Softwire Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

Y. Cui
Tsinghua University
Q. Sun
China Telecom
M. Boucadair
France Telecom
T. Tsou
Huawei Technologies
Y. Lee
Comcast
I. Farrer
Deutsche Telekom AG
February 14, 2014

Lightweight 4over6: An Extension to the DS-Lite Architecture
draft-ietf-softwire-lw4over6-07

Abstract

Dual-Stack Lite (RFC 6333) describes an architecture for transporting IPv4 packets over an IPv6 network. This document specifies an extension to DS-Lite called Lightweight 4over6 which moves the Network Address and Port Translation (NAPT) function from the centralized DS-Lite tunnel concentrator to the tunnel client located in the Customer Premises Equipment (CPE). This removes the requirement for a Carrier Grade NAT function in the tunnel concentrator and reduces the amount of centralized state that must be held to a per-subscriber level. In order to delegate the NAPT function and make IPv4 Address sharing possible, port-restricted IPv4 addresses are allocated to the CPEs.

Status of This Memo

This Internet-Draft is submitted 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 August 18, 2014.

Table of Contents

1. Introduction

Dual-Stack Lite (DS-Lite, [RFC6333]) defines a model for providing IPv4 access over an IPv6 network using two well-known technologies:

IP in IP [RFC2473] and Network Address Translation (NAT). The DS-Lite architecture defines two major functional elements as follows:

Basic Bridging BroadBand element: A B4 element is a function implemented on a dual-stack capable node, either a directly connected device or a CPE, that creates a tunnel to an AFTR.

Address Family Transition Router: An AFTR element is the combination of an IPv4-in-IPv6 tunnel endpoint and an IPv4-IPv4 NAT implemented on the same node.

As the AFTR performs the centralized NAT44 function, it dynamically assigns public IPv4 addresses and ports to requesting host's traffic (as described in [RFC3022]). To achieve this, the AFTR must dynamically maintain per-flow state in the form of active NAPT sessions. For service providers with a large number of B4 clients, the size and associated costs for scaling the AFTR can quickly become prohibitive. It can also place a large NAPT logging overhead upon the service provider in countries where legal requirements mandate this.

This document describes a mechanism called Lightweight 4 over 6 (lw4o6), which provides a solution for these problems. By relocating the NAPT functionality from the centralized AFTR to the distributed B4s, a number of benefits can be realised:

- NAPT44 functionality is already widely supported and used in today's CPE devices. Lw4o6 uses this to provide private<->public NAPT44, meaning that the service provider does not need a centralized NAT44 function.

- The amount of state that must be maintained centrally in the AFTR can be reduced from per-flow to per-subscriber. This reduces the amount of resources (memory and processing power) necessary in the AFTR.

- The reduction of maintained state results in a greatly reduced logging overhead on the service provider.

Operator's IPv6 and IPv4 addressing architectures remain independent of each other. Therefore, flexible IPv4/IPv6 addressing schemes can be deployed.

Lightweight 4over6 provides a solution for a hub-and-spoke softwire architecture only. It does not offer direct, meshed IPv4

connectivity between subscribers without packets traversing the AFTR. If this type of meshed interconnectivity is required, [I-D.ietf-softwire-map] provides a suitable solution.

The tunneling mechanism remains the same for DS-Lite and Lightweight 4over6. This document describes the changes to DS-Lite that are necessary to implement Lightweight 4over6. These changes mainly concern the configuration parameters and provisioning method necessary for the functional elements.

Lightweight 4over6 features keeping per-subscriber state in the service provider's network. It is categorized as Binding approach in [I-D.ietf-softwire-unified-cpe] which defines a unified IPv4-in-IPv6 Softwire CPE.

This document is an extended case, which covers address sharing for [RFC7040]. 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 the 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 (lw4o6): An IPv4-over-IPv6 hub and spoke mechanism, which extends DS-Lite by moving the IPv4 translation (NAPT44) function from the AFTR to the B4.

Lightweight B4 (lwB4): A B4 element (Basic Bridging BroadBand element [RFC6333]), which supports Lightweight 4over6 extensions. An lwB4 is a function implemented on a dual-stack capable node, (either a directly connected device or a CPE), that supports port-restricted IPv4 address allocation, implements NAPT44

	functionality and creates a tunnel to an lwAFTR
Lightweight AFTR (lwAFTR):	An AFTR element (Address Family Transition Router element [RFC6333]), which supports Lightweight 4over6 extension. An lwAFTR is an IPv4-in-IPv6 tunnel endpoint which maintains per-subscriber address binding only and does not perform a NAPT44 function.
Restricted Port-Set:	A non-overlapping range of allowed external ports allocated to the lwB4 to use for NAPT44. Source ports of IPv4 packets sent by the B4 must belong to the assigned port-set. The port set is used for all port aware IP protocols (TCP, UDP, SCTP etc.)
Port-restricted IPv4 Address:	A public IPv4 address with a restricted port-set. In Lightweight 4over6, multiple B4s may share the same IPv4 address, however, their port-sets must be non-overlapping.

Throughout the remainder of this document, the terms B4/AFTR should be understood to refer specifically to a DS-Lite implementation. The terms lwB4/lwAFTR refer to a Lightweight 4over6 implementation.

4. Lightweight 4over6 Architecture

The Lightweight 4over6 architecture is functionally similar to DS-Lite. lwB4s and an lwAFTR are connected through an IPv6-enabled network. Both approaches use an IPv4-in-IPv6 encapsulation scheme to deliver IPv4 connectivity. The following figure shows the data plane with the main functional change between DS-Lite and lw4o6:

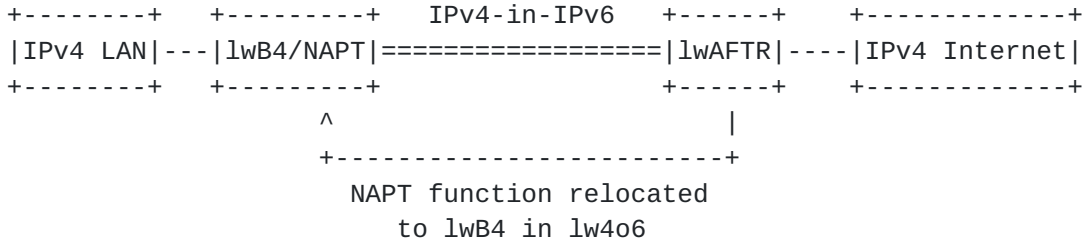

Figure 1 Lightweight 4over6 Data Plane Overview

There are three main components in the Lightweight 4over6 architecture:

- The lwB4, which performs the NAPT function and encapsulation/de-capsulation IPv4/IPv6.

- The lwAFTR, which performs the encapsulation/de-capsulation IPv4/IPv6.

- The provisioning system, which tells the lwB4 which IPv4 address and port set to use.

The lwB4 differs from a regular B4 in that it now performs the NAPT functionality. This means that it needs to be provisioned with the public IPv4 address and port set it is allowed to use. This information is provided though a provisioning mechanism such as DHCP, PCP or TR-69.

The lwAFTR needs to know the binding between the IPv6 address of each subscriber and the IPv4 address and port set allocated to that subscriber. This information is used to perform ingress filtering upstream and encapsulation downstream. Note that this is per-subscriber state as opposed to per-flow state in the regular AFTR case.

The consequence of this architecture is that the information maintained by the provisioning mechanism and the one maintained by the lwAFTR MUST be synchronized (See figure 2). The details of this synchronization depend on the exact provisioning mechanism and will be discussed in a companion document.

The solution specified in this document allows the assignment of either a full or a shared IPv4 address requesting CPEs. [RFC7040] provides a mechanism for assigning a full IPv4 address only.

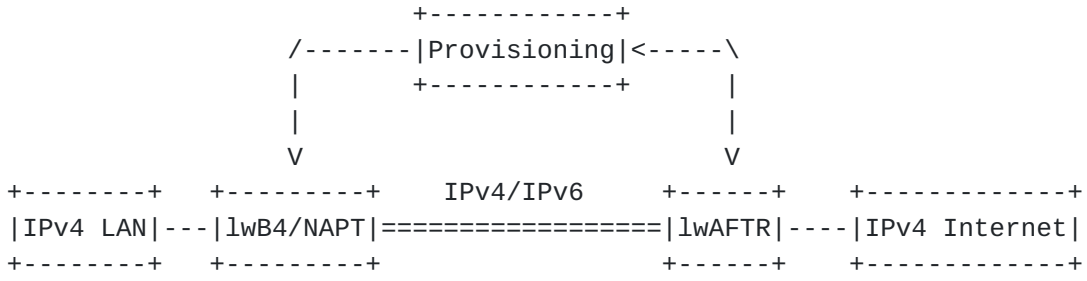

Figure 2 Lightweight 4over6 Provisioning Synchronization

5. Lightweight B4 Behavior

5.1. Lightweight B4 Provisioning with DHCPv6

With DS-Lite, the B4 element only needs to be configured with a single DS-Lite specific parameter so that it can set up the softwire (the IPv6 address of the AFTR). Its IPv4 address can be taken from the well-known range 192.0.0.0/29.

In lw4o6, due to the distributed nature of the NAPT function, a number of lw4o6 specific configuration parameters must be provisioned to the lwB4. These are:

- IPv6 Address for the lwAFTR
- IPv4 External (Public) Address for NAPT44
- Restricted port-set to use for NAPT44

For DHCPv6 based configuration of these parameters, the lwB4 SHOULD implement OPTION_S46_CONT_LW as described in section 6.3 of [I-D.ietf-softwire-map-dhcp]. This means that the lifetime of the softwire and the derived configuration information (e.g. IPv4 shared address, IPv4 address) is bound to the lifetime of the DHCPv6 lease. If stateful IPv4 configuration or additional IPv4 configuration information is required, DHCPv4 [RFC2131] must be used.

Some other mechanisms which may be adapted for the provisioning of IPv4 addresses and port-sets are described in section 7 below.

An IPv6 address from an assigned prefix is also required for the lwB4 to use as the encapsulation source address for the softwire. In order to enable end-to-end provisioning, the IPv6 address is constructed by taking a /64 prefix assigned to the WAN interface and suffixing 64-bits for the interface identifier. As there may be multiple WAN prefixes, of which only one can be used for lw4o6, the CPE creates a new WAN prefix specifically for use as the tunnel source address. The /128 prefix is then constructed in the same manner as [I-D.ietf-softwire-map]:

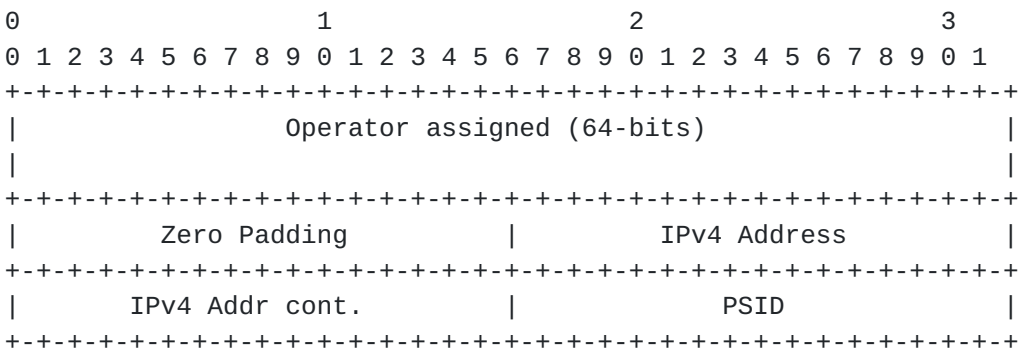

Figure 3 Construction of the lw4o6 /128 Prefix

Padding: Padding (all zeros)

IPv4 Address: Public IPv4 address allocated to the client

PSID: Port Set ID allocated to the client, left padded with zeros to 16-bits. If no PSID is provisioned, all zeros.

In the event that the lwB4's encapsulation source address is changed for any reason (such as the DHCPv6 lease expiring), the lwB4's dynamic provisioning process must be re-initiated.

An lwB4 MUST support dynamic port-restricted IPv4 address provisioning. The port set algorithm for provisioning this is described in Section 5.1 of [I-D.ietf-softwire-map]. For lw4o6, the number of a-bits SHOULD be 0.

In the event that the lwB4 receives an ICMPv6 error message (type 1, code 5) originating from the lwAFTR, the lwB4 SHOULD interpret this to mean that no matching entry in the lwAFTR's binding table has been found. The lwB4 MAY then re-initiate the dynamic port-restricted provisioning process. The lwB4's re-initiation policy SHOULD be configurable.

The DNS considerations described in Section 5.5 and Section 6.4 of [RFC6333] SHOULD be followed.

5.2. Lightweight B4 Data Plane Behavior

Several sections of [RFC6333] provide background information on the B4's data plane functionality and MUST be implemented by the lwB4 as they are common to both solutions. The relevant sections are:

5.2 Encapsulation	Covering encapsulation and de-capsulation of tunneled traffic

5.3 Fragmentation and Reassembly	Covering MTU and fragmentation considerations (referencing [RFC2473]), with the exception noted below.
7.1 Tunneling	Covering tunneling and traffic class mapping between IPv4 and IPv6 (referencing [RFC2473] and [RFC4213])

The lwB4 element performs IPv4 address translation (NAPT44) as well as encapsulation and de-capsulation. It runs standard NAPT44 [RFC3022] using the allocated port-restricted address as its external IPv4 address and port numbers.

The working flow of the lwB4 is illustrated in figure 4.

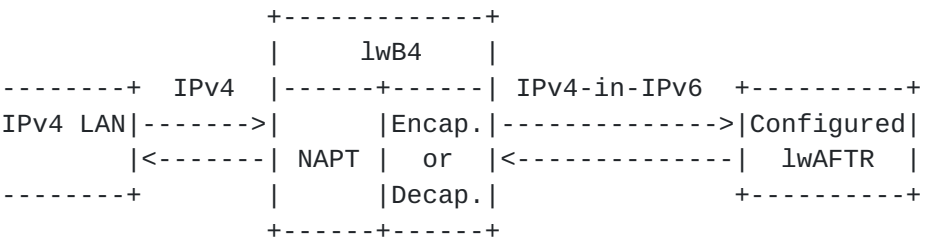

Figure 4 Working Flow of the lwB4

Internally connected hosts source IPv4 packets with an [RFC1918] address. When the lwB4 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 lwAFTR's IPv6 address and the source IPv6 address is the lwB4's IPv6 tunnel endpoint address. Finally, the lwB4 forwards the encapsulated packet to the configured lwAFTR.

When the lwB4 receives an IPv4-in-IPv6 packet from the lwAFTR, it de-capsulates the IPv4 packet from the IPv6 packet. Then, it performs NAPT44 translation on the destination address and port, based on the available information in its local NAPT44 table.

If the IPv6 source address does not match the configured lwAFTR address, then the packet MUST be discarded. If the decapsulated IPv4 packet does not match the lwB4's configuration (i.e. invalid destination IPv4 address or port) then the packet MUST be dropped. An ICMPv4 error message (type 13 - Communication Administratively Prohibited) message MAY be sent back to the lwAFTR. The ICMP policy SHOULD be configurable.

The lwB4 is responsible for performing ALG functions (e.g., SIP, FTP), and other NAPT traversal mechanisms (e.g., UPnP, NAPT-PMP, manual binding configuration, PCP) for the internal hosts. This requirement is typical for NAPT44 gateways available today.

It is possible that a lwB4 is co-located in a host. In this case, the functions of NAPT44 and encapsulation/de-capsulation are implemented inside the host.

5.2.1. Changes to RFC2473 and RFC6333 Fragmentation Behaviour

5.2.1.1. Processing of Incoming IPv4 Fragments Encapsulated in IPv6

The following process for handling fragmented packets is taken from Section 3.4 of [RFC6146], adapted for use with fragmentation of encapsulated IPv4 tunnel traffic and NAT44.

On receiving an encapsulated packet containing an IPv4 fragment, then more processing may be needed than for non-fragmented payloads. This specification leaves open the exact details of how the lwB4 handles incoming encapsulated IPv6 packets containing IPv4 fragments, and simply requires that the external behavior of the lwB4 be compliant with the following conditions:

The lwB4 MUST handle encapsulated IPv4 fragments. In particular, the lwB4 MUST handle fragments arriving out of order, conditional on the following:

- The lwB4 MUST limit the amount of resources devoted to the storage of fragmented packets in order to protect from DoS attacks.

- As long as the lwB4 has available resources, the lwB4 MUST allow the fragments to arrive over a time interval. The time interval SHOULD be configurable and the default value MUST be of at least FRAGMENT_MIN.

- The lwB4 MAY require that the UDP, TCP, or ICMP header be completely contained within the fragment that contains fragment offset equal to zero.

For incoming packets carrying TCP or UDP IPv4 fragments with a non-zero checksum, after de-capsulation, the lwB4 MAY elect to queue the fragments as they arrive and perform NAPT44 on all fragments at the same time. In this case, the incoming 5-tuple is determined by extracting the appropriate fields from the received packet, as described in [RFC2473]. Alternatively, a lwB4 MAY translate the de-capsulated fragments as they arrive, by storing information that allows it to compute the 5-tuple for fragments other than the first.

In the latter case, subsequent fragments may arrive before the first, and the rules (in the bulleted list above) about how the lwB4 handles (out-of-order) fragments apply.

For incoming de-capsulated IPv4 packets carrying UDP packets with a zero checksum, if the lwB4 has enough resources, the lwB4 MUST reassemble the packets and MUST calculate the checksum. If the lwB4 does not have enough resources, then it MUST silently discard the packets. The handling of fragmented and un-fragmented UDP packets with a zero checksum as specified above deviates from that specified in [RFC6145].

Implementers of an lwB4 should be aware that there are a number of well-known attacks against IP fragmentation; see [RFC1858] and [RFC3128]. Implementers should also be aware of additional issues with reassembling packets at high rates, described in [RFC4963].

5.2.1.2. Processing of Outbound IPv4 Packets Requiring Fragmentation for Encapsulation

When an lwB4 receives an IPv4 packet from a connected host that exceeds the IPv6 MTU size after encapsulation, the lwB4 SHOULD fragment the IPv4 packet before encapsulation. This lwB4 behavior avoids IPv6 fragmentation, so that the lwAFTR is not required to re-assemble fragmented IPv6 packets. If the the Don't Fragment (DF) bit is set in the IPv4 packet header (e.g. for PMTUD discovery), then the IPv4 packet is dropped by the lwB4 and an ICMP Fragmentation Needed (Type 3, Code 4) with the correct tunnel MTU is sent.

6. Lightweight AFTR Behavior

6.1. Binding Table Maintenance

The lwAFTR maintains an address binding table containing the binding between the lwB4's IPv6 address, the allocated IPv4 address and restricted port-set. Unlike the DS-Lite extended binding table defined in section 6.6 of [RFC6333] which is a 5-tuple NAPT table, each entry in the Lightweight 4over6 binding table contains the following 3-tuples:

- IPv6 Address for a single lwB4
- Public IPv4 Address
- Restricted port-set

The entry has two functions: the IPv6 encapsulation of inbound IPv4 packets destined to the lwB4 and the validation of outbound IPv4-in-IPv6 packets received from the lwB4 for de-capsulation.

The lwAFTR does not perform NAPT and so does not need session entries.

The lwAFTR MUST synchronize the binding information with the port-restricted address provisioning process. If the lwAFTR 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).

If the lwAFTR participates in the port-restricted provisioning process, then its binding table MUST be created as part of this process.

For all provisioning processes, the lifetime of binding table entries MUST be synchronized with the lifetime of address allocations.

6.2. lwAFTR Data Plane Behavior

Several sections of [RFC6333] provide background information on the AFTR's data plane functionality and MUST be implemented by the lwAFTR as they are common to both solutions. The relevant sections are:

6.2 Encapsulation	Covering encapsulation and de-capsulation of tunneled traffic
6.3 Fragmentation and Reassembly	Fragmentation and re-assembly considerations (referencing [RFC2473])
7.1 Tunneling	Covering tunneling and traffic class mapping between IPv4 and IPv6 (referencing [RFC2473] and [RFC4213])

When the lwAFTR receives an IPv4-in-IPv6 packet from an lwB4, it de-capsulates the IPv6 header and verifies the source addresses and port in the binding table. If both the source IPv4 and IPv6 addresses match a single entry in the binding table and the source port is in the allowed port-set for that entry, the lwAFTR forwards the packet to the IPv4 destination.

If no match is found (e.g., no matching IPv4 address entry, port out of range, etc.), the lwAFTR MUST discard or implement a policy (such

as redirection) on the packet. An ICMPv6 type 1, code 5 (source address failed ingress/egress policy) error message MAY be sent back to the requesting lwB4. The ICMP policy SHOULD be configurable.

When the lwAFTR receives an inbound IPv4 packet, it uses the IPv4 destination address and port to lookup the destination lwB4's IPv6 address in its binding table. If a match is found, the lwAFTR encapsulates the IPv4 packet. The source is the lwAFTR's IPv6 address and the destination is the lwB4's IPv6 address from the matched entry. Then, the lwAFTR forwards the packet to the lwB4 natively over the IPv6 network.

If no match is found, the lwAFTR MUST discard the packet. An ICMPv4 type 3, code 1 (Destination unreachable, host unreachable) error message MAY be sent back. The ICMP policy SHOULD be configurable.

The lwAFTR MUST support hairpinning of traffic between two lwB4s, by performing de-capsulation and re-encapsulation of packets. The hairpinning policy MUST be configurable.

7. Additional IPv4 address and Port Set Provisioning Mechanisms

In addition to the DHCPv6 based mechanism described in section 5.1, several other IPv4 provisioning protocols have been suggested. These protocols MAY be implemented. These alternatives include:

- DHCPv4 over DHCPv6: [I-D.ietf-dhc-dhcpv4-over-dhcpv6] describes implementing DHCPv4 messages over an IPv6 only service providers network. This enables leasing of IPv4 addresses and makes DHCPv4 options available to the DHCPv4 over DHCPv6 client.

- PCP[RFC6887]: an lwB4 MAY use [I-D.ietf-pcp-port-set] to retrieve a restricted IPv4 address and a set of ports.

In a Lightweight 4over6 domain, the binding information MUST be aligned between the lwB4s, the lwAFTRs and the provisioning server.

8. ICMP Processing

For both the lwAFTR and the lwB4, ICMPv6 MUST be handled as described in [RFC2473].

ICMPv4 does not work in an address sharing environment without special handling [RFC6269]. Due to the port-set style address sharing, Lightweight 4over6 requires specific ICMP message handling not required by DS-Lite.

8.1. ICMPv4 Processing by the lwAFTR

For inbound ICMP messages The following behavior SHOULD be implemented by the lwAFTR to provide ICMP error handling and basic remote IPv4 service diagnostics for a port restricted CPE:

1. Check the ICMP Type field.

2. If the ICMP type is set to 0 or 8 (echo reply or request), then the lwAFTR MUST take the value of the ICMP identifier field as the source port, and use this value to lookup the binding table for an encapsulation destination. If a match is found, the lwAFTR forwards the ICMP packet to the IPv6 address stored in the entry; otherwise it MUST discard the packet.

3. If the ICMP type field is set to any other value, then the lwAFTR 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; otherwise it MUST discard the packet.

Additionally, the lwAFTR MAY implement:

o Discarding of all inbound ICMP messages.

The ICMP policy SHOULD be configurable.

8.2. ICMPv4 Processing by the lwB4

The lwB4 SHOULD implement the requirements defined in [RFC5508] for ICMP forwarding. For ICMP echo request packets originating from the private IPv4 network, the lwB4 SHOULD implement the method described in [RFC6346] and use an available port from its port-set as the ICMP Identifier.

9. Security Considerations

As the port space for a subscriber shrinks due to address sharing, the randomness for the port numbers of the subscriber is decreased significantly. This means 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.

10. IANA Considerations

This document does not include an IANA request.

11. 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

Qi Sun

Tsinghua University

Beijing 100084

P.R.China

Phone: +86-10-62785822

Email: sunqi@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

Alain Durand

Juniper Networks

1194 North Mathilda Avenue

Sunnyvale, CA 94089-1206

USA

Email: adurand@juniper.net

Reinaldo Penno

Cisco Systems, Inc.

170 West Tasman Drive

San Jose, California 95134

USA

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

12. Acknowledgement

The authors would like to thank Ole Troan, Ralph Droms and Suresh Krishnan 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].

13. References

13.1. Normative References

[I-D.ietf-softwire-map-dhcp]
Mrugalski, T., Troan, O., Dec, W., Bao, C., leaf.yeh.sdo@gmail.com, l., and X. Deng, "DHCPv6 Options for configuration of Softwire Address and Port Mapped Clients", draft-ietf-softwire-map-dhcp-06 (work in progress), November 2013.

[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.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

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

[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.

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

13.2. Informative References

[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-11 (work in progress), February 2013.

[I-D.ietf-dhc-dhcpv4-over-dhcpv6]
Sun, Q., Cui, Y., Siodelski, M., Krishnan, S., and I. Farrer, "DHCPv4 over DHCPv6 Transport", draft-ietf-dhc-dhcpv4-over-dhcpv6-04 (work in progress), January 2014.

[I-D.ietf-pcp-port-set]
Qiong, Q., Boucadair, M., Sivakumar, S., Zhou, C., Tsou, T., and S. Perreault, "Port Control Protocol (PCP) Extension for Port Set Allocation", draft-ietf-pcp-port-set-04 (work in progress), November 2013.

[I-D.ietf-softwire-map-dhcp]
Mrugalski, T., Troan, O., Dec, W., Bao, C., leaf.yeh.sdo@gmail.com, l., and X. Deng, "DHCPv6 Options for configuration of Softwire Address and Port Mapped Clients", draft-ietf-softwire-map-dhcp-06 (work in progress), November 2013.

[I-D.ietf-softwire-map]
Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, "Mapping of Address and Port with Encapsulation (MAP)", draft-ietf-softwire-map-10 (work in progress), January 2014.

[I-D.ietf-softwire-unified-cpe]
Boucadair, M., Farrer, I., Perreault, S., and S. Sivakumar, "Unified IPv4-in-IPv6 Softwire CPE", draft-ietf-softwire-unified-cpe-01 (work in progress), May 2013.

[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.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.

[RFC1858] Ziemba, G., Reed, D., and P. Traina, "Security Considerations for IP Fragment Filtering", RFC 1858, October 1995.

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

[RFC3128] Miller, I., "Protection Against a Variant of the Tiny Fragment Attack (RFC 1858)", RFC 3128, June 2001.

[RFC4963] Heffner, J., Mathis, M., and B. Chandler, "IPv4 Reassembly Errors at High Data Rates", RFC 4963, July 2007.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, April 2011.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

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

[RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[RFC7040] Cui, Y., Wu, J., Wu, P., Vautrin, O., and Y. Lee, "Public IPv4-over-IPv6 Access Network", RFC 7040, November 2013.

Authors' Addresses

Yong Cui
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
CTO-ATI, Landgrabenweg 151
Bonn, NRW 53227
Germany

Email: ian.farrer@telekom.de