Softwire Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 14, 2013

Y. Cui
Tsinghua University
Q. Sun
China Telecom
M. Boucadair
France Telecom
T. Tsou
Huawei Technologies
Y. Lee
Comcast
I. Farrer
Deutsche Telekom AG
July 13, 2012

Lightweight 4over6: An Extension to the DS-Lite Architecture
draft-cui-softwire-b4-translated-ds-lite-07

Abstract

This document specifies an extension to DS-Lite called Lightweight 4over6. This mechanism moves the translation function from the tunnel concentrator (AFTR) to initiators (B4s), and hence reduces the mapping scale on the concentrator to a per-subscriber level. To delegate the NAPT function to the initiators, port-restricted IPv4 addresses are allocated to the initiators.

Table of Contents

1. Introduction

Dual-Stack Lite (DS-Lite, [RFC6333]) provides IPv4 access over an IPv6 network relying on two functional elements: B4 and AFTR. The B4 element establishes an IPv4-in-IPv6 softwire to the AFTR and encapsulates IPv4 packets within IPv6 packets. When the AFTR receives these IPv6 packets, it de-capsulates them and then performs NAPT44 [RFC3022] on the IPv4 packets. This procedure allows the AFTR to dynamically assign port numbers to requesting hosts; hence, increasing the port-sharing ratio and utilization (see [RFC6269]). There is a trade-off, however: the AFTR is required to maintain active NAPT sessions. In the centralized deployment model where one AFTR serves a large number of hosts, the huge number of NAPT sessions may become a performance bottleneck. A large NAPT table demands more processing power for maintaining and searching, as well as consumes more memory space. On the other hand, NAPT44 function is already widely supported and used in today's CPE devices. By leveraging this existing NAPT function and perform NATPT44 on the CPEs, the binding table in the centralized AFTR can be significantly reduced, and the AFTR can offload the NAPT functionality.

This document proposes such an extension to the DS-Lite model. The extension is designed to simplify the AFTR element by moving NAPT functionality to the B4 elements. The B4 element is provisioned with an IPv6 prefix, an IPv4 address and a port-set. An IPv6 address from the assigned prefix is used to create the softwire, while the IPv4 address and port-set is used for NAPT44 in the home gateway (CPE). The CPE performs NAPT on the end user's packets with the IPv4 address and port-set. IPv4 packets are forwarded between the CPE and the AFTR using IPv4-in-IPv6 encapsulation. The AFTR maintains a mapping entry with the CPE's IPv6 address, IPv4 address and port-set per subscriber. For inbound IPv4 packets received by the AFTR, the IPv4 destination address and port are used to find the IPv6 encapsulation destination in the binding table. The AFTR does not maintain any NAPT session entries.

Compared to stateless solutions with port-set allocation such as MAP [I-D.mdt-softwire-mapping-address-and-port], this mechanism is suitable for operators who prefer to keep IPv6 and IPv4 addressing architectures separated. They can administer native IPv6 network addressing without the influence of IPv4-over-IPv6 requirements. For example, an operator may want to provide IPv4 as an on-demand service in its IPv6 network, based on subscriber requests. The dynamic allocation of IPv4 addresses and port-sets makes more efficient usage of IPv4 resources than stateless solutions in this case.

Another example is: An operator may only have many small and non-contiguous IPv4 blocks available to provide IPv4 over IPv6, rather

than a few large contiguous IPv4 blocks. This mechanism preserves the dynamic feature of IPv4/IPv6 address binding as in DS-Lite, so it does not require the administration and management of many MAP domains in the network and corresponding mapping rules in the CPEs.

The model that is presented here offers a solution for a hub-and-spoke architecture only. It does not offer meshed IPv4 connectivity between subscribers. The simplicity and flexibility of IPv4/v6 address planning and provisioning described here are a tradeoff for this reduced functionality: the subscriber does not need the information of other subscribers.

This document is an extended case, which covers address sharing for [I-D.ietf-softwire-public-4over6]. It is also a variant of A+P called Binding Table Mode (see Section 4.4 of [RFC6346]).

This document focuses on architectural considerations and particularly on the expected behavior of involved functional elements and their interfaces. Deployment-specific issues are discussed in a companion document. As such, discussions about redundancy and provisioning policy are out of scope.

2. Conventions

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

3. Terminology

The document defines the following terms:

- Lightweight 4over6: Lightweight 4over6 is an IPv4-over-IPv6 hub and spoke mechanism, which supports address sharing [RFC6269] and performs the IPv4 translation (NAPT44) on the initiator (spoke) side.

- Lightweight 4over6 initiator (or "initiator"): the tunnel initiator in the Lightweight 4over6 mechanism. The Lightweight 4over6 initiator may be a host directly connected to an IPv6 network, or a dual-stack CPE connecting an IPv4 local network to an IPv6 network. It is collocated with a NAPT44 function in addition to IPv4-in-IPv6 encapsulation and de-capsulation functions.

- Lightweight 4over6 concentrator (or "concentrator"): the tunnel concentrator in the Lightweight 4over6 mechanism. The Lightweight 4over6 concentrator tunnels IPv4 packets to the IPv4 Internet over an IPv6 network. It provides IPv4-in-IPv6 encapsulation and de-capsulation functions but does not perform a NAPT function.

- Port-restricted IPv4 address: A public IPv4 address with a restricted port-set. In Lightweight 4over6, multiple initiators may share the same IPv4 address, however, their port-sets must be non-overlapping. Source ports of IPv4 packets sent by the initiator must belong to the assigned port-set.

4. Lightweight 4over6 Overview

Lightweight 4over6 initiators and a Lightweight 4over6 concentrator are connected through an IPv6-enabled network (Figure 1). Both use an IPv4-in-IPv6 encapsulation scheme to deliver IPv4 connectivity services. An initiator uses a port-restricted IPv4 address for IPv4 services delivered over the IPv6-enabled network (See Section 5 for further detail). The concentrator keeps the binding between the initiator's IPv6 address and the allocated IPv4 address + port-set.

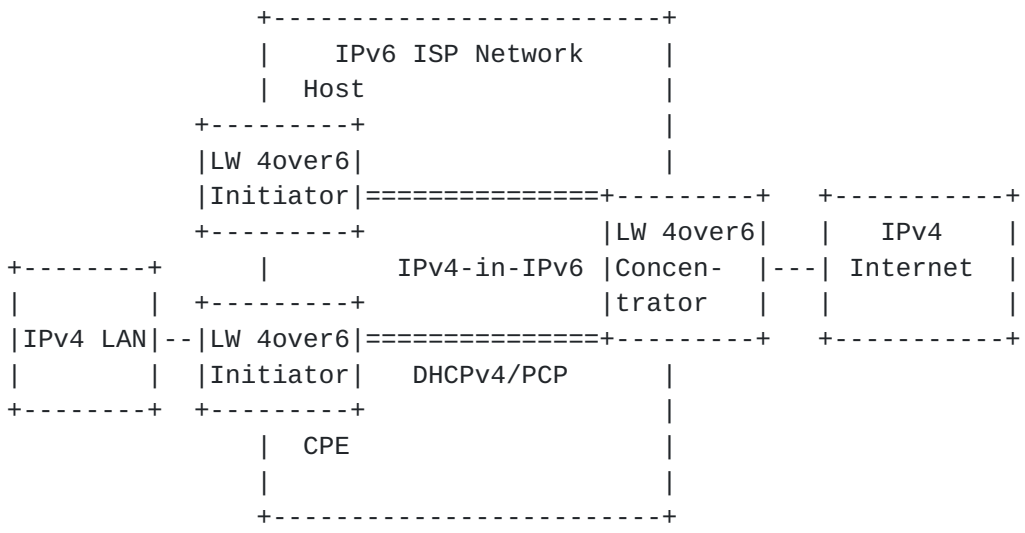

Figure 1 Lightweight 4over6 Overview

5. Port-Restricted IPv4 Address Allocation

In Lightweight 4over6, an initiator is provisioned with a public address and port-set. Different mechanisms can be used for port-restricted IPv4 address provisioning, e.g.- DHCPv4, DHCPv6, PCP, PPP IPCP. The mechanism described in this document uses DHCPv4 as it is

widely deployed in services providers networks and supports all IPv4 and IPv6 addressing models.

DHCPv4 messages between the initiator and the DHCPv4 server MUST be sent over IPv6 [I-D.ietf-dhc-dhcpv4-over-ipv6], and [I-D.bajko-pripaddrassign] MUST be supported for port-set allocation.

Other optional alternatives to retrieve the public address and port-set also exist. The specific protocol extensions are out of scope in this document, however some alternatives are mentioned in the Appendix Section.

6. Lightweight 4over6 Initiator Behavior

6.1. Initiator Provisioning

To configure the IPv4-in-IPv6 tunnel, the Lightweight 4over6 initiator MUST have the concentrator's IPv6 address. This IPv6 address can be learned through a variety of mechanisms, ranging from an out-of-band mechanism, manual configuration, DHCPv6, etc. In order to guarantee interoperability, a Lightweight 4over6 initiator SHOULD implement the DHCPv6 option defined in [RFC6334]. The initiator MUST use its WAN interface for sourcing the DHCPv6 request as defined in [RFC6333].

Multi-homed CPE devices connected to two or more service providers are not covered as part of this document.

A Lightweight 4over6 initiator MUST support dynamic port-restricted IPv4 address provisioning, by means of implementing the DHCPv4 mechanism (including [I-D.ietf-dhc-dhcpv4-over-ipv6] and [I-D.bajko-pripaddrassign]). The IPv6 address of the DHCPv4 server/relay can be configured using a variety of methods, too, ranging from an out-of-band mechanism, manual configuration, a variety of DHCPv6 options, or taking the concentrator address configuration when collocating with concentrator. In order to guarantee interoperability, an initiator SHOULD implement the DHCPv6 option defined in [I-D.mrugalski-softwire-dhcpv4-over-v6-option]. If the DHCPv4 over IPv6 client has multiple IPv6 addresses assigned to it's WAN interface, the mechanisms defined in RFC3484 MUST be applied for selecting the correct address as the source of the DHCPv4 over IPv6 request. A DHCPv4 over IPv6 client embedded within the initiator MUST use the same IPv6 address as the data plane encapsulation source address for all DHCPv4 over IPv6 requests. In the event the encapsulation source address is changed for any reason (such as the DHCP lease expiring), the DHCPv4 over IPv6 process MUST be re-initiated.

6.2. Initiator Data Plane Behavior

The data plane functions of the initiator include address translation (NAPT44), encapsulation and de-capsulation. The initiator runs standard NAPT44 [RFC3022] using the allocated port-restricted address as its external IP and port numbers.

Internally connected hosts source IPv4 packets with an [RFC1918] address. When the initiator receives such an IPv4 packet, it performs a NAPT44 function on the source address and port by using the public IPv4 address and a port number from the allocated port-set. Then, it encapsulates the packet with an IPv6 header. The destination IPv6 address is the concentrator's IPv6 address and the source IPv6 address is the initiator's IPv6 address. Finally, the initiator forwards the encapsulated packet to the configured concentrator.

When the initiator receives an IPv4-in-IPv6 packet from the concentrator, it de-capsulates the IPv4 packet from the IPv6 packet. Then, it performs the NAPT44 function and translates the destination address and port, based on the available information in its local NAPT44 table.

Tunneling MUST be done in accordance with [RFC2473] and [RFC4213].

The initiator is responsible for performing ALG functions (e.g., SIP, FTP), and other NAPT traversal mechanisms (e.g., UPnP, NAPT-PMP, manual mapping configuration, PCP) for the internal hosts. This is the same requirement for typical NAPT44 gateways available today.

It's possible that an initiator is co-located in a host. In this case, the functions of NAPT44 and encapsulation/de-capsulation are implemented inside the host.

7. Lightweight 4over6 Concentrator Behavior

7.1. Binding Table Maintenance

The Lightweight 4over6 concentrator MUST maintain an address binding table. Each entry in the table contains a public IPv4 address, a port-set and an IPv6 address for a single initiator. The entry has two functions: IPv6 encapsulation of inbound IPv4 packets destined to the initiator and validation of outbound IPv4-in-IPv6 packets received from the initiator for de-capsulation.

The concentrator MUST synchronize the binding information with the port-restricted address provisioning process. With DHCPv4 as the

provisioning method, the initiators send DHCP messages to the DHCP server or relay agent over IPv6. If the concentrator implements a local DHCPv4 server or relay agent, the initiators MAY send the messages to the concentrator; then the concentrator is able to learn the bindings between IPv6 address and IPv4 address with port set directly. If the concentrator does not participate in the port-restricted address provisioning process, the binding MUST be synchronized through other methods (e.g. out-of-band static update). The exact mechanism for this is deployment-specific and out of scope. For all provisioning processes, the lifetime of binding table entries MUST be synchronized with the lifetime of address allocations.

7.2. Concentrator Data Plane Behavior

The data plane functions of the concentrator are encapsulation and de-capsulation. When the concentrator receives an IPv4-in-IPv6 packet from an initiator, it de-capsulates the IPv6 header and verifies the source addresses and port in the binding table. If the source addresses and port match an entry in the binding table (that is to say, the source IPv6 address in the IPv6 header is identical to the IPv6 address of the entry, the source IPv4 address in the IPv4 header is identical to the IPv4 address of the entry, and the source port falls into the port-set of the entry), the concentrator forwards the packet to the IPv4 destination. If no match is found (e.g., not authorized IPv4 address, port out of range, etc.), the concentrator MUST discard the packet. An ICMP error message MAY be sent back to the requesting initiator. The ICMP policy SHOULD be configurable.

When the concentrator receives an inbound IPv4 packet, it uses the IPv4 destination address and port to lookup the destination initiator's IPv6 address in the binding table. If a match is found, the concentrator encapsulates the IPv4 packet. The source is the concentrator's IPv6 address and the destination is the initiator's IPv6 address from the matched entry. Then, the concentrator forwards the packet to the initiator natively over the IPv6 network. If no match is found, the concentrator MUST discard the packet. An ICMP error message MAY be sent back. The ICMP policy SHOULD be configurable.

Tunneling MUST be done in accordance with [RFC2473] and [RFC4213].

The concentrator MUST support hairpinning of traffic between two initiators, by performing de-capsulation and re-encapsulation of packets.

8. Fragmentation and Reassembly

The same considerations as described in Section 5.3 and Section 6.3 of [RFC6333] are to be taken into account.

9. DNS

The procedure described in Section 5.5 and Section 6.4 of [RFC6333] is to be followed.

10. ICMP Processing

ICMP does not work in an address sharing environment without special handling [RFC6269]. When implementing Lightweight 4over6, the following behaviour SHOULD be implemented to provide basic remote IPv4 service diagnostics for a port restricted CPE: For inbound ICMP messages, the concentrator MAY behave in two modes:

Either:

1. Check the ICMP Type field.

2. If the ICMP type is set to 8 (echo request), then the concentrator MUST discard the packet.

3. If the ICMP field is set to 0 (echo reply), then the concentrator must take the value of the ICMP identifier field as the source port.

4. If the ICMP type field is set to any other value, then the concentrator MUST use the method described in REQ-3 of [RFC5508] to locate the source port within the transport layer header in ICMP packet's data field. The destination IPv4 address and source port extracted from the ICMP packet are then used to make a lookup in the binding table. If a match is found, it MUST forward the ICMP reply packet to the IPv6 address stored in the entry.

Or:

o Discard all inbound ICMP requests.

The ICMP policy SHOULD be configurable.

The initiator should implement the requirements defined in [RFC5508] for ICMP forwarding. For ICMP echo request packets originating from

the private IPv4 network, the initiator SHOULD implement the method described in [RFC6346] and use an available port from it's port-set as the ICMP Identifier.

For both the concentrator and the initiator, ICMPv6 MUST be handled as described in [RFC2473].

11. Security Consideration

As the port space for a subscriber shrinks significantly due to the address sharing, the randomness for the port numbers of the subscriber is decreased significantly. In other words, it is much easier for an attacker to guess the port number used, which could result in attacks ranging from throughput reduction to broken connections or data corruption. The port-set for a subscriber can be a set of contiguous ports or non-contiguous ports. Contiguous port-sets do not reduce this threat. However, with non-contiguous port-set (which may be generated in a pseudo-random way [RFC6431]), the randomness of the port number is improved, provided that the attacker is outside the Lightweight 4over6 domain and hence does not know the port-set generation algorithm.

More considerations about IP address sharing are discussed in Section 13 of [RFC6269], which is applicable to this solution.

12. IANA Considerations

This document does not include any IANA request.

13. Author List

The following are extended authors who contributed to the effort:

Jianping Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-62785983
Email: jianping@cernet.edu.cn

Peng Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-62785822
Email: pengwu.thu@gmail.com

Chongfeng Xie
China Telecom
Room 708, No.118, Xizhimennei Street
Beijing 100035
P.R.China

Phone: +86-10-58552116
Email: xiechf@ctbri.com.cn

Xiaohong Deng
France Telecom

Email: xiaohong.deng@orange.com

Cathy Zhou
Huawei Technologies
Section B, Huawei Industrial Base, Bantian Longgang
Shenzhen 518129
P.R.China

Email: cathyzhou@huawei.com

Alain Durand
Juniper Networks
1194 North Mathilda Avenue
Sunnyvale, CA 94089-1206
USA

Email: adurand@juniper.net

Reinaldo Penno
Cisco

Email: repenno@cisco.com

Alex Clauberg
Deutsche Telekom AG
GTN-FM4
Landgrabenweg 151
Bonn, CA 53227
Germany

Email: axel.clauberg@telekom.de

Lionel Hoffmann
Bouygues Telecom
TECHNOPOLE
13/15 Avenue du Marechal Juin
Meudon 92360
France

Email: lhoffman@bouyguestelecom.fr

Maoke Chen
FreeBit Co., Ltd.
13F E-space Tower, Maruyama-cho 3-6
Shibuya-ku, Tokyo 150-0044
Japan

Email: fibrib@gmail.com

14. Acknowledgement

The authors would like to thank Ole Troan, Ralph Droms for their comments and feedback.

This document is a merge of three documents: [I-D.cui-softwire-b4-translated-ds-lite], [I-D.zhou-softwire-b4-nat] and [I-D.penno-softwire-sdnat].

15. Appendix: Alternatives for Port-Restricted Address Allocation

Besides DHCPv4, other alternatives for address and port-set provisioning, e.g.- PCP, DHCPv6, IPCP, MAY also be implemented.

- PCP[I-D.ietf-pcp-base]: an initiator MAY use [I-D.tsou-pcp-natcoord] to retrieve a restricted IPv4 address and a set of ports.

- DHCPv6: the DHCPv6 protocol MAY be extended to support port-set allocation [I-D.boucadair-dhcpv6-shared-address-option], along with IPv6-mapped IPv4 address allocation.

- IPCP: IPCP MAY be extended to carry the port-set (e.g., [RFC6431]).

In a Lightweight 4over6 domain, the same provisioning mechanism MUST be enabled in the initiator, the concentrator and the provisioning server.

16. References

16.1. Normative References

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

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

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, December 1998.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC5508] Srisuresh, P., Ford, B., Sivakumar, S., and S. Guha, "NAT Behavioral Requirements for ICMP", BCP 148, RFC 5508, April 2009.

[RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269,

June 2011.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

[RFC6334] Hankins, D. and T. Mrugalski, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite", RFC 6334, August 2011.

[RFC6346] Bush, R., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, August 2011.

[RFC6431] Boucadair, M., Levis, P., Bajko, G., Savolainen, T., and T. Tsou, "Huawei Port Range Configuration Options for PPP IP Control Protocol (IPCP)", RFC 6431, November 2011.

16.2. Informative References

[I-D.bajko-pripaddrassign]
Bajko, G., Savolainen, T., Boucadair, M., and P. Levis, "Port Restricted IP Address Assignment", draft-bajko-pripaddrassign-04 (work in progress), April 2012.

[I-D.boucadair-dhcpv6-shared-address-option]
Boucadair, M., Levis, P., Grimault, J., Savolainen, T., and G. Bajko, "Dynamic Host Configuration Protocol (DHCPv6) Options for Shared IP Addresses Solutions", draft-boucadair-dhcpv6-shared-address-option-01 (work in progress), December 2009.

[I-D.cui-softwire-b4-translated-ds-lite]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-cui-softwire-b4-translated-ds-lite-06 (work in progress), May 2012.

[I-D.ietf-dhc-dhcpv4-over-ipv6]
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-03 (work in progress), May 2012.

[I-D.ietf-pcp-base]
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-26 (work in progress), June 2012.

[I-D.ietf-softwire-public-4over6]
Cui, Y., Wu, J., Wu, P., Metz, C., Vautrin, O., and Y. Lee, "Public IPv4 over IPv6 Access Network", draft-ietf-softwire-public-4over6-01 (work in progress), March 2012.

[I-D.mdt-softwire-mapping-address-and-port]
Bao, C., Troan, O., Matsushima, S., Murakami, T., and X. Li, "Mapping of Address and Port (MAP)", draft-mdt-softwire-mapping-address-and-port-03 (work in progress), January 2012.

[I-D.mrugalski-softwire-dhcpv4-over-v6-option]
Mrugalski, T. and P. Wu, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for DHCPv4 over IPv6 Transport", draft-mrugalski-softwire-dhcpv4-over-v6-option-00 (work in progress), April 2012.

[I-D.penno-softwire-sdnat]
Penno, R., Durand, A., Hoffmann, L., and A. Clauberg, "Stateless DS-Lite", draft-penno-softwire-sdnat-02 (work in progress), March 2012.

[I-D.tsou-pcp-natcoord]
Sun, Q., Boucadair, M., Deng, X., Zhou, C., and T. Tsou, "Using PCP To Coordinate Between the CGN and Home Gateway Via Port Allocation", draft-tsou-pcp-natcoord-05 (work in progress), March 2012.

[I-D.zhou-softwire-b4-nat]
Zhou, C., Boucadair, M., and X. Deng, "NAT offload extension to Dual-Stack lite", draft-zhou-softwire-b4-nat-04 (work in progress), October 2011.

Authors' Addresses

Yong Cui
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-62603059
Email: yong@csnet1.cs.tsinghua.edu.cn

Qiong Sun
China Telecom
Room 708, No.118, Xizhimennei Street
Beijing 100035
P.R.China

Phone: +86-10-58552936
Email: sunqiong@ctbri.com.cn

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Tina Tsou
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

Phone: +1-408-330-4424
Email: tena@huawei.com

Yiu L. Lee
Comcast
One Comcast Center
Philadelphia, PA 19103
USA

Email: yiu_lee@cable.comcast.com

Ian Farrer
Deutsche Telekom AG
GTN-FM4,Landgrabenweg 151
Philadelphia, Bonn 53227
Germany

Email: ian.farrer@telekom.de