (outer encapsulation) without the need to modify the existing header compression scheme and it does not compress the IP header of the tunnel to be used for routing purposes. As, tunnels are bi-directional, header compression mechanisms will be able to perform at both the ends of the tunnel and use feedbacks.

3.1 TuCP Profiles

This section defines different profiles of the TuCP protocol. Tunneling protocols add one or more additional tunnel headers to the tunneled header (inner IP encapsulation) of the IP packet carried in the tunnel and are used to identify different tunnels. In the outer IP encapsulation, IP protocol is used together with one or more tunneling protocols or without any protocol. These protocols can be UDP, L2TP, and PPP etc. We have defined four TuCP profiles for the header compression of the tunnel headers as shown in Table 1. TuCP profile 3 is used when UDP is used for NAT traversal i.e., UDP header is not compressed. TuCP classifies the tunnel header fields into static and dynamic fields. The TuCP compressor first establishes the required context at the decompressor. The context stores the information about the tunnel header fields. The decompressor processes the received compressed packet to regenerate the uncompressed packet. Figure 2 gives a general classification of header fields for tunneling protocols UDP, L2TP, and PPP.

Protocol	Header Field	Classification	Size (Bits)
UDP (8 Bytes)	Source Port	Static	16
	Destination Port	Static	16
	Datagram Length	Static(Inferred)	16
	Checksum	Dynamic	16
L2TP (6-16 Bytes)	Flags (T, L, S, O, P)	Static	12
	Version	Static	4
	Length	Static	16
	Tunnel ID	Static	16
	Session ID	Static	16
	Ns	Dynamic	16
	Nr	Dynamic	16
	Offset Size	Dynamic	16
Offset Pad	Static	16	
PPP (4 Bytes)	Address	Static	8
	Control	Static	8
	Protocol	Static	16

Figure 2. Header Fields Classification

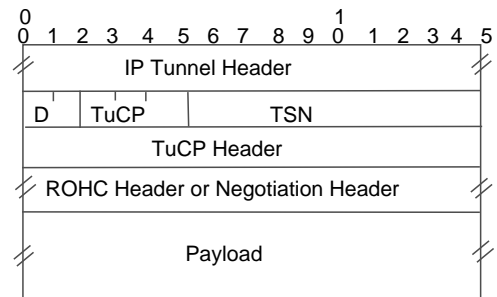


Figure 3. TuCP Packet

Table 1. TuCP Profiles.

Profiles (Ti)	Tunnel Headers
T0 (Profile 0)	No tunneling header
T1 (Profile 1)	UDP
T2 (Profile 2)	UDP/L2TP/PPP
T3 (Profile 3)	L2TP/PPP

3.2 TuCP General Packet Format

The general format of TuCP compressed packet is shown in Figure 3, where “/” represents the variable length of the field. In the TuCP packet, IP header of the tunnel is used by routers to forward the IP packet to the tunnel end-point. Description (D) type bits (2 bits) are used to identify ROHC negotiation packet and ROHC header compressed packets. TuCP (3 bits) bits are used to identify the TuCP profile.

Most header compression mechanisms such as ROHC are designed to work over an ordered delivery transmission between the end-points. Tunneled transport does not guarantee ordered delivery of packets. TuCP handles packet re-ordering. We have introduced a Transfer Sequence Number (TSN) in the TuCP packet to indicate packet re-ordering. TSN gives the decompressor, the transmission order in which the packets have been sent by the compressor. In the presence of out of order packets, before making decompression of an early arriving packet, the decompressor has to wait until the ordered delivery packet arrives or a timer expires. When the timer expires, missing packets and the following are assumed to be lost.

4 Nested Tunnel Header Compression with TuCP

The mobile communication environment involves nested mobile networks. These nested mobile networks require multiple bi-directional tunnels in order to forward packets to the nested mobile network nodes. The MR-HA tunnel approach to provide network mobility support as shown in Figure 1 offers several benefits in comparison to other routing-based approaches. However, nested tunnels lead to an undesirable overhead.

We propose the use of TuCP to compress the multiple tunnel headers in nested tunneling scenarios such as those found in nested mobile networks. TuCP provides an end-to-end tunnel header compression for nested tunnels. Also, if there is a wireless link in between the tunnel end-points, we can use ROHC header compression over wireless link to save bandwidth further. IP tunneling methods use an IP encapsulation of minimum 2 IP headers. Thus, level-1 tunneling uses an IP encapsulation of 2 IP headers. However, in level-n tunneling, the IP encapsulation increases to n+1 IP

Table 2. Src. and Dst. Addresses.

	CN Packet	Tunnel A1A2	Tunnel B1B2
SA	CN add	HA_MR2 add	HA_MR1 add
DA	MNN add	CoA(MR2)	CoA(MR1)

headers. Figure 4 shows packet encapsulations in level-2 nested tunnels when TuCP compression is not used. Figure 5 shows an end-2-end TuCP compression in a level-2 nested tunneling scenario. We consider a scenario of level-2 IP/UDP/L2TP/PPP tunnel with 2 nested tunnels namely A1↔A2 and B1↔B2 as shown in Figure 5. We can have a set of TuCP compressor and decompressor at the tunnel end-points A1, A2, B1, and B2. A1 and B1 are the compression points and, A2 and B2 are the decompression points.

We have a set of ROHC (C/D) compressor and decompressor at each HA and MR to compress the IP packet. In the nested tunnel case, for each level of nesting, we have a set of TuCP (C/D) compressor and decompressor at the tunnel end-points of each bi-directional tunnel. Tunnel headers of the packets are compressed/decompressed at the tunnel end-points. In the level-1 tunnel A1A2, at the point A1, TuCP compresses the tunnel header UDP/L2TP/PPP of the tunnel 1 to give the TuCP compressed packet. The IP header (IP2) of the tunnel A1A2 is used for routing purposes, hence it is not compressed. TuCP compression is done before sending the ROHC compressed tunneled packet through the tunnel. ROHC compression is done at the HA before the routing decision has been taken. Similarly, in the level-2 tunnel B1B2, at the point B1, TuCP compresses the tunnel header to give TuCP2 header. The IP header (IP3) of the tunnel is not compressed as it is used for routing purposes. However, the inner IP header (IP2) is compressed using ROHC. The source (SA) and destination (DA) addresses of the encapsulations are given in Table 2. At the decompression end-points B2 and A2, TuCP decompresses the TuCP packets. TuCP compression can be applied for N number of tunnels to compress the tunnel headers in nested tunnel cases.

In the presence of several nested tunnels, TuCP compresses the tunnel header of the outermost tunnel and ROHC compresses the inner headers of the IP packet carried in the tunnel. When, the TuCP + ROHC compressed packet enters into another tunnel, the residual headers are compressed by ROHC using the corresponding ROHC profile as shown in Figure 4 and 5. TuCP compresses the tunnel headers end-to-end reducing the multiple tunnel overhead and saving bandwidth on intermediate links. The TuCP tunneling header compression solution is scalable. At every tunnel end-point, there is a set of TuCP compressor and decompressor and different contexts are maintained for different flows. At each tunnel end-point, each flow will manage

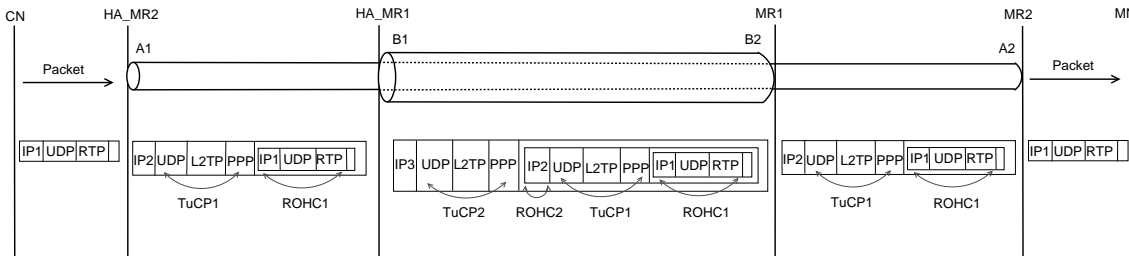


Figure 4. Nested Tunnels without TuCP Compression

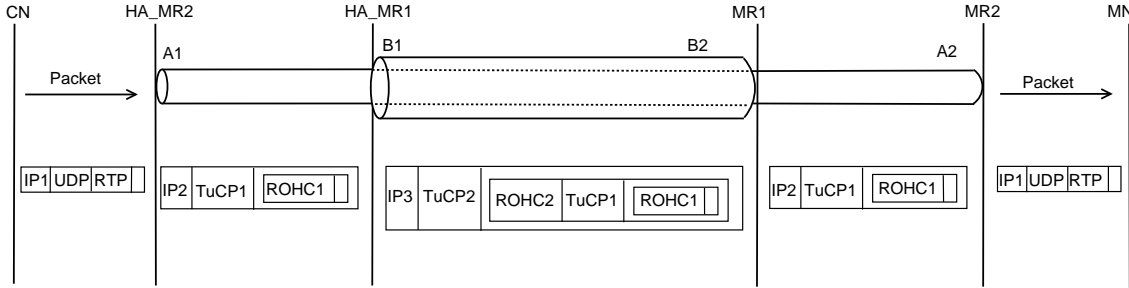


Figure 5. Nested Tunnels with End-2-End TuCP Compression

its own context.

5 Results

We focus on the nested tunneling in NEMO networks. We show some preliminary results on the performance evaluation of the header overhead reduction for different types of flows. We evaluated 3 different profiles of ROHC and found that ROHC compresses the inner headers of the IP packet from 40 bytes for IPv4 or 60 bytes for IPv6 into 2-3 bytes [10]. Figure 6 and 7 show the header size without any compression and with TuCP + ROHC compression, respectively for various combinations of TuCP and ROHC, (T_i , R_j) profiles in level-2 tunneling. T_i represents the TuCP profiles and R_j represents the ROHC profiles. Figure 6 shows the header compression achieved for IPv6/IPv4 case i.e., when the inner IP header (IP header of the IP packet carried in the tunnel) is IPv4 and the outermost tunnel header is IPv6. Figure 7 shows the header compression for IPv6/IPv6 case i.e., when the inner IP header is also IPv6. The ROHC (R_j) profiles tested are $R_1 = \text{IP/UDP/RTP}$, $R_2 = \text{IP/UDP}$, and $R_3 = \text{IPv4 or IPv6}$. TuCP (T_i) profiles are shown in Table 1. TuCP profile 1 (T_1) reduces the overhead of 8 bytes of UDP tunnel to 2 bytes. Profile 2 (T_2) compresses the header size of UDP/L2TP/PPP tunnel from 23 bytes to 2 bytes. Profile 3 (T_3) reduces the tunnel overhead from 15 bytes to 1 byte when UDP is used for NAT traversal. TuCP profile 0 (T_0) adds 1 byte to the header to allow reordering of packets when required. Thus, TuCP reduces the tunnel overhead

to 2 bytes. Figure 8 shows the compression efficiency of ROHC + TuCP compression for different (T_i , R_j) profiles in level-2 tunneling. Header compression is more efficient for IPv6 flows compared to IPv4 flows. It is possible to achieve 66% compression efficiency for IPv6 flows. The use of TuCP in conjunction with ROHC in IP tunnels reduces the header overhead to 2 bytes. This makes tunneling mechanisms virtually costless in terms of bandwidth consumption.

6 Conclusion and Future Work

In this paper, we have presented an overview of nested tunneling problem. We have focused on nested tunneling problem in nested mobile networks. The use of multiple levels of tunnels leads to an undesirable overhead. This leads to bad performance in bandwidth constrained networks. To overcome the problems of increased length of packet route and header overhead, RO solution is used for mobile nodes, NEMO, and in network security. However, RO is a complicated solution and is not feasible for large scale deployment. Thus, we will continue to use various IP tunneling mechanisms in network security, IP transition, mobileIP, and NEMO networks. However, the use of tunneling mechanisms adds high overhead due to nested tunneling.

Several header compression mechanisms have been proposed to reduce the protocol header size in wireless links. However, the existing header compression protocols do not

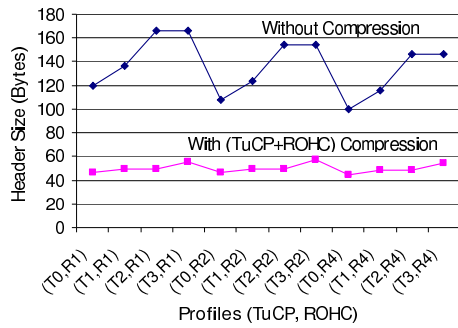


Figure 6. Header Compression for IPv6/IPv4

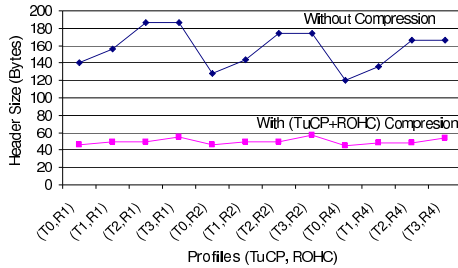


Figure 7. Header Compression for IPv6/IPv6

deal with the case of nested tunnels as found in nested mobile networks. We have proposed the use of TuCP, a novel tunneling compression protocol to compress the tunnel headers when several nested tunnels are used. TuCP is used in conjunction with ROHC protocol. ROHC compresses the inner headers of the IP packet carried in the tunnel whereas TuCP is used to compress the tunnel headers of the tunnel (except the outermost IP header of the tunnel).

TuCP provides an end-2-end tunnel header compression solution for an arbitrary number of nested tunnels. Use of TuCP to reduce the tunnel header overhead in nested tunnels will give better throughput and more efficient network bandwidth usage. Moreover, the TuCP approach is scalable and it handles packet reordering. Most header compression mechanisms such as ROHC are designed to work over an ordered delivery transmission between the end-points. Tunneled transport does not guarantee ordered delivery of packets. TuCP uses TSN in each TuCP packet to indicate packet reordering. We have shown preliminary results to show that TuCP + ROHC compression drastically reduces the tunneling overhead and makes tunneling mechanisms more efficient especially in the presence of several nested tunnels. We are presently implementing various profiles of the TuCP protocol to test it in different levels of tunneling. Our future work involves the study of context transfer between the MR and the HA in nested mobile networks.

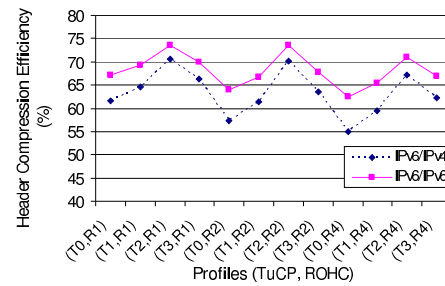


Figure 8. Header Compression Efficiency for ROHC + TuCP Compression

References

- [1] R. Bhagavathula, N. Thanthry, K. Namuduri, and R. Pendse. Issues with Nested Mobility. *VTC*, 2003.
- [2] C. Borman and et al. ROBust Header Compression ROHC: Framework and four profiles: RTP, UDP, ESP, and uncompressed. RFC 3095, IETF, July 2001.
- [3] S. Deering and R. Hinden. Internet Protocol Version 6 IPv6. RFC 2460, IETF, Dec. 1998.
- [4] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 1889, IETF, Jan. 1996.
- [5] M. Calderon, C. J. Bernardos, M. Bagnulo, I. Soto, and A. D. Oliva. Design and experimental evaluation of a route optimization solution for nemo. *IEEE J-SAC*, 2006.
- [6] V. Devarapalli. et al. NEMO Basic Support Protocol. RFC 3963, IETF, Jan. 2005.
- [7] B. Gaabab, J. M. Bonnin, A. Belghith, and M. Watari. IPv6 deployment over IPv4 only cellular networks. *ISIVC, France*, 2004.
- [8] D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. RFC 3775, IETF, June 2004.
- [9] R. Koodli. Fast Handovers for Mobile IPv6. RFC 3775, IETF, July 2005.
- [10] A. Minaburo, L. Nuaymi, K. D. Singh, and L. Toutain. Configuration and Analysis of Robust Header Compression in UMTS. *PIMRC, Beijing*, 2003.
- [11] A. Minaburo, L. Toutain, J.-M. Bonnin, P. Rawat, and E. K. Paik. IP Tunneling Header Compression in NEMO Networks. (*WONEMO*), *Japan*, 2006.
- [12] J. Postel. User Datagram Protocol. RFC 768, IETF, Aug. 1980.
- [13] W. Simpson. The Point-to-Point Protocol for PPP. RFC 1661, IETF, July 1994.
- [14] W. Simpson. IP in IP Tunneling. RFC 1853, IETF, Oct. 1995.
- [15] R. Thayer, N. Doraswamy, and R. Glenn. IP Security Document Roadmap. RFC 2411, IETF, Nov. 1998.
- [16] W. Townsley, A. Valencia, A. Rubens, G. Pall, G. Zorn, and B. Palter. Layer Two Tunneling Protocol "L2TP". RFC 2661, IETF, Aug. 1999.
- [17] B. Williams. *CALM Handbook, ISO TC 204 WG16*. 2006.