Softwire Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 5, 2012

Y. Cui
Tsinghua University
Q. Sun
China Telecom
M. Boucadair
France Telecom
T. Tsou
Huawei Technologies
Y. Lee
Comcast

February 2, 2012

Lightweight 4over6: An Extension to DS-Lite Architecture
draft-cui-softwire-b4-translated-ds-lite-05

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 per-subscriber level. To delegate the NAT function to the initiators, port-restricted IPv4 addresses are allocated to the initiators.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Dual-Stack Lite technique (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 an AFTR and encapsulates IPv4 packets in IPv6 packets. When the AFTR receives theses IPv6 packets, it decapsulates them and then performs NAT44 [RFC3022]. This procedure allows AFTR to dynamically assign port numbers to requesting users; hence, increases port-sharing ratio and utilization (see [RFC6269]). There is a trade-off, though: the AFTR is required to maintain active NAT sessions. In the centralized deployment model where one AFTR serves thousands of users, the large numbers of NAT sessions may become a performance bottleneck. First, maintaining active NAT sessions requires AFTR constantly creating and purging NAT sessions. Second, a large NAT table demands more processing power for searching and consumes more memory space.

To address these issues, this document proposes an extension to DS-Lite technique. The extension is designed to simplify the AFTR element by distributing NAT function among B4 elements. The B4 element is provisioned with an IPv6 address, an IPv4 address and a port-set. The IPv6 address is used to create the Softwire, while the IPv4 address and port-set is used for NAT44 in the home gateway (CPE). The CPE performs NAPT on end user's packets with the IPv4 address and port-set. IPv4 packets are forwarded between the CPE and the AFTR using 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 on AFTR, it uses the IPv4 destination address and port to match the IPv6 encapsulation destination in the mapping table. The AFTR does not maintain any NAT session entry. Therefore, this extension removes the NAT module from the AFTR and significantly reduce the AFTR's mapping table size, and it also relaxes the requirement to create a log entry per active NAT session. In fact, the mechanism is an extended case which covers addressing sharing for [I-D.ietf-softwire-public-4over6].

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 separated. For example, an operator may want to provide IPv4 with an on-demand way in its IPv6 network when subscriber requested, the dynamic allocation of IPv4 address and port-set makes more efficient usage of IPv4 resource. Another example is an operator may only have many small and discontinuous IPv4 blocks available to provide IPv4 over IPv6, rather than few large IPv4 blocks. This mechanism preserves the dynamic feature of IPv4/IPv6 address binding as in DS-Lite, so it won't require to administrate and manage many MAP domains in the network and mapping rules in the CPEs.

This document is a variant of A+P called Binding Table Mode (see Section 4.4 of [RFC6346]).

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 (NAT44) on the initiator (spoke) side.

- Lightweight 4over6 initiator (or "initiator" for short): the tunnel initiator in 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 NAT44 function in addition to IPv4-in-IPv6 encapsulation and de-capsulation functions.

- Lightweight 4over6 concentrator (or "concentrator" for short): the tunnel concentrator in 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 decapsulation functions but not NAT function.

- Port-restricted IPv4 address: A public IPv4 address with restricted port-set. In lightweight 4over6, multiple initiators may share the same IPv4 address, while the 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. Both use IPv4-in-IPv6 encapsulation scheme to deliver IPv4 connectivity services. An initiator uses port-restricted IPv4 address for IPv4 services provisioned by the network. This address may be provisioned via PCP, DHCPv4, DHCPv6, PPP IPCP, etc.(See Section 5 for further detail). The concentrator keeps the mapping between the initiator's IPv6 address and the allocated IPv4 address + port-set.

```
                +--------------------------+
                |    IPv6 ISP Network      |
                |  Host                    |
            +----------+                   |
            |LW 4over6|                    |
            |Initiator|===============+----------+   +-----------+
            +----------+               |LW 4over6|   |   IPv4    |
+---------+      |        IPv4-in-IPv6 |Concen-  |---| Internet  |
|         |  +----------+              |trator   |   |           |
|IPv4 LAN|--|LW 4over6|===============+----------+   +-----------+
|         |  |Initiator|   DHCPv4/PCP      |
+---------+  +----------+                   |
                |  CPE                     |
                |                          |
                +--------------------------+
```

Figure 1 Lightweight 4over6 overview

5. Port-Restricted IPv4 Address Allocation

In lightweight 4over6, an initiator needs to be provisioned with the public address and port-set. Different mechanisms can be used here for port-restricted IPv4 address provisioning, e.g., DHCPv4, DHCPv6, PCP, PPP IPCP, and even manual configuration.

Below are listed examples of implementing the provisioning through the above mechanisms. They are not mandatory requirements, and the specific protocol extensions is out of scope in this document.

- o DHCPv4: the DHCPv4 protocol should be extended to support port-set allocation [I-D.bajko-pripaddrassign]. Besides, the DHCP message should send to the concentrator over IPv6. The concentrator can be the DHCP server or DHCP relay agent[I-D.ietf-dhc-dhcpv4-over-ipv6].

- o PCP[I-D.ietf-pcp-base]: an initiator can launch multiple PCP requests simultaneously to acquire a number ports within the same IPv4 address, or use [I-D.tsou-pcp-natcoord] for one-time port-set allocation.

- o DHCPv6: the DHCPv6 protocol should be extended to support port-set allocation [I-D.boucadair-dhcpv6-shared-address-option].

- o IPCP: IPCP should be extended to carry the port-set. [RFC6431] gives an example.

When using dynamic provisioning mechanism such as DHCP or PCP, the initiator gets the IPv4 address and port-set allocated dynamically from the concentrator. While provisioning the initiator, the concentrator records the dynamic mapping rule between the IPv4 address, port-set and the IPv6 address simultaneously. The port-restricted address allocation in lightweight 4over6 does not couple with IPv6 addressing. Hence, there is no requirement for IPv4-IPv6 mapping relationship such as IPv4 address encoding, IPv6 prefix length, multiplexing ratio, etc.

6. Lightweight 4over6 Initiator Behavior

A lightweight 4over6 initiator must discover 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. The mechanism is out of scope in this document.

A lightweight 4over6 initiator should support dynamic port-restricted IPv4 address provisioning, by means of implementing the appropriate mechanism (e.g., DHCP, PCP, etc.). The mechanism must be align between the initiator and the concentrator (i.e. the PCP Server, DHCPv4 server, etc.), which may be co-located with the concentrator.

The data plane functions of the initiator include address translation (NAT44) and encapsulation/de-capsulation. The initiator runs a standard NAT44 [RFC3022] using the allocated port-restricted address as external IP and port numbers. Internal connected hosts source IPv4 packets with a [RFC1918] address. When the initiator receives the IPv4 packet, it performs NAT44 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. 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 IPv6 packet to retrieve the embedded IPv4 packet. Then it performs NAT44 function and translates the destination address and port based on the available information in its local NAPT44 table.

If the initiator is acting as a CPE, it is responsible for performing ALG functions (e.g., SIP, FTP), and other NAT traversal mechanisms (e.g., UPnP, NAT-PMP, manual mapping configuration, PCP) for the connected hosts. This is the same requirement for typical NAT44 gateways available today.

If the initiator is collocated in the host, the host will be provisioned with the public IPv4 address and port-set directly. Some applications relies on the socket API to allocate IPv4 source port, the API will randomly allocate an available port to the application. To ensure the port is from the provisioned port-set, the host should either implement a local NAT to map the randomly generated port by the API to the restricted port-set or modify the API to return a port from the restricted port-set. Both options enable the host to source IPv4 packet using the restricted port-set without modifying the IPv4 applications.

7. Lightweight 4over6 Concentrator Behavior

The lightweight 4over6 concentrator must create a table in which each entry contains a public IPv4 address, a port-set and an initiator's IPv6 address. The concentrator MUST synchronize the port-restricted address provisioning process such as DHCP and PCP used by the Initiator. This synchronization is deployment-specific (e.g., embed PCP Server, DHCP relay or Server, RADIUS, etc.) When the IPv4 address and port-set is successfully provisioned to the Initiator, the concentrator simultaneously creates a map entry in its table. This entry contains the public IPv4 address, the port-set and the initiator's IPv6 address. The lifetime is determined by the provisioning mechanism. The IPv6 address in the map entry is used as the index for encapsulating inbound packets. This map entry will be deleted when the lifetime expires. The lifetime of the map entry should be refreshed when the initiator renews/extends the allocation.

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 port and address in the table. If the source port and address matches the Initiator's IPv6 address in the table, the concentrator forwards the packet to the IPv4 destination. If not, the concentrator must discard the packet.

When the concentrator receives an IPv4 packet from the Internet, it uses the destination address and port to lookup the destination initiator's IPv6 address in the 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. Then, the concentrator forwards the packet to the initiator natively over the IPv6 network. When no match is found, the concentrator must discard the packet.

8. Fragmentation and Reassembly

Same considerations as Section 5.3 and Section 6.3 of [RFC6333] are to be taken into account.

9. DNS

Section 5.5 and Section 6.4 of [RFC6333] are to be followed.

10. ICMP processing

ICMP does not work through NAT44 [RFC6269]). When implementing lightweight 4over6, ICMP Identifier MUST be treated as port number for UDP/TCP. Therefore, when the initiator generates an ICMP packet, it MUST use an available port from its port-set as the ICMP identifier. When the concentrator receives an ICMP reply packet from the IPv4 network, it must use the identifier as the port number and search its table. If a match is found, it must forward the ICMP reply packet to the initiator stored in the entry. The lookup process is identical to normal TCP/UDP lookup. For inbound ICMP request packet, the concentrator may be configured in two modes:

- o Forward the request to the appropriate initiator using the Identifier field when a mapping entry is found; if not the ICMP request is silently dropped.

- o Discard all inbound ICMP requests.

The ICMP policy is determined by service providers.

11. Mechanism Analysis

Compared with original DS-Lite, lightweight 4over6 move the translation function from the concentrator and distribute it to initiators. This reduces states on the concentrator from per-session level down to per-subscriber level. This potentially reduces the concentrator complexity to manage a relatively large NAT table. In theory, the concentrator should scale better and serve more subscribers on the same hardware platform.

The initiator is provisioned with a public IPv4 address and port-set, and translation is performed only once, so it is a single NAT architecture.

When the initiator acts as a CPE, it is required to implement ALG and NAT referral problem. These problems are solved today by combination of UPnP, NAT-PmP, etc. Many existing CPEs have already implemented these functionalities. So lightweight 4over6 initiator can leverage these existing mechanisms to address the same problems.

When the AFTR performs per-session log, the volume could increase rapidly because each new session may create a new log entry. Some optimization has been discussed to reduce log volume [I-D.donley-behave-deterministic-cgn]. When the concentrator performs per-subscriber tunnel log, each subscriber creates only one log. This is identical to logging subscriber by IPv4 address. This mechanism is widely used today. Service providers can re-use the same mechanism with minor modification.

Lightweight 4over6 does not couple port-restricted address and the IPv6 addressing. No specific IPv6 address format is required. IPv4 and IPv6 addressing and routing remain separated. The service provider can provide IPv4 in a flexible, on-demand way, as well as manage the native IPv6 network without the influence of IPv4-over-IPv6 requirements. This would ease to achieve future adjustment of IPv4 address pool.

The trade-offs of lightweight 4over6 are possibility of lower IPv4 address utilization ratio and extra signaling behavior for provisioning. When compared to stateless solutions, lightweight 4over6 still keeps subscriber-level states rather than becoming purely stateless.

12. 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 more easier for an attacker to guess the port number used, which results in attacks ranging from throughput reduction to broken connections or data corruption. Here the port-set for a subscriber can be a bulk of contiguous ports or non-contiguous ports. Contiguous port-set can't help the situation, while 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 doesn't know the port-set generation algorithm.

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

13. IANA Considerations

This document does not include any IANA request.

14. 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: weapon@csnet1.cs.tsinghua.edu.cn

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

15. Acknowledgement

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

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

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.

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

[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.

[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-03 (work in progress), September 2010.

[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., Wu, J., Wu, P., Sun, Q., Xie, C., Zhou, C., Lee, Y., and T. ZOU), "Lightweight 4over6 in access network",

draft-cui-softwire-b4-translated-ds-lite-04 (work in progress), October 2011.

[I-D.donley-behave-deterministic-cgn]
Donley, C., Grundemann, C., Sarawat, V., and K. Sundaresan, "Deterministic Address Mapping to Reduce Logging in Carrier Grade NAT Deployments", draft-donley-behave-deterministic-cgn-01 (work in progress), January 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-ipv6-00 (work in progress), November 2011.

[I-D.ietf-pcp-base]
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-22 (work in progress), January 2012.

[I-D.ietf-softwire-public-4over6]
Cui, Y., Wu, J., Wu, P., Metz, C., Vautrin, O., and Y. Lee, "Public IPv4 over Access IPv6 Network", draft-ietf-softwire-public-4over6-00 (work in progress), September 2011.

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

[I-D.sun-v6ops-laft6]
Sun, Q. and C. Xie, "LAFT6: Lightweight address family transition for IPv6", draft-sun-v6ops-laft6-01 (work in progress), March 2011.

[I-D.tsou-pcp-natcoord]
Zhou, C., Tsou, T., Deng, X., Boucadair, M., and Q. Sun, "Using PCP To Coordinate Between the CGN and Home Gateway Via Port Allocation", draft-tsou-pcp-natcoord-04 (work in progress), January 2012.

[I-D.zhou-softwire-b4-nat]
Deng, X., Boucadair, M., and C. Zhou, "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