Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 01, 2014

Y. Chen
J. Wu
Tsinghua University
X. Tang
G. Zhou
China Unicom Research Institute
August 28, 2013

Gateway-Initiated 4over6 Deployment
draft-chen-softwire-gw-init-4over6-02

Abstract

Gateway-Initiated 4over6 is a variant of Lightweight 4over6. A Lightweight B4 in Lightweight 4over6 mechanism is a router which acts as a tunnel initiator for the IPv4-in-IPv6 tunnel. This mechanism mainly focuses on the scenario in which an IPv4 address and related configuration information is configured to the device behind Lightweight B4. Gateway-Initiated 4over6 uses the full IPv4 address rather than a shared address. This enables an unmodified end server or host that is behind a Lightweight B4 to get access to the IPv4 Internet through an IPv6 network.

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 March 01, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents

(http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 2
3. Requirements Language 3
4. GI-4over6 Architecture 3
5. GI-4over6 in ICP Network 4
5.1. Static Configuration to Establish Tunnel 4
5.2. Dynamic Configuration to Establish Tunnel 5
6. 4over6 Gateway Data Plane Behaviors 5
7. Security Considerations 5
8. IANA Considerations 5
9. References 5
9.1. Normative References 5
9.2. Informative References 5
Authors' Addresses 6

1. Introduction

In typical use case of Lightweight 4over6 (Lw4over6) ([I-D.ietf-softwire-lw4over6]), IPv4 address (and available port set) is provisioned to the Lightweight B4 (LwB4), the tunnel initiator. However, there are some cases in which IPv4 address and related configuration are not be provisioned to LwB4, but the end device behind it. There is a typical scenario in this case, that is Lw4over6 is used in an Internet Content Provider (ICP) network, and the device behind LwB4 is an ICP server.

Gateway-Initiated 4over6 (GI-4over6) is a variant of Lw4over6. It mainly focuses on the scenario in which an IPv4 address and related configuration information is provisioned to the device behind LwB4. Provisioning full address is preferred to provisioning shared address (port-restricted address) in GI-4over6. It enables an unmodified IPv4 device that behind the LwB4 to get access to IPv4 Internet through IPv6 network.

2. Terminology

This document uses the terms defined in [I-D.ietf-softwire-lw4over6].

The other terms used are defined as follows:

- o End device: The device in the IPv4 network behind the 4over6 gateway. It can be an IPv4-only or a dual-stack device.
- o End server: The end device in an ICP network is supposed to be an end server.
- o 4over6 gateway: The dual-stack gateway device located at the border of both IPv4 and IPv6 networks. It should be configured with an IPv4 address and the IPv6 address of LwAFTR, and act as the LwB4 on the data plane.

3. Requirements Language

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].

4. GI-4over6 Architecture

The general architecture of GI-4over6 is illustrated as Figure 1. The 4over6 gateway is a dual-stack gateway device which establishes IPv4-in-IPv6 tunnel with the Lightweight Address Family Transition Router (LwAFTR) and performs the LwB4 function on data plane. The LwAFTR is a dual-stack border router deployed at the edge of the IPv6 network and the Internet. The IPv4 network can be either an ICP network, or a customer network of an ISP. The IPv6 network can be either an ICP access network or an ISP access network. Either or both of these networks could be dual-stack.

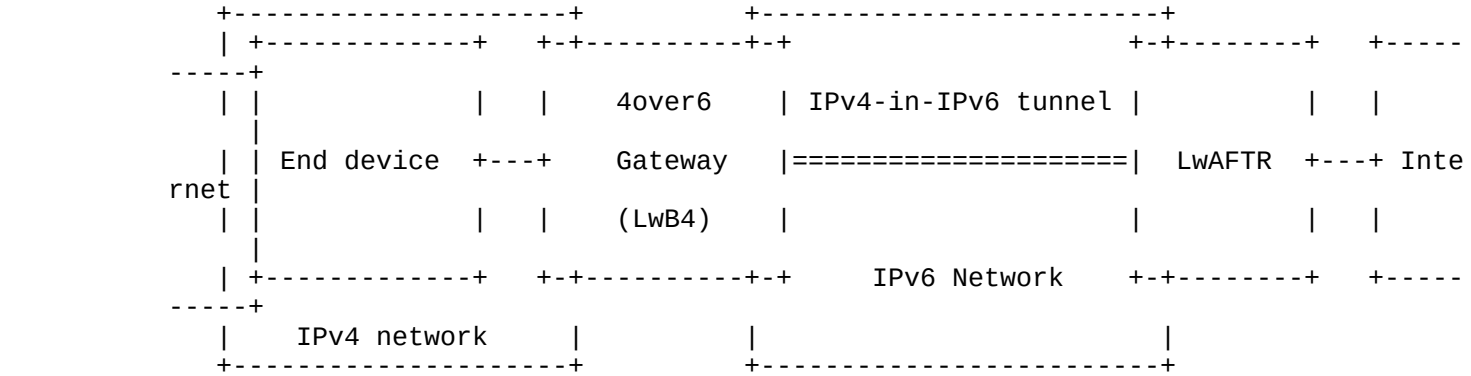


Figure 1 GI-4over6 Architecture

The 4over6 gateway is configured with an public IPv4 address on its

"left" side, an IPv6 address on its "right" side, either by static (in ICP network) or dynamic (in customer network) way. It is also configured with the IPv6 address of LwAFTR as the address of the tunnel endpoint. Each end device has a public IPv4 address with all ports (0-65535) available, hence there is no need to implement NAPT44 on the 4over6 gateway.

One typical scenario of this framework is that using Lw4over6 in an ICP network. There might be other similar scenarios, and they could be included in this document in the future.

5. GI-4over6 in ICP Network

Considering an ISP that plans to update its network to IPv6, one of the major issues it may be faced with is the update of its ICP network. If the ICP network is to be updated to run IPv6, the server in the network should also be updated to support IPv6. Obviously it is not trivial to update upper layer service running on the server to support a network layer protocol. It’s ideal if the ICP access network is updated to IPv6, but still capable of providing the server with access to IPv4 Internet, meanwhile the ICP network (and the servers inside) stay unmodified.

In this scenario, the end server has already been configured with a full public IPv4 address, and it’s expected to stay unchanged during the update of the network. It has also been configured with other IPv4 related configurations like the network mask of the IPv4 ICP network, the IPv4 address of DNS server, etc.

The 4over6 gateway has already been configured with the routing to the end server. It MUST establish the IPv4-in-IPv6 tunnel with the LwAFTR, in order to forward the IPv4 traffics between the end server and the IPv4 Internet. The establishment of the IPv4-in-IPv6 tunnel could be done either by static - the most likely way - or dynamic configuration.

5.1. Static Configuration to Establish Tunnel

The LwAFTR is statically configured with the binding of the public IPv4 address of the end server, the available port set (0-65535), and IPv6 address of the 4over6 gateway in its binding table statically.

In a more general case, the addresses of servers behind the same 4over6 gateway can aggregate. And as the 4over6 gateway and the LwAFTR are both managed by the ISP, people who configure the LwAFTR are usually aware of the routing to the ICP network behind the 4over6 gateway. Hence the LwAFTR can be configured with the following binding: the network prefix of the ICP network, the available port

set (0-65535), and IPv6 address of the 4over6 gateway.

5.2. Dynamic Configuration to Establish Tunnel

Dynamic configuration could be adopted in case the static configuration is not feasible or practical.

The 4over6 gateway MUST inform the LwAFTR of all of its IPv4 routing information (i.e. the whole IPv4 routing table). The detail of this process could be clarified in related draft in future.

Once the LwAFTR received the routing information from the 4over6 gateway, it should add the entry(s) into its binding table, with the given routing information. The binding may looks like: the ICP network prefix, available port set (0-65535), the IPv6 address of the 4over6 gateway.

6. 4over6 Gateway Data Plane Behaviors

The 4over6 gateway must perform the LwB4 function on the data plane. The data plane behavior of 4over6 gateway uses the description in section 5.2 of [I-D.ietf-softwire-lw4over6]. However, there is no need to implement NAPT44 function on 4over6 gateway, because each end server behind the 4over6 gateway has a public IPv4 address with all ports available.

7. Security Considerations

TBD

8. IANA Considerations

This document does not include an IANA request.

9. References

9.1. Normative References

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-01 (work in progress), July 2013.

9.2. Informative References

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

Authors' Addresses

Yuchi Chen
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86 10 6278 5822
Email: chenycmx@gmail.com

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

Phone: +86 10 6278 5983
Email: jianping@cernet.edu.cn

Xiongyan Tang
China Unicom Research Institute
33 Erlong Road, Xicheng District
Beijing 100032
P.R.China

Phone: +86 10 6652 2558
Email: tangxy@chinaunicom.cn

Guangtao Zhou
China Unicom Research Institute
9 Shouti South Road, Haidian District
Beijing 100048
P.R.China

Phone: +86 10 6789 9600
Email: zhouguangtao@chinaunicom.cn

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: March 30, 2014

E. Cordeiro
R. Carnier
A. Moreiras
NIC.br

September 26, 2013

Experience from MAP-T Testing
draft-cordeiro-softwire-experience-mapt-02

Abstract

This document describes the testing result of a network using MAP-T dual translation solution, by providing an overview of user applications' behavior with a shared IPv4 address.

The MAP-T software is from CERNET Center and the test environment is on NIC.br network with real and virtualized machines.

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 March 30, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
1.1. Requirements Language 4
2. Testbed Overview 5
2.1. Network Topology 5
2.2. Configurantion files 6
2.2.1. MAP-T Core 6
2.2.2. IPv6 only Router 7
2.2.3. MAP-T CPE 1 8
2.2.4. Host 1 8
3. Operating systems used in this test 9
3.1. Linux 9
3.2. Windows 7 9
3.3. Windows XP 9
4. Applications Testing Summary 10
5. Effect analysis 13
5.1. User experience 13
5.2. Testing summary 13
6. Acknowledgements 15
7. IANA Considerations 16
8. Security Considerations 17
9. References 18
9.1. Normative References 18
9.2. Informative References 18
Appendix A. Applications Testing Details 19
A.1. Browsers 19
A.1.1. Google Chrome 19
A.1.2. Mozilla Firefox 20
A.1.3. Internet Explorer 21
A.1.4. Safari 22
A.1.5. Lynx (text browser) 23
A.2. Web browsing 24
A.2.1. www.google.com 24
A.2.2. www.msn.com 25
A.3. Web dynamic content 26
A.3.1. Flash Player 26
A.3.2. Silverlight 27
A.3.3. Java applets 28
A.3.4. HTML5 websites 29
A.4. Video stream websites 30
A.4.1. www.youtube.com 30
A.4.2. www.dailymotion.com 31
A.4.3. www.zappiens.br 32

A.5. Social networking websites 33
A.5.1. www.facebook.com 33
A.5.2. www.twitter.com 34
A.5.3. www.orkut.com 35
A.6. Webmails 36
A.6.1. www.gmail.com 36
A.6.2. www.hotmail.com 37
A.7. Real-time Internet text messaging (chat) website 38
A.7.1. Chat rooms of UOL content provider 38
A.8. Image hosting site 39
A.8.1. www.flickr.com 39
A.9. Communication protocols 40
A.9.1. Skype 40
A.9.2. Googletalk 41
A.9.3. Jabber (XMPP) 42
A.9.4. MSN Messenger (Microsoft Notification Protocol) . . . 43
A.9.5. IRC (Internet Relay Chat) 44
A.10. Torrent clients 45
A.10.1. Vuze 45
A.10.2. uTorrent 46
A.10.3. Ktorrent 47
A.10.4. Note aout bittorrent seeders 47
A.11. Remote access and file transfer software 48
A.11.1. ssh 48
A.11.2. ftp 49
A.11.3. Filezilla ftp 50
A.11.4. wget 51
A.12. Antivirus updates 52
A.12.1. Avira 52
A.12.2. AVG 53
A.12.3. Avast 54
A.13. Media player updates and video streaming 55
A.13.1. VLC 55
A.13.2. Realplayer 56
A.13.3. Windows Media Player 57
A.14. Network testing tools 58
A.14.1. ping 58
A.14.2. traceroute 59
A.14.3. tracert 60
Authors' Addresses 61

1. Introduction

This testing is based on most common applications used by home users. The main purpose is to check if those applications work correctly on a network using MAP-T [draft-mdt-softwire-map-translation-01].

Based on testing we know which applications could be used on a network with MAP-T and the impact on a typical Internet user in Brazil. The classification as a working application is based on user experience, not on network measurements.

1.1. Requirements Language

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 RFC 2119 [RFC2119].

2. Testbed Overview

2.1. Network Topology

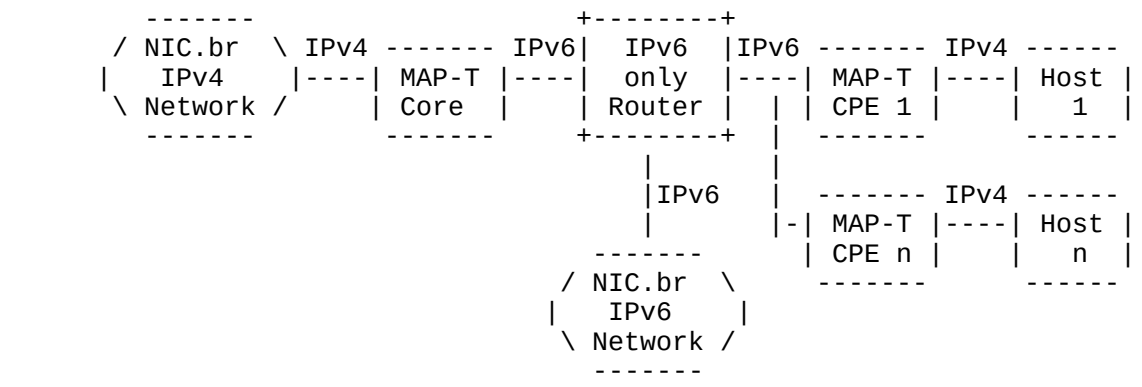


Figure 1

The MAP-T Core and MAP-T CPE are virtualized Linux machines running Fedora 11 and MAP-T 1.1 software developed by CERNET (http://mapt.ivi2.org:8039/mapt.html).

The host machine is in fact three virtualized machines, one with Linux Kubuntu 12.04, one with Windows 7 and one with Windows XP. The host machine is replaced in the topology to test each operating system.

The IPv6 router is a Linux machine with static routes to match the MAP-T network topology routes.

NIC.br IPv4 Network is a IPv4 network with public valid IPv4 address. For the MAP-T topology it is considered like IPv4 Internet, as this network topology is unknown for the testing network and can not be changed.

NIC.br IPv6 Network is a IPv6 network with public valid IPv6 address. For the MAP-T topology it is considered like IPv6 Internet, as this network topology is unknown for the testing network and can not be changed.

The host may have both IPv4 and IPv6 addresses, but to guarantee that the translation was being tested the host received only an IPv4 address. If the host is assigned with an IPv6 address, this address should have preference as specified on RFC6555 [RFC6555] and the translation might not be tested in some cases were the application or the content provider is available on IPv6.

2.2. Configurantion files

2.2.1. MAP-T Core

```
#!/bin/sh

./control stop

# configure system profile
echo 1 > /proc/sys/net/ipv4/ip_forward
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
echo 0 > /proc/sys/net/ipv6/conf/eth0/autoconf
echo 0 > /proc/sys/net/ipv6/conf/eth1/autoconf

# configure eth0 -- IPv6 interface
ifconfig eth0 down
ifconfig eth0 up
ifconfig eth0 inet6 add 2001:db8:6:e000::2/64
ip -6 route add 2001:db8:6:d600::/56 via 2001:db8:6:e000::1 dev eth0
route -A inet6 add default gw 2001:db8:6:e000::1

# configure eth1 -- IPv4 interface
ifconfig eth1 down
ifconfig eth1 up
ifconfig eth1 192.0.2.171/27
ip route add default via 192.0.2.161 dev eth1

./control start
./utils/ivictl -r -p 198.51.100.248/29 -P 2001:db8:6:d600::/56 -R 16
-M 2
./utils/ivictl -r -d -P 2001:db8:6:d6ff::/64
./utils/ivictl -s -i eth1 -I eth0

service iptables stop
service ip6tables stop
```

2.2.2. IPv6 only Router

```
#!/bin/sh

# configure system profile
echo 1 > /proc/sys/net/ipv4/ip_forward
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
echo 0 > /proc/sys/net/ipv6/conf/eth0/autoconf
echo 0 > /proc/sys/net/ipv6/conf/eth1/autoconf
echo 0 > /proc/sys/net/ipv6/conf/eth2/autoconf

# configure eth0 -- IPv6 interface to core
ifconfig eth0 down
ifconfig eth0 up
ifconfig eth0 inet6 add 2001:db8:6:e000::1/64

# configure eth1 -- IPv6 interface to cpe
ifconfig eth1 down
ifconfig eth1 up
ifconfig eth1 inet6 add 2001:db8:6:e001::1/64

# configure eth2 -- IPv6 interface gateway
ifconfig eth2 down
ifconfig eth2 up
ifconfig eth2 inet6 add 2001:db8:0:6160::ed19/64

ip -6 route add 2001:db8:6:d640::/64 via 2001:db8:6:e001::2 dev eth1
ip -6 route add 2001:db8:6:d6ff::/64 via 2001:db8:6:e000::2 dev eth0

ip -6 route add 2001:db8:6:e000::/64 dev eth0
ip -6 route add 2001:db8:6:e001::/64 dev eth1

ip -6 route add ::/0 via 2001:db8:0:6160::1ab6 dev eth2

service iptables stop
service ip6tables stop
```

2.2.3. MAP-T CPE 1

```
#!/bin/sh

./control stop

# configure system profile
echo 1 > /proc/sys/net/ipv4/ip_forward
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
echo 0 > /proc/sys/net/ipv6/conf/eth0/autoconf
echo 0 > /proc/sys/net/ipv6/conf/eth1/autoconf

# configure eth0 -- IPv6 interface
ip -6 link set eth0 down
ip -6 link set eth0 up
ip -6 addr add 2001:db8:6:e001::2/64 dev eth0
ip -6 route add default via 2001:db8:6:e001::1 dev eth0

# configure eth1 -- IPv4 interface
ip link set eth1 down
ip link set eth1 up
ip addr add 198.51.100.249/29 dev eth1

./control start
./utils/ivictl -r -d -P 2001:db8:6:d6ff::/64
./utils/ivictl -s -i eth1 -I eth0 -H -a 198.51.100.250/29
-P 2001:db8:6:d600::/56 -R 16 -M 2 -o 0 -c 1440

service iptables stop
service ip6tables stop
```

2.2.4. Host 1

The host could be Linux, Windows 7 or Windows XP, so there isn't a script for each of them, but the following configuration must be configured manually:

- IPv4 address: 198.51.100.250/29
- IPv4 gateway: 198.51.100.249
- IPv4 DNS: 8.8.8.8
- IPv6 is disabled

The communication using IPv6 from/to the host to/from the Internet has no limitation and is not impacted by the translation mechanism. The IPv6 is disabled to guarantee that the translation is being used.

3. Operating systems used in this test

3.1. Linux

O.S.	Linux
Details	Ubuntu 12.04 LTS Kernel 3.2.0-23
Architecture	32 bits

3.2. Windows 7

O.S.	Windows 7
Details	Windows 7 Ultimate
Architecture	64 bits

3.3. Windows XP

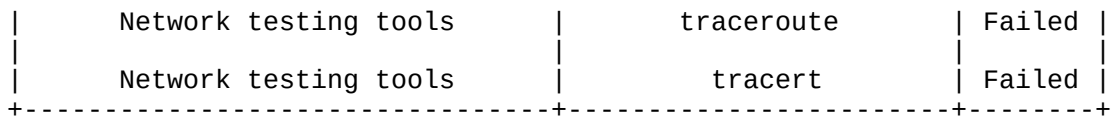
O.S.	Windows XP
Details	Windows XP Professional Service Pack 3
Architecture	32 bits

4. Applications Testing Summary

The table below contains the summary of the testing results. The details of each test is included on Appendix A.

Category	Application	Result
Browsers	Google Chrome	Passed
Browsers	Mozilla Firefo	Passed
Browsers	Internet Explorer	Passed
Browsers	Safari	Passed
Browsers	Lynx (text browser)	Passed
Web browsing	www.google.com	Passed
Web browsing	www.msn.com	Passed
Web dynamic content	Flash Player	Passed
Web dynamic content	Silverlight	Passed
Web dynamic content	Java applets	Passed
Web dynamic content	HTML5 websites	Passed
Video stream websites	www.youtube.com	Passed
Video stream websites	www.dailymotion.com	Passed
Video stream websites	www.zappiens.br	Passed
Social networking websites	www.facebook.com	Passed
Social networking websites	www.twitter.com	Passed
Social networking websites	www.orkut.com	Passed
Webmails	www.gmail.com	Passed
Webmails	www.hotmail.com	Passed
Real-time Internet text messaging (chat) website	Chat rooms of UOL content provider	Passed

Image hosting site	www.flickr.com	Passed
Communication protocols	Skype	Passed
Communication protocols	Googletalk	Passed
Communication protocols	Jabber (XMPP)	Passed
Communication protocols	MSN Messenger	Passed
Communication protocols	IRC	Passed
Torrent clients	Vuze	Passed
Torrent clients	uTorrent	Passed
Torrent clients	Ktorrent	Passed
Remote access and file transfer softwares	ssh	Passed
Remote access and file transfer softwares	ftp	Failed
Remote access and file transfer softwares	Filezilla ftp	Passed
Remote access and file transfer softwares	wget	Passed
Antivirus updates	Avira	Passed
Antivirus updates	AVG	Passed
Antivirus updates	Avast	Passed
Media player updates and video streaming	VLC	Passed
Media player updates and video streaming	Realplayer	Passed
Media player updates and video streaming	Windows Media Player	Passed
Network testing tools	ping outbound	Passed
Network testing tools	ping inbound	Failed


| Network testing tools | traceroute | Failed |
| Network testing tools | tracert | Failed |

5. Effect analysis

5.1. User experience

User experience can only be evaluated subjectively, there is no quantitative rule to define if the user experience is acceptable. Network delay, streaming experience and download time are similar to a network without MAP-T.

The user experience was very good. Almost all the software and websites worked correctly, the exception were the network traceroute and command line FTP.

The traceroute is only capable to reach the MAP-T CPE and receive a return message that the destination net is unreachable.

The command line FTP is capable to connect to the host with or without authentication, create, delete and navigate folders, but it is not capable list folder contents, to send or receive files in active mode.

The bittorrent applications can’t seed or share files, since there isn’t incoming connections to the host, it may cause some difficulties and low downloading speeds.

5.2. Testing summary

The working applications had no need of a special configuration to work.

The command line FTP doesn’t work correctly because active mode requeries incoming connections to specific ports without having a outbound connection on those ports. When the test was made on FTP passive mode on Linux, the FTP works correctly. When the test was made on FTP passive mode on Windows 7 and Windows XP, the FTP didn’t work correctly. When using FileZilla FTP in passive mode, FTP works on all tested operating systems.

The network traceroute doesn’t work in inbound or outbound directions because there is no continuity of the IPv4 network, as it is interrupted by an IPv6 only network. The tool is not capable to detect the hosts in this IPv6 only network and because of that the traceroute doesn’t succeed.

The testing was made with the version 1.1 of the MAP-T software developed by CERNET. In this version the MAP-T CPE uses a NAT44, so it is not possible to receive incoming connections even on the ports assigned to the host by the address plus port division. Because of

that is not possible to configure a server on the host. After those testing CERNET developed a new version of the software (2.2c) that is capable to receive incoming connections on some ports that are assigned to each CPE.

6. Acknowledgements

We would like to thank the CERNET folks for providing their MAP-T software for our tests.

We would like to thank NIC.br for offering the infrastructure for the testing.

Future tests will consider testing of MAP-E too that is now supported in the new version of CERNET’s MAP software. Other operating systems (Mac OS, Android, IOS etc) and devices (mobile phones, tablets, video games etc) should be tested too.

7. IANA Considerations

There are no new IANA considerations pertaining to this document.

8. Security Considerations

There are no new security considerations pertaining to this document.

9. References

9.1. Normative References

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

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

[RFC6791] Li, X., Bao, C., Wing, D., Vaithianathan, R., and G. Huston, "Stateless Source Address Mapping for ICMPv6 Packets", RFC 6791, November 2012.

9.2. Informative References

[I-D.boucadair-pcp-bittorrent]
Boucadair, M., Zheng, T., Deng, X., and J. Queiroz, "Behavior of BitTorrent service in PCP-enabled networks with Address Sharing", draft-boucadair-pcp-bittorrent-00 (work in progress), May 2012.

[I-D.ietf-softwire-map]
Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., and T. Murakami, "Mapping of Address and Port with Encapsulation (MAP)", draft-ietf-softwire-map-04 (work in progress), February 2013.

[I-D.xli-softwire-map-testing]
Li, X., Bao, C., Han, G., and W. Dec, "MAP Interoperability Testing Results", draft-xli-softwire-map-testing-01 (work in progress), January 2013.

Appendix A. Applications Testing Details

A.1. Browsers

A.1.1. Google Chrome

Test Item	Browsers
Sub-Item	Google Chrome
Test S.O.	Linux, Windows 7, Windows XP
Software Version	20.0.1132 (Linux), 21.0.1180 (Windows 7, Windows XP)
Test Objective	Check the browsing experience with Chrome browser behind the double translation of MAP-T.
Test Procedure	1.Open browser and input a content provider address. 2.After loading the page, load one of the last news page that have not been accessed yet. 3.Check whether contents can be accessed and visualized.
Expected Result	Browser can open and visualize content correctly.
Actual Result	Passed
Remarks	

A.1.2. Mozilla Firefox

Test Item	Browsers
Sub-Item	Mozilla Firefox
Test S.O.	Linux, Windows 7, Windows XP
Software Version	13.0.1 (Linux), 14.0.1 (Windows 7, Windows XP)
Test Objective	Check the browsing experience with Firefox browser behind the double translation of MAP-T.
Test Procedure	1.Open browser and input a content provider address. 2.After loading the page, load one of the last news pages that have not been accessed yet. 3.Check whether contents can be accessed and visualized.
Expected Result	Browser can open and visualize content correctly.
Actual Result	Passed
Remarks	

A.1.3. Internet Explorer

Test Item	Browsers
Sub-Item	Internet Explorer
Test S.O.	Windows 7, Windows XP
Software Version	8.0.7600 (Windows 7), 8.0.6001 (Windows XP)
Test Objective	Check the browsing experience with Internet Explorer behind the double translation of MAP-T.
Test Procedure	1.Open browser and input a content provider address. 2.After loading the page, load one of the last news pages that have not been accessed yet. 3.Check whether contents can be accessed and visualized.
Expected Result	Browser can open and visualize content correctly.
Actual Result	Passed
Remarks	

A.1.4. Safari

Test Item	Browsers
Sub-Item	Safari
Test S.O.	Windows 7, Windows XP
Software Version	5.1.7 (Windows 7, Windows XP)
Test Objective	Check the browsing experience with Safari browser behind the double translation of MAP-T.
Test Procedure	1.Open browser and input a content provider address. 2.After loading the page, load one of the last news pages that have not been accessed yet. 3.Check whether contents can be accessed and visualized.
Expected Result	Browser can open and visualize content correctly.
Actual Result	Passed
Remarks	

A.1.5. Lynx (text browser)

Test Item	Browsers
Sub-Item	Lynx (text browser)
Test S.O.	Linux
Software Version	2.8.8 (Linux)
Test Objective	Check the browsing experience with Lynx browser behind the double translation of MAP-T.
Test Procedure	1.Open browser, input a webmail address and perform the login. 2.After loading the page, verify the directories of the account and check for stored emails. 3.Check whether the webmail content is accessed and visualized.
Expected Result	Browser can open and visualize content correctly.
Actual Result	Passed
Remarks	

A.2. Web browsing

A.2.1. www.google.com

Test Item	Web browsing
Sub-Item	www.google.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether we can access the services of Google, including the search mechanism, behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Google search engine address. 2.After loading the page, perform a search not done yet. 3.Check whether the Google outputs websites address that match the search. 4.Access the Google Maps service. 5.After loading the page, switch to satellite visualization. 6.Perform a known street address search. 7.Check whether the satellite images of the street are exhibited.
Expected Result	The search engine recognizes requests from user. The Maps service exhibits the satellite images of the appointed localization.
Actual Result	Passed
Remarks	

A.2.2. www.msn.com

Test Item	Web browsing
Sub-Item	www.msn.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether the various contents provided by the MSN website can be accessed behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the MSN website address. 2.After loading the page, load one of the last news page that have not been accessed yet. 3.Check whether contents can be accessed and visualized.
Expected Result	MSN content can be opened and visualized correctly.
Actual Result	Passed
Remarks	

A.3. Web dynamic content

A.3.1. Flash Player

Test Item	Web dynamic content
Sub-Item	Flash Player
Test S.O.	Linux, Windows 7, Windows XP
Software Version	13.3.31.109 (Linux), 11.3.31.227 (Windows 7), 11.3.300 (Windows XP)
Test Objective	Check whether Flash content (videos, mainly) can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Adobe website address. 2.Download and install the version of Adobe Flash Player better suited to the system in use. 3.Refresh or reopen the browser and input the Youtube address. 3.After loading the page, load one of the last videos posted, not visualized yet. 4.Check whether the video is loaded and exhibited in the Flash version of Youtube player.
Expected Result	Video played in Flash can be visualized correctly.
Actual Result	Passed
Remarks	Example used: http://www.flashexample.com/

A.3.2. Silverlight

Test Item	Web dynamic content
Sub-Item	Silverlight
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Moonlight 3.99.0.3 (Linux), 5.1.10411 (Windows 7, Windows XP)
Test Objective	Check whether Silverlight content can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Microsoft website address. 2.Download and install the version of Microsoft Silverlight better suited to the system in use. 3.Refresh or reopen the browser, input the Google address, search for a 'Silverlight Example' and open the link of a result (examples can be found in the references at the end of this document). 4.After loading the page with and example, check whether the plugin is working properly.
Expected Result	Silverlight applet executes correctly.
Actual Result	Passed
Remarks	Linux must use Moonlight as alternative to Silverlight as it is not available for Linux. The Moonlight plugin for Linux has problems to refresh the images of applets. Images are only refreshed when the user comes back to the visualization of the tab running Moonlight. Example used: http://flashenabled.wordpress.com/2007/07/09/from-a-to-z-50-silverlight-applications/

A.3.3. Java applets

Test Item	Web dynamic content
Sub-Item	Java applets
Test S.O.	Linux, Windows 7, Windows XP
Software Version	1.6.0_24 open jdk (Linux), jdk 7.0.5 (Windows 7, Windows XP)
Test Objective	Check whether Java applications can be downloaded and executed behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Java website address. 2.Download and install the version of Java better better suited to the system in use. 3.Refresh or reopen the browser, input the Google address, search for a 'Java Example' and open the link of a result (examples can be found in the references at the end of this document). 4.After loading the page with and example, check whether the plugin is working properly.
Expected Result	Java applet executes correctly.
Actual Result	Passed
Remarks	Example used: http://profs.etsmtl.ca/mmcguffin/learn/java/

A.3.4. HTML5 websites

Test Item	Web dynamic content
Sub-Item	HTML5 websites
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether the contents of HTML5 websites can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Google address. 2.Search for a 'HTML5 Example' and open the link of a result (examples can be found in the references at the end of this document). 3.After loading one page with video content, load a video. 4.Check whether the video is visualized properly.
Expected Result	HTML5 website content can be downloaded and visualized without errors.
Actual Result	Passed
Remarks	Example used: http://101besthtml5sites.com/

A.4. Video stream websites

A.4.1. www.youtube.com

Test Item	Video stream websites
Sub-Item	www.youtube.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether videos and contents of the Youtube streaming video website can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Youtube address. 2.After loading the page, load one of the last videos posted. 3.Check whether the video is loaded and exhibited.
Expected Result	Youtube video can be downloaded and visualized correctly.
Actual Result	Passed
Remarks	

A.4.2. www.dailymotion.com

Test Item	Video stream websites
Sub-Item	www.dailymotion.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether videos and contents of the Dailymotion streaming video website can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Dailymotion address. 2.After loading the page, load one of the last videos posted. 3.Check whether the video is loaded and exhibited.
Expected Result	Dailymotion video can be downloaded and visualized correctly.
Actual Result	Passed
Remarks	

A.4.3. www.zappiens.br

Test Item	Video stream websites
Sub-Item	www.zappiens.br
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether videos and contents of the Zappiens streaming video website (brazilian digital content website) can be downloaded and visualized behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Zappiens address. 2.After loading the page, load one of the last videos posted. Alternate between the three formats of exhibition avaiable. 3.Check whether the video is loaded and exhibited.
Expected Result	Zappiens video can be downloaded and visualized correctly in the three avaiable formats.
Actual Result	Passed
Remarks	

A.5. Social networking websites

A.5.1. www.facebook.com

Test Item	Social networking websites
Sub-Item	www.facebook.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether a Facebook account can be accessed and Facebook applets (as chat) can be executed behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Facebook address. 2.Navigate between profiles. 3.Check whether contents are loaded and exhibited. 4.Open the chat applet and perform messages exchanges. 5.Check whether contacts are exhibited and messages can be exchanged.
Expected Result	Profiles can be accessed and visualized correctly. Applets executes without errors.
Actual Result	Passed
Remarks	

A.5.2. www.twitter.com

Test Item	Social networking websites
Sub-Item	www.twitter.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether a Twitter account can be accessed and Twitter applets can be executed behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Twitter address. 2.Navigate between profiles. 3.Check whether contents are loaded and exhibited.
Expected Result	Profiles are accessed and visualized correctly.
Actual Result	Passed
Remarks	

A.5.3. www.orkut.com

Test Item	Social networking websites
Sub-Item	www.orkut.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether a Orkut account can be accessed and Orkut applets (as chat) can be executed behind the double translation of MAP-T.
Test Procedure	1.Open browser and input the Orkut address. 2.Navigate between profiles. 3.Check whether contents are loaded and exhibited. 4.Open the chat applet and perform messages exchanges. 5.Check whether contacts are exhibited and messages can be exchanged.
Expected Result	Profiles are accessed and visualized correctly. Applets executes without errors.
Actual Result	Passed
Remarks	

A.6. Webmails

A.6.1. www.gmail.com

Test Item	Webmails
Sub-Item	www.gmail.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether the Gmail webmail can be accessed and Gmail applets can be executed behind the double . translation of MAP-T.
Test Procedure	1.Open browser, input Gmail address and perform the login. 2.After loading the page, verify the directories of the account and check for stored emails. 3.Check whether the webmail content is accessed and visualized.
Expected Result	Account can be accessed and emails can be visualized whithout errors.
Actual Result	Passed
Remarks	

A.6.2. www.hotmail.com

Test Item	Webmails
Sub-Item	www.hotmail.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether the Hotmail webmail can be accessed behind the double translation of MAP-T.
Test Procedure	1.Open browser, input Hotmail address and perform the login. 2.After loading the page, verify the directories of the account and check for stored emails. 3.Check whether the webmail content is accessed and visualized.
Expected Result	Account can be accessed and emails can be visualized whithout errors.
Actual Result	Passed
Remarks	

A.7. Real-time Internet text messaging (chat) website

A.7.1. Chat rooms of UOL content provider

Test Item	Real-time Internet text messaging (chat) website
Sub-Item	Chat rooms of UOL content provider
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether it is possible to access and use chat rooms behind the double translation of MAP-T.
Test Procedure	1.Open browser and input UOL address. 2.After loading the page, access the "BATE-PAPO" link, enter a chat room, fill information fields and submit the solicitation. 3.Send messages to the chat room. 4.Check whether messages are sent and exhibited.
Expected Result	The user can enter chat rooms, post messages and visualize others' messages.
Actual Result	Passed
Remarks	

A.8. Image hosting site

A.8.1. www.flickr.com

Test Item	Image hosting site
Sub-Item	www.flickr.com
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Chrome and Firefox (Linux, Windows 7, Windows XP)
Test Objective	Check whether it is possible to access, upload images and use the interface of the Flickr site behind the double translation of MAP-T.
Test Procedure	1.Open browser, input Flickr address and perform login. 2.After loading the page, upload an image. 3.Check whether image was uploaded.
Expected Result	The user can login the Flickr site and upload images.
Actual Result	Passed
Remarks	

A.9. Communication protocols

A.9.1. Skype

Test Item	Communication protocols
Sub-Item	Skype
Test S.O.	Linux, Windows 7, Windows XP
Software Version	4.0.0.7 (Linux), 5.10.0.116 (Windows 7, Windows XP)
Test Objective	Check whether it is possible to login a Skype account and to initiate and mantain a Skype session behind the double translation of MAP-T.
Test Procedure	1.Download and install Skype. 2.Open Skype and perform login. 3.Search for an online contact, start a chat and exchange messages. 4.Check the exchange of messages.
Expected Result	Skype can login, identify contacts online, establish a session with a contact and exchange messages.
Actual Result	Passed
Remarks	

A.9.2. Googletalk

Test Item	Communication protocols
Sub-Item	Googletalk
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Pidgin 2.10.3 (Linux), Pidgin 2.10.6 (Windows 7, Windows XP)
Test Objective	Check whether it is possible to login a Googletalk account and to initiate and mantain a Googletalk session behind the double translation of MAP-T.
Test Procedure	1.Download and install Pidgin. 2.Open Pidgin, configure an Googletalk account and perform login. 3.Search for an online contact, start a chat and exchange messages. 4.Check the exchange of messages.
Expected Result	Googletalk can login, identify contacts online, establish a session with a contact and exchange messages.
Actual Result	Passed
Remarks	

A.9.3. Jabber (XMPP)

Test Item	Communication protocols
Sub-Item	Jabber (XMPP)
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Pidgin 2.10.3 (Linux), Pidgin 2.10.6 (Windows 7, Windows XP)
Test Objective	Check whether it is possible to login a Jabber account and to initiate and mantain a Jabber session behind the double translation of MAP-T.
Test Procedure	1.Download and install Pidgin. 2.Open Pidgin, configure an XMPP account and perform login. 3.Search for an online contact, start a chat and exchange messages. 4.Check the exchange of messages.
Expected Result	Jabber can login, identify contacts online, establish a session with a contact and exchange messages.
Actual Result	Passed
Remarks	

A.9.4. MSN Messenger (Microsoft Notification Protocol)

Test Item	Communication protocols
Sub-Item	MSN Messenger (Microsoft Notification Protocol)
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Pidgin 2.10.3 (Linux), Pidgin 2.10.6 and Windows Live Messenger 16.4.3503 (Windows 7), Pidgin 2.10.6 and Windows Live Messenger 14.0.8117 (Windows XP)
Test Objective	Check whether it is possible to login to a MSN account, and to initiate and mantain a MSN session behind the double translation of MAP-T.
Test Procedure	1.Download and install Pidgin or Windows Live Messenger. 2.Open the software installed, configure an MSN (if Pidgin is the software used) and perform login. 3.Search for an online contact, start a chat and exchange messages. 4.Check the exchange of messages.
Expected Result	MSN can login, identify contacts online, establish a session with a contact and exchange messages.
Actual Result	Passed
Remarks	

A.9.5. IRC (Internet Relay Chat)

Test Item	Communication protocols
Sub-Item	IRC (Internet Relay Chat)
Test S.O.	Linux, Windows 7, Windows XP
Software Version	Pidgin 2.10.3 (Linux), Pidgin 2.10.6 (Windows 7, Windows XP)
Test Objective	Check whether it is possible to login an IRC account, to join a channel and to send and receive messages in a channel, behind the double translation of MAP-T.
Test Procedure	1.Download and install Pidgin. 2.Open Pidgin, configure an IRC account and perform login. 3.Join a channel and send messages. 4.Check whether the channel was joined and messages were exhibited.
Expected Result	IRC can login, join a channel, identify users inside the channel, send and visualize messages in the channel.
Actual Result	Passed
Remarks	

A.10. Torrent clients

A.10.1. Vuze

Test Item	Torrent clients
Sub-Item	Vuze
Test S.O.	Linux, Windows 7, Windows XP
Software Version	4.3.0.6 (Linux), 4.7.1.2 (Windows 7, Windows XP)
Test Objective	Check whether the Vuze BitTorrent client can connect to a tracker, identify peers and make a direct connection to them behind the double translation of MAP-T.
Test Procedure	1.Download and install the Vuze client. 2.Load a torrent file with many seeds and start the download. 3.Wait some minutes and check whether the client connects to a tracker, identifies at least one peer and the download starts.
Expected Result	Vuze client can connect to trackers, identify peers and start downloads.
Actual Result	Passed
Remarks	As it is not possible to receive incoming connections the software is not capable to perform the seeding of downloaded files.

A.10.2. uTorrent

Test Item	Torrent clients
Sub-Item	uTorrent
Test S.O.	Windows 7, Windows XP
Software Version	3.2 (Windows 7, Windows XP)
Test Objective	Check whether the uTorrent BitTorrent client can connect to a tracker, identify peers and make a direct connection to them behind the double translation of MAP-T.
Test Procedure	1.Download and install the uTorrent client. 2.Load a torrent file with many seeds and start the download. 3.Wait some minutes and check whether the client connects to a tracker, identifies at least one peer and the download starts.
Expected Result	uTorrent client can connect to trackers, identify peers and start downloads.
Actual Result	Passed
Remarks	As it is not possible to receive incoming connections the software is not capable to perform the seeding of downloaded files.

A.10.3. Ktorrent

Test Item	Torrent clients
Sub-Item	Ktorrent
Test S.O.	Linux
Software Version	4.8.2
Test Objective	Check whether the KTorrent BitTorrent client can connect to a tracker, identify peers and make a direct connection to them behind the double translation of MAP-T.
Test Procedure	1.Download and install the Ktorrent client. 2.Load a torrent file with many seeds and start the download. 3.Wait some minutes and check whether the client connects to a tracker, identifies at least one peer and the download starts.
Expected Result	Ktorrent client can connect to trackers, identify peers and start downloads.
Actual Result	Passed
Remarks	As it is not possible to receive incoming connections the software is not capable to perform the seeding of downloaded files.

A.10.4. Note aout bittorrent seeders

Bittorrent uses distributed queues, each seeder owns the queues for the files they have. The seeder informs the tracker that it has the file and the tracker informs the clients about this seeder. The client send messages to this announced seeder to try to download the file. With no incoming connection the bittorret fails here on the MAP-T. Despite being a possible seeder the machine doesn’t upload the file and without upload, it will be considered a leech and will be penalized on future downloading speeds. This is the same problem that happens on a network with IPv4 without port forwarding. For details about the influence on shared IPv4 address on torrent look I-D.draft-boucadair-pcp-bittorrent

A.11. Remote access and file transfer software

A.11.1. ssh

Test Item	Remote access and file transfer software
Sub-Item	ssh
Test S.O.	Linux, Windows 7, Windows XP
Software Version	openssh 5.9 (Linux), putty 0.62 (Windows 7, Windows XP)
Test Objective	Check whether it is possible to log into a remote machine, via ssh, behind the double translation of MAP-T.
Test Procedure	1.Install openssh-client (Linux) or putty (Windows) on a host inside MAP-T network. 2.Attempt a connection to the host from a machine outside the MAP-T network. 3.Check whether the remote access is established.
Expected Result	MAP-T client can make or accept remote access to hosts outside MAP-T network via ssh.
Actual Result	Passed
Remarks	

A.11.2. ftp

Test Item	Remote access and file transfer software
Sub-Item	ftp
Test S.O.	Linux, Windows 7, Windows XP
Software Version	standard command line FTP (Linux, Windows 7, Windows XP)
Test Objective	Check whether it is possible to log into a remote FTP server behind the double translation of MAP-T.
Test Procedure	1.Connect to a FTP server outside MAP-T network. 2.Attempt to create, delete and browser folders. 3.Attempt to list, send and receive files.
Expected Result	To complete all de described actions in the server.
Actual Result	Passed *
Remarks	It is possible to connect to the server with or without authentication. It is possible to create, delete and browser folders. It is not possible to list, send and receive files in active mode. Using passive mode in Linux is possible to list, send and receive files. Using passive mode on Windows 7 and XP didn’t solve the problem with list, send and receive files, looks like passive mode is not correctly implemented. Active mode doesn’t work on networks with NAT, so it is acceptable that active mode doesn’t work with MAP-T too.

A.11.3. Filezilla ftp

Test Item	Remote access and file transfer software
Sub-Item	Filezilla ftp
Test S.O.	Linux, Windows 7, Windows XP
Software Version	3.5.3 (Linux, Windows 7, Windows XP)
Test Objective	Check whether it is possible to log into a remote FTP server behind the double translation of MAP-T.
Test Procedure	1.Download and install FileZilla. 2.Connect to a FTP server outside MAP-T network. 3.Attempt to create, delete and browser folders. 4.Attempt to list, send and receive files.
Expected Result	To complete all de described actions in the server.
Actual Result	Passed
Remarks	Filezilla default configuration uses passive mode. When using active mode FileZilla can’t list, send and receive files.

A.11.4. wget

Test Item	Remote access and file transfer software
Sub-Item	wget
Test S.O.	Linux
Software Version	1.13.4
Test Objective	Check whether it is possible to download files with the wget tool and utilize its functionalities behind the double translation of MAP-T.
Test Procedure	1.Install the wget tool. 2.Perform the download of a file in the Internet. 3.Check whether the download is completed.
Expected Result	User can download files from the Internet with wget.
Actual Result	Passed
Remarks	

A.12. Antivirus updates

A.12.1. Avira

Test Item	Antivirus updates
Sub-Item	Avira
Test S.O.	Windows 7, Windows XP
Software Version	12.0.0.289 (Windows 7, Windows XP)
Test Objective	Check whether the update software of Avira Antivirus can connect to the update server and download files behind the double translation of MAP-T.
Test Procedure	1.Download and install Avira Antivirus. 2.Initiate the update of the software. 3.Check whether the main window indicates that the definitions of virus are updated.
Expected Result	Avira can update virus definitions from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.12.2. AVG

Test Item	Antivirus updates
Sub-Item	AVG
Test S.O.	Windows 7, Windows XP
Software Version	2012.0.2197 (Windows 7, Windows XP)
Test Objective	Check whether the update software of AVG Antivirus can connect to the update server and download files behind the double translation of MAP-T.
Test Procedure	1.Download and install AVG Antivirus. 2.Initiate the update of the software. 3.Check whether the main window indicates that the definitions of virus are updated.
Expected Result	AVG can update virus definitions from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.12.3. Avast

Test Item	Antivirus updates
Sub-Item	Avast
Test S.O.	Windows 7, Windows XP
Software Version	7.0.1456 (Windows 7, Windows XP)
Test Objective	Check whether the update software of Avast Antivirus can connect to the update server and download files behind the double translation of MAP-T.
Test Procedure	1.Download and install Avast Antivirus. 2.Initiate the update of the software. 3.Check whether the main window indicates that the definitions of virus are updated.
Expected Result	Avast can update virus definitions from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.13. Media player updates and video streaming

A.13.1. VLC

Test Item	Media player updates and video streaming
Sub-Item	VLC
Test S.O.	Linux, Windows 7, Windows XP
Software Version	2.0.3 Twoflower (Linux, Windows 7), 2.0.1 Twoflower (Windows XP)
Test Objective	Check whether VLC can connect to the update server and download files, and access media content providers behind the double translation of MAP-T.
Test Procedure	1.Download and install VLC. 2.Initiate the update of the software. 3.Check whether the update window indicates a connection to the server. 4.Open the menu "Internet" and access a channel. 5.Check whether the media content is loaded.
Expected Result	VLC can be updated and access content providers from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.13.2. Realplayer

Test Item	Media player updates and video streaming
Sub-Item	Realplayer
Test S.O.	Windows 7, Windows XP
Software Version	15.0.6.14 (Windows 7, Windows XP)
Test Objective	Check whether Realplayer can connect to the update server and download files, and acces media content providers behind the double translation of MAP-T.
Test Procedure	1.Download and install Realplayer. 2.Initiate the update of the software. 3.Check whether the update window indicates a connection to the server. 4.Open the "Online Guide". 5.Check whether a webpage is loaded.
Expected Result	Realplayer can be updated and connect to content providers from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.13.3. Windows Media Player

Test Item	Media player updates and video streaming
Sub-Item	Windows Media Player
Test S.O.	Windows 7, Windows XP
Software Version	12.0.7600 (Windows 7), 11.0.5721 (Windows XP)
Test Objective	Check whether the Windows Media Player can connect to the update server and download files, and access music content providers behind the double translation of MAP-T.
Test Procedure	1.Download and install Realplayer. 2.Initiate the update of the software. 3.Check whether the update window indicates a connection to the server. 4.Open the "Media Guide". 5.Check whether a webpage is loaded.
Expected Result	Windows Media Player can be updated and connect to content providers from inside a MAP-T network.
Actual Result	Passed
Remarks	

A.14. Network testing tools

A.14.1. ping

Test Item	Network testing tools
Sub-Item	ping
Test S.O.	Linux, Windows 7, Windows XP
Software Version	
Test Objective	Check whether the ping tool can send and answer ICMP packets to and from hosts outside the MAP-T network.
Test Procedure	1.Ping a known host outside the MAP-T network (www.google.com for instance). 2.Check whether there are answer to the pings. 3.Ping the MAP-T client tested from a known host outside the MAP-T network. 4.Check whether there are answer to the pings.
Expected Result	MAP-T client can send and answer pings to and from hosts outside the MAP-T network.
Actual Result Version	Failed to receive inbound pings, but replies to requested pings are received.
Remarks	The version of MAP-T 1.1 tested does not allow incoming connections to the MAP-T client.

A.14.2. traceroute

Test Item	Network testing tools
Sub-Item	traceroute
Test S.O.	Linux
Software Version	
Test Objective	Check whether the traceroute tool can identify each router in the path of a packet sent to a host outside the MAP-T network or sent from that host to the MAP-T client.
Test Procedure	1.Install the traceroute tool in both machines considered. 2.Open a terminal on the MAP-T client and execute a traceroute to a known host outside the MAP-T network (www.google.com for instance). 3.Check whether the route is traced until the known host and the list of intermediary routers is complete. 4.Open a terminal on a host outside the MAP-T network and execute a traceroute to the MAP-T client. 5.Check whether the route is traced until the MAP-T client and the list of intermediary routers is complete.
Expected Result	traceroute tool can map the complete route of a packet to and from the MAP-T client.
Actual Result	Failed
Remarks	After the first v4/v6 translation, the routers are not mapped by the traceroute.

A.14.3. tracert

Test Item	Network testing tools
Sub-Item	tracert
Test S.O.	Windows 7, Windows XP
Software Version	
Test Objective	Check whether the tracert tool can identify each router in the path of a packet sent to a host outside the MAP-T network.
Test Procedure	1.Open a terminal on the MAP-T client and execute a tracert to a known host outside the MAP-T network (www.google.com for instance). 2.Check whether the route is traced until the known host and the list of intermediary routers is complete. 3.Open a terminal on a host outside the MAP-T network and execute a tracert to the MAP-T client. 4.Check whether the route is traced until the MAP-T client and the list of intermediary routers is complete.
Expected Result	tracert tool can map the complete route of a packet to and from the MAP-T client.
Actual Result	Failed
Remarks	After the first v4/v6 translation, the routers are not mapped by the traceroute.

Authors' Addresses

Edwin Cordeiro
NIC.br
Sao Paulo,
Brazil

Phone: +55 11 5509 3537
Email: ecordeiro@nic.br

Rodrigo Carnier
NIC.br
Sao Paulo,
Brazil

Phone: +55 11 5509 3537
Email: rmatos@nic.br

Antonio Marcos Moreiras
NIC.br
Sao Paulo,
Brazil

Phone: +55 11 5509 3537
Email: moreiras@nic.br

Softwire WG
Internet-Draft
Intended status: Standards Track
Expires: December 5, 2015

Q. Wang
China Telecom
W. Meng
C. Wang
ZTE Corporation
M. Boucadair
France Telecom
June 3, 2015

RADIUS Extensions for IPv4-Embedded Multicast and Unicast IPv6 Prefixes
draft-hu-softwire-multicast-radius-ext-08

Abstract

This document specifies a new Remote Authentication Dial-In User Service (RADIUS) attribute to carry the Multicast-Prefixes-64 information, aiming to delivery the Multicast and Unicast IPv6 Prefixes to be used to build multicast and unicast IPv4-Embedded IPv6 addresses. this RADIUS attribute is defined based on the equivalent DHCPv6 OPTION_v6_PREFIX64 option.

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 December 5, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Convention and Terminology 4
3. Multicast-Prefixes-64 Configuration with RADIUS and DHCPv6 . . 5
4. RADIUS Attribute 8
4.1. Multicast-Prefixes-64 8
5. Table of Attributes 11
6. Security Considerations 12
7. IANA Considerations 13
8. Acknowledgments 14
9. Normative References 15
Authors' Addresses 16

1. Introduction

The solution specified in [I-D.ietf-softwire-dslite-multicast] relies on stateless functions to graft part of the IPv6 multicast distribution tree and IPv4 multicast distribution tree, also uses IPv4-in-IPv6 encapsulation scheme to deliver IPv4 multicast traffic over an IPv6 multicast-enabled network to IPv4 receivers.

To inform the mB4 element of the PREFIX64,a PREFIX64 option may be used. [I-D.ietf-softwire-multicast-prefix-option] defines a DHCPv6 PREFIX64 option to convey the IPv6 prefixes to be used for constructing IPv4-embedded IPv6 addresses.

In broadband environments, a customer profile may be managed by Authentication, Authorization, and Accounting (AAA) servers, together with AAA for users. The Remote Authentication Dial-In User Service (RADIUS) protocol [RFC2865] is usually used by AAA servers to communicate with network elements. Since the Multicast-Prefixes-64 information can be stored in AAA servers and the client configuration is mainly provided through DHCP running between the NAS and the requesting clients, a new RADIUS attribute is needed to send Multicast-Prefixes-64 information from the AAA server to the NAS.

This document defines a new RADIUS attribute to be used for carrying the Multicast-Prefixes-64, based on the equivalent DHCPv6 option already specified in [I-D.ietf-softwire-multicast-prefix-option].

This document makes use of the same terminology defined in [I-D.ietf-softwire-dslite-multicast].

This attribute can be in particular used in the context of DS-Lite Mulitcast, MAP-E Multicast and other IPv4-IPv6 Multicast techniques. However it is not limited to DS-Lite Multicast.

DS-Lite unicast RADIUS extentions are defined in [RFC6519] .

2. Convention and Terminology

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].

The terms DS-Lite multicast Basic Bridging BroadBand element (mB4) and the DS-Lite multicast Address Family Transition Router element (mAFTR) are defined in [I-D.ietf-softwire-dslite-multicast]

3. Multicast-Prefixes-64 Configuration with RADIUS and DHCPv6

Figure 1 illustrates in DS-Lite scenario how the RADIUS protocol and DHCPv6 work together to accomplish Multicast-Prefixes-64 configuration on the mB4 element for multicast service when an IP session is used to provide connectivity to the user.

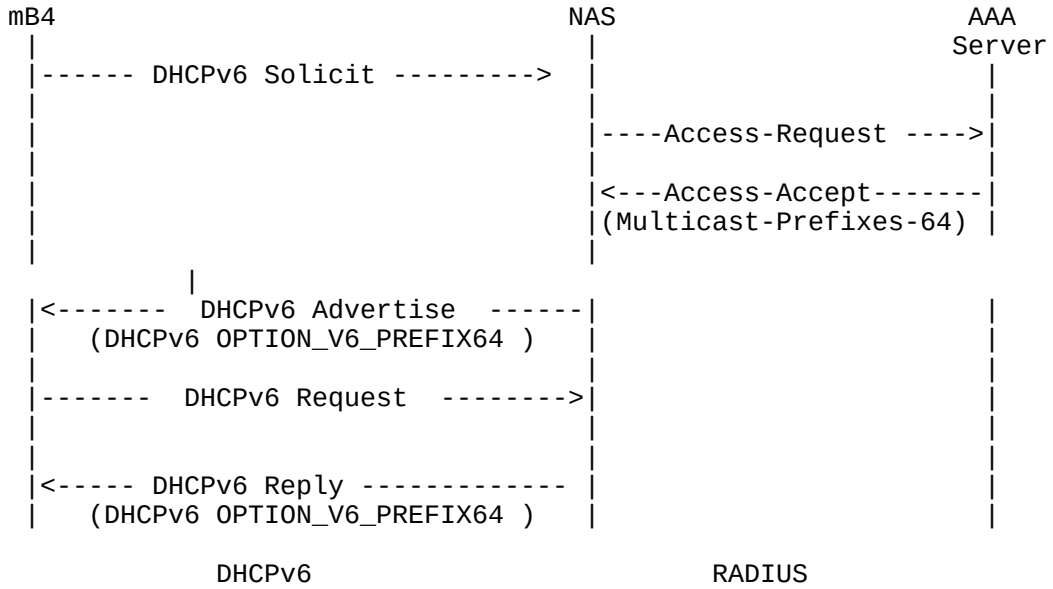


Figure 1: RADIUS and DHCPv6 Message Flow for an IP Session

The NAS operates as a client of RADIUS and as a DHCP Server/Relay for mB4. When the mB4 sends a DHCPv6 Solicit message to NAS(DHCP Server/Relay). The NAS sends a RADIUS Access-Request message to the RADIUS server, requesting authentication. Once the RADIUS server receives the request, it validates the sending client, and if the request is approved, the AAA server replies with an Access-Accept message including a list of attribute-value pairs that describe the parameters to be used for this session. This list MAY contain the Multicast-Prefixes-64 attribute (asm-length,ASM_PREFIX64,ssm-length, SSM_PREFIX64,unicast-length,U_PREFIX64). Then, when the NAS receives the DHCPv6 Request message containing the OPTION_V6_PREFIX64 option in its Option Request option,the NAS SHALL use the prefixes returned in the RADIUS Multicast-Prefixes-64 attribute to populate the DHCPv6 OPTION_V6_PREFIX64 option in the DHCPv6 reply message.

NAS MAY be configured to return the configured Multicast-Prefixes-64 by the AAA Server to any requesting client without relaying each received request to the AAA Server.

Figure 2 describes another scenario, which accomplish DS-Lite Multicast-Prefixes-64 configuration on the mB4 element for multicast service when a PPP session is used to provide connectivity to the user. Once the NAS obtains the Multicast-Prefixes-64 attribute from the AAA server through the RADIUS protocol, the NAS MUST store the received Multicast-Prefixes-64 locally. When a user is online and sends a DHCPv6 Request message containing the OPTION_V6_PREFIX64 option in its Option Request option, the NAS retrieves the previously stored Multicast-Prefixes-64 and uses it as OPTION_V6_PREFIX64 option in DHCPv6 Reply message.

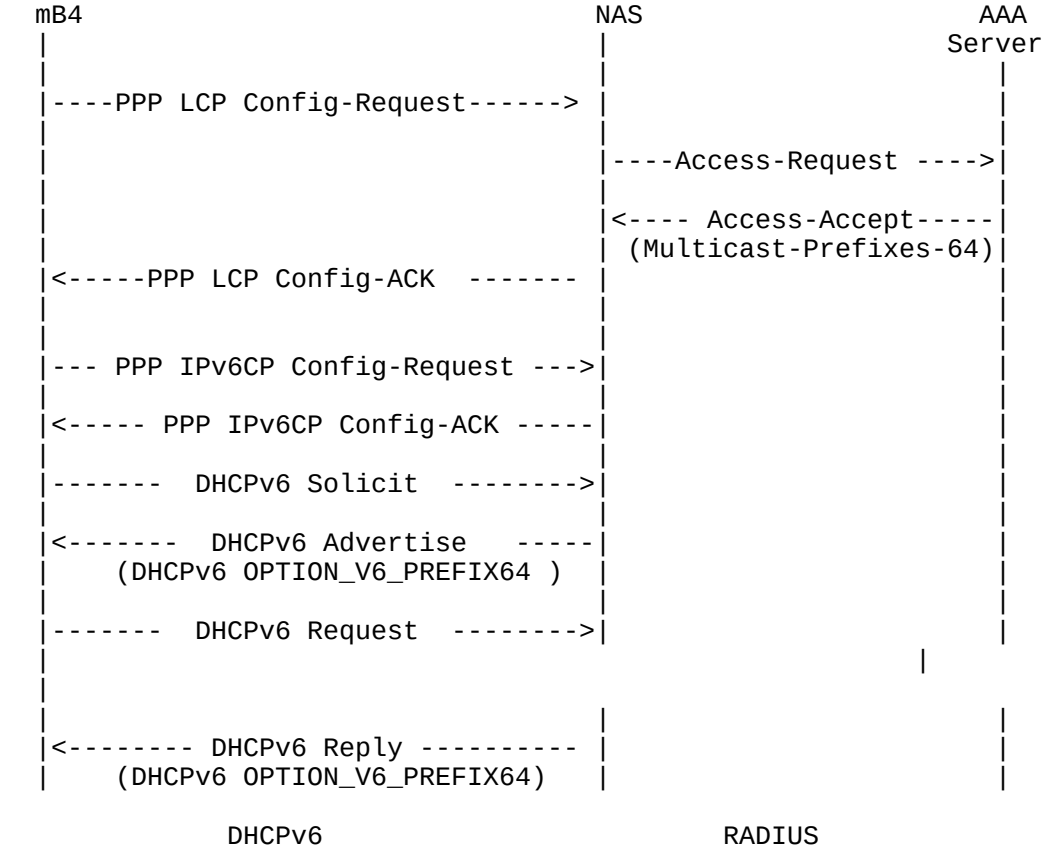


Figure 2: RADIUS and DHCPv6 Message Flow for a PPP Session

According to [RFC3315], after receiving the Multicast-Prefixes-64 attribute in the initial Access-Accept packet, the NAS MUST store the received V6_PREFIX64 locally. When the mB4 sends a DHCPv6 Renew message to request an extension of the lifetimes for the assigned address or prefix, the NAS does not have to initiate a new Access-

Request packet towards the AAA server to request the Multicast-Prefixes-64. The NAS retrieves the previously stored Multicast-Prefixes-64 and uses it in its reply.

Also, if the DHCPv6 server to which the DHCPv6 Renew message was sent at time T1 has not responded, the DHCPv6 client initiates a Rebind/Reply message exchange with any available server. In this scenario, the NAS receiving the DHCPv6 Rebind message MUST initiate a new Access-Request message towards the AAA server. The NAS MAY include the Multicast-Prefixes-64 attribute in its Access-Request message.

4. RADIUS Attribute

This section specifies the format of the new RADIUS attribute.

4.1. Multicast-Prefixes-64

The Multicast-Prefixes-64 attribute conveys the IPv6 prefixes to be used in [I-D.ietf-softwire-dslite-multicast] to synthesize IPv4-embedded IPv6 addresses. The NAS SHALL use the IPv6 prefixes returned in the RADIUS Multicast-Prefixes-64 attribute to populate the DHCPv6 PREFIX64 Option [I-D.ietf-softwire-multicast-prefix-option] .

This attribute MAY be used in Access-Request packets as a hint to the RADIUS server, for example, if the NAS is pre-configured with Multicast-Prefixes-64, these prefixes MAY be inserted in the attribute. The RADIUS server MAY ignore the hint sent by the NAS, and it MAY assign a different Multicast-Prefixes-64 attribute.

If the NAS includes the Multicast-Prefixes-64 attribute, but the AAA server does not recognize this attribute, this attribute MUST be ignored by the AAA server.

NAS MAY be configured with both ASM_PREFIX64 and SSM_PREFIX64 or only one of them. Concretely, AAA server MAY return ASM_PREFIX64 or SSM_PREFIX64 based on the user profile and service policies. AAA MAY return both ASM_PREFIX64 and SSM_PREFIX64. When SSM_PREFIX64 is returned by the AAA server, U_PREFIX64 MUST also be returned by the AAA server.

If the NAS does not receive the Multicast-Prefixes-64 attribute in the Access-Accept message, it MAY fall back to a pre-configured default Multicast-Prefixes-64, if any. If the NAS does not have any pre-configured, the delivery of multicast traffic is not supported.

If the NAS is pre-provisioned with a default Multicast-Prefixes-64 and the Multicast-Prefixes-64 received in the Access-Accept message are different from the configured default, then the Multicast-Prefixes-64 attribute received in the Access-Accept message MUST be used for the session.

A summary of the Multicast-Prefixes-64 RADIUS attribute format is shown Figure 3. The fields are transmitted from left to right.

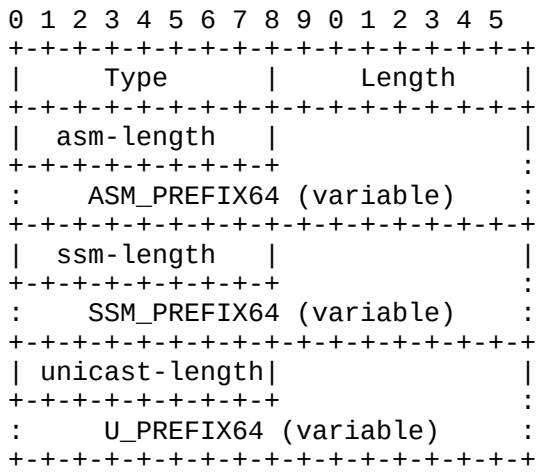


Figure 3: RADIUS attribute format for Multicast-Prefixes-64

Type:

145 for Multicast-Prefixes-64

Length:

This field indicates the total length in octets of this attribute including the Type and Length fields, and the length in octets of all PREFIX fields.

asm-length:

the prefix-length for the ASM IPv4-embedded prefix, as an 8-bit unsigned integer (0 to 128). This field represents the number of valid leading bits in the prefix.

ASM_PREFIX64:

this field identifies the IPv6 multicast prefix to be used to synthesize the IPv4-embedded IPv6 addresses of the multicast groups in the ASM mode. It is a variable size field with the length of the field defined by the asm-length field and is rounded up to the nearest octet boundary. In such case any additional padding bits must be zeroed. The conveyed multicast IPv6 prefix MUST belong to the ASM range. This prefix is likely to be a /96.

ssm-length:

the prefix-length for the SSM IPv4-embedded prefix, as an 8-bit unsigned integer (0 to 128). This field represents the number of valid leading bits in the prefix.

SSM_PREFIX64:

this field identifies the IPv6 multicast prefix to be used to synthesize the IPv4-embedded IPv6 addresses of the multicast groups in the SSM mode. It is a variable size field with the length of the field defined by the ssm-length field and is rounded up to the nearest octet boundary. In such case any additional padding bits must be zeroed. The conveyed multicast IPv6 prefix MUST belong to the SSM range. This prefix is likely to be a /96.

unicast-length:

the prefix-length for the IPv6 unicast prefix to be used to synthesize the IPv4-embedded IPv6 addresses of the multicast sources, as an 8-bit unsigned integer (0 to 128). This field represents the number of valid leading bits in the prefix.

U_PREFIX64:

this field identifies the IPv6 unicast prefix to be used in SSM mode for constructing the IPv4-embedded IPv6 addresses representing the IPv4 multicast sources in the IPv6 domain. U_PREFIX64 may also be used to extract the IPv4 address from the received multicast data flows. It is a variable size field with the length of the field defined by the unicast-length field and is rounded up to the nearest octet boundary. In such case any additional padding bits must be zeroed. The address mapping MUST follow the guidelines documented in [RFC6052].

5. Table of Attributes

The following tables provide a guide to which attributes may be found in which kinds of packets, and in what quantity.

The following table defines the meaning of the above table entries.

Access-Request	Access-Accept	Access-Reject	Challenge	Accounting-Request	#	Attribute
0-1	0-1	0	0	0-1	145	Multicast-Prefixes-64

CoA-Request	CoA-ACK	CoA-NACK	#	Attribute
0-1	0	0	145	Multicast-Prefixes-64

0 This attribute MUST NOT be present in the packet.
0+ Zero or more instances of this attribute MAY be present in the packet.
0-1 Zero or one instances of this attribute MAY be present in the packet.
1 Exactly one instances of this attribute MAY be present in the packet.

6. Security Considerations

This document has no additional security considerations beyond those already identified in [RFC2865] for the RADIUS protocol and in [RFC5176] for CoA messages.

The security considerations documented in [RFC3315] and [RFC6052] are to be considered.

7. IANA Considerations

Per this document, IANA has allocated a new RADIUS attribute type from the IANA registry "Radius Attribute Types" located at http://www.iana.org/assignments/radius-types.

Multicast-Prefixes-64 - 145

8. Acknowledgments

The authors would like to thank Ian Farrer, Chongfen Xie, Qi Sun, Linhui Sun and Hao Wang for their contributions to this work.

9. Normative References

[I-D.ietf-softwire-dslite-multicast]
Qin, J., Boucadair, M., Jacquenet, C., Lee, Y., and Q. Wang, "Delivery of IPv4 Multicast Services to IPv4 Clients over an IPv6 Multicast Network", draft-ietf-softwire-dslite-multicast-09 (work in progress), March 2015.

[I-D.ietf-softwire-multicast-prefix-option]
Boucadair, M., Qin, J., Tsou, T., and X. Deng, "DHCPv6 Option for IPv4-Embedded Multicast and Unicast IPv6 Prefixes", draft-ietf-softwire-multicast-prefix-option-08 (work in progress), March 2015.

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

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC5176] Chiba, M., Dommety, G., Eklund, M., Mitton, D., and B. Aboba, "Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)", RFC 5176, January 2008.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[RFC6519] Maglione, R. and A. Durand, "RADIUS Extensions for Dual-Stack Lite", RFC 6519, February 2012.

Authors' Addresses

Qian Wang
China Telecom
No.118, Xizhimennei
Beijing 100035
China

Email: wangqian@ctbri.com.cn

Wei Meng
ZTE Corporation
No.50 Software Avenue, Yuhuatai District
Nanjing
China

Email: meng.wei2@zte.com.cn,vally.meng@gmail.com

Cui Wang
ZTE Corporation
No.50 Software Avenue, Yuhuatai District
Nanjing
China

Email: wang.cui1@zte.com.cn

Mohamed Boucadair
France Telecom
Rennes, 35000
France

Email: mohamed.boucadair@orange.com

Internet Engineering Task Force
Internet-Draft
Intended status: Experimental
Expires: June 11, 2015

R. Despres
RD-IPtech
S. Jiang, Ed.
Huawei Technologies Co., Ltd
R. Penno
Cisco Systems, Inc.
Y. Lee
Comcast
G. Chen
China Mobile
M. Chen
Freebit Co, Ltd.
December 8, 2014

IPv4 Residual Deployment via IPv6 - a Stateless Solution (4rd) draft-ietf-softwire-4rd-10

Abstract

For service providers to progressively deploy IPv6-only network domains while still offering IPv4 service to customers, this document specifies a stateless solution. Its distinctive property is that TCP/UDP IPv4 packets are valid TCP/UDP IPv6 packets during domain traversal, and that IPv4 fragmentation rules are fully preserved end-to-end. Each customer can be assigned one public IPv4 address, or several, or a shared address with a restricted port set.

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 June 11, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Terminology 4
3. The 4rd Model 5
4. Protocol Specifications 8
4.1. NAT44 on CE 8
4.2. Mapping rules and other Domain parameters 8
4.3. Reversible Packet Translations at Domain entries and exits 9
4.4. Address Mapping from CE IPv6 Prefixes to 4rd IPv4 prefixes 14
4.5. Address Mapping from 4rd IPv4 addresses to 4rd IPv6 Addresses 16
4.6. Fragmentation Processing 21
4.6.1. Fragmentation at Domain Entry 21
4.6.2. Ports of Fragments addressed to Shared-Address CEs . 21
4.6.3. Packet Identifications from Shared-Address CEs . . . 22
4.7. TOS and Traffic-Class Processing 23
4.8. Tunnel-Generated ICMPv6 Error Messages 23
4.9. Provisioning 4rd Parameters to CEs 24
5. Security Considerations 27
6. IANA Considerations 28
7. Relationship with Previous Works 28
8. Acknowledgements 29
9. References 30
9.1. Normative References 30
9.2. Informative References 31
Appendix A. Textual representation of Mapping rules 32
Appendix B. Configuring multiple Mapping Rules 33
Appendix C. ADDING SHARED IPv4 ADDRESSES TO AN IPv6 NETWORK . . 35
C.1. With CEs within CPEs 35
C.2. With some CEs behind Third-party Router CPEs 36

Appendix D. REPLACING DUAL-STACK ROUTING BY IPv6-ONLY ROUTING . 37
Appendix E. ADDING IPv6 AND 4rd SERVICE TO A NET-10 NETWORK . . 38
Authors' Addresses 39

1. Introduction

For service providers to progressively deploy IPv6-only network domains while still offering IPv4 service to customers, the need for a stateless solution, i.e. one where no per-customer state is needed in IPv4-IPv6 gateway nodes of the provider, is expressed in [I-D.ietf-softwire-stateless-4v6-motivation]. This document specifies such a solution, named "4rd" for IPv4 Residual Deployment.

Using the solution, IPv4 packets are transparently tunneled across IPv6 networks (reverse of 6rd [RFC5969] in which IPv6 packets are statelessly tunneled across IPv4 networks).

While IPv6 headers are too long to be mapped into IPv4 headers (why 6rd requires encapsulation of full IPv6 packets in IPv4 packets), IPv4 headers can be reversibly translated into IPv6 headers in such a way that, during IPv6 domain traversal, UDP packets having checksums and TCP packets are valid IPv6 packets. IPv6-only middle boxes that perform deep-packet- inspection can operate on them, in particular for port inspection and web caches.

In order to deal with the IPv4-address shortage, customers can be assigned shared public IPv4 addresses, with statically assigned restricted port sets. As such, it is a particular application of the A+P approach of [RFC6346].

Deploying 4rd in the networks that have enough public IPv4 address, customer sites can also be assigned full public IPv4 addresses. 4rd also supports the scenarios that a set of public IPv4 addresses are assigned to customer sites.

The design of 4rd builds on a number of previous proposals made for IPv4-via-IPv6 transition technologies listed in Section 8.

In some use cases, IPv4-only applications of 4rd-capable customer nodes can also work with stateful NAT64s of [RFC6146], provided these are upgraded to support 4rd tunnels in addition their IP/ICMP translation of [RFC6145]. The advantage is then a more complete IPv4 transparency than with double translation.

How the 4rd model fits in the Internet architecture is summarized in Section 3. The protocol specification is detailed in Section 4. Section 5 and Section 6 respectively deal with security and IANA considerations. Previous proposals that influenced this

specification are listed in Section 8. A few typical 4rd use cases are presented in Appendices.

2. Terminology

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].

ISP: Internet-Service Provider. In this document, the service it offers can be DSL, fiber-optics, cable, or mobile. The ISP can also be a private-network operator.

4rd (IPv4 Residual Deployment): An extension of the IPv4 service where public IPv4 addresses can be statically shared among several customer sites, each one being assigned an exclusive port set. This service is supported across IPv6-routing domains.

4rd domain (or Domain): An ISP-operated IPv6 network across which 4rd is supported according to the present specification.

Tunnel packet: An IPv6 packet that transparently conveys an IPv4 packet across a 4rd domain. Its header has enough information to reconstitute the IPv4 header at Domain exit. Its payload is the original IPv4 payload.

CE (Customer Edge): A customer-side tunnel endpoint. It can be in a node that is a host, a router, or both.

BR (Border Relay): An ISP-side tunnel-endpoint. Because its operation is stateless (neither per CE nor per session state) it can be replicated in as many nodes as needed for scalability.

4rd IPv6 address: IPv6 address used as destination of a Tunnel packet sent to a CE or a BR.

NAT64+: An ISP NAT64 of [RFC6146] that is upgraded to support 4rd tunneling when IPv6 addresses it deals with are 4rd IPv6 addresses.

4rd IPv4 address: A public IPv4 address or, in case of a shared public IPv4 address, a public transport address (public IPv4 address plus port number).

PSID (Port-Set Identifier): A flexible-length field that algorithmically identifies a port set.

4rd IPv4 prefix: A flexible-length prefix that may be a a public IPv4 prefix, a public IPv4 address, or a public IPv4 address followed by a PSID.

Mapping rule: A set of parameters that are used by BRs and CEs to derive 4rd IPv6 addresses from 4rd IPv4 addresses. Mapping rules are also used by each CE to derive a 4rd IPv4 prefix from an IPv6 prefix that it has been delegated.

EA bits (Embedded Address bits): Bits that are the same in a 4rd IPv4 address and in the 4rd IPv6 address derived from it.

BR mapping rule: The mapping rule applicable to off-domain IPv4 addresses (addresses reachable via BRs). It can also apply to some or all of CE-assigned IPv4 addresses.

CE mapping rule: A mapping rule that is applicable only to CE-assigned IPv4 addresses (shared or not).

NAT64+ mapping rule: Mapping rule applicable to IPv4 addresses reachable via a NAT64+.

CNP (Checksum Neutrality preserver): A field of 4rd IPv6 addresses that ensures that TCP-like checksums do not change when IPv4 addresses are replaced by 4rd IPv6 addresses.

4rd Tag: A 16-bit tag whose value permits, in 4rd CEs, BRs, and NAT64+s, to distinguish 4rd IPv6 addresses from other IPv6 addresses.

3. The 4rd Model

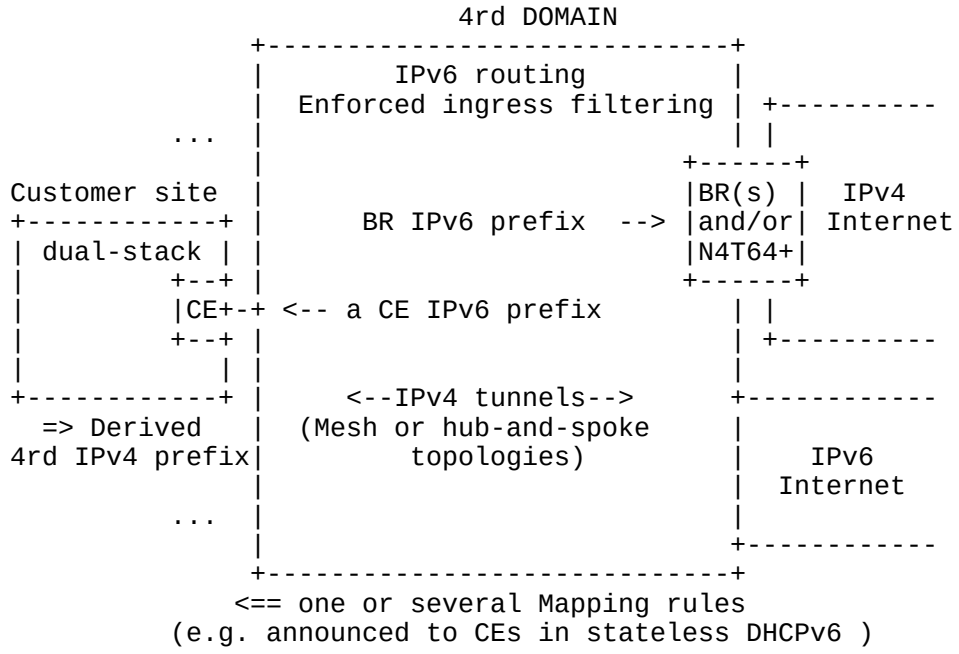


Figure 1

How the 4rd model fits in the Internet architecture is represented in Figure 1.

A 4rd domain is an IPv6 network that includes one or several 4rd BRs or NAT64+s at its border with the public IPv4 Internet, and can advertise its IPv4-IPv6 Mapping rule(s) to CEs according to Section 4.9.

BRs of a 4rd Domain are all identical as far as 4rd is concerned. In a 4rd CE, the IPv4 packets will be transformed (detailed in Section 4.3) into IPv6 packets that have the same anycast IPv6 prefix, which is the 80-bit BR prefix, in their destination addresses. They are then routed to any of the BRs. The 80-bit BR IPv6 prefix is an arbitrarily chosen /64 prefix from the IPv6 address space of the network operator and appended 0x0300 (16-bit 4rd Tag, see R-9 in Section 4.5).

Using the Mapping rule that applies, each CE derives its 4rd IPv4 prefix from its delegated IPv6 prefix, or one of them if it has several, details in Section 4.4. If the obtained IPv4 prefix has more than 32 bits, the assigned IPv4 address is shared among several CEs. Bits beyond the first 32 specify a set of ports whose use is reserved for the CE.

IPv4 traffic is automatically tunnelled across the Domain, either in mesh topology or in Hub&spoke topology [RFC4925]. By default, IPv4 traffic between two CEs follows a direct IPv6 route between them (mesh topology). If the ISP configures the Hub&spoke option, each IPv4 packet from a CE to another is routed via a BR.

During Domain traversal, each tunnelled TCP/UDP IPv4 packet looks like a valid TCP/UDP IPv6 packet. Thus, TCP/UDP access-control lists that apply to IPv6, and possibly some other functions using deep packet inspections, also apply to IPv4.

For IPv4 anti-spoofing protection, as is in CEs and BRs, to remain effective when combined with 4rd tunneling, ingress filtering has to be effective in IPv6 (R-12 and Section 5).

If an ISP wishes to support dynamic IPv4 address sharing, in addition or in place of 4rd stateless address sharing, it can do it by means of a stateful NAT64. By upgrading this NAT to add 4rd-tunnels support, which makes it a NAT64+, CEs that are assigned no static IPv4 space can benefit from complete IPv4 transparency between CE and NAT64. (Without this NAT64 upgrade, IPv4 traffic is translated to IPv6 and back to IPv4, which looses the DF=MF=1 combination of IPv4, that which is recommended for host fragmentation in Section 8 of [RFC4821].)

IPv4 packets are kept unchanged by Domain traversal except that:

- The IPv4 Time to live (TTL), unless it is 1 or 255 at Domain entry, decreases during Domain traversal by the number of traversed routers. This is acceptable because it is undetectable end to end, and because TTL values that can be used with some protocols to test adjacency of communicating routers are preserved ([RFC4271], [RFC5082]). Effect on the traceroute utility, which uses TTL expiry to discover routers of end-to-end paths, is noted in Section 4.3.

- IPv4 packets whose lengths are =< 68 octets always have their "Don’t fragment flags" DF=1 at Domain exit even if they had DF=0 at Domain entry. This is acceptable because these packets are too short to be fragmented [RFC0791] so that their DF bits have no meaning. Besides, both [RFC1191] and [RFC4821] recommend that sources always set DF=1.

- Unless the Tunnel-traffic-class option applies to a Domain (Section 4.2), IPv4 packets may have their TOS fields modified after Domain traversal (Section 4.7).

4. Protocol Specifications

This section describes detailed 4rd protocol specifications. They are mainly organized by functions. As a brief summary, a 4rd CE MUST follow R-1, R-2, R-3, R-4, R-6, R-7, R-8, R-9, R-10, R-11, R-12, R-13, R-14, R-16, R-17, R-18, R-19, R-20, R-21, R-22, R-23, R-24, R-25, R-26 and R-27; while a 4rd BR MUST follow R-2, R-3, R-4, R-5, R-6, R-9, R-12, R-13, R-14, R-15, R-19, R-20, R-21, R-22 and R-24.

4.1. NAT44 on CE

R-1: A CE node that is assigned a shared public IPv4 address MUST include a NAT44 [RFC3022]. This NAT44 MUST only use external ports that are in the CE assigned port set.

NOTE: This specification only concerns IPv4 communication between IPv4-capable endpoints. For communication between IPv4-only endpoints and IPv6 only remote endpoints, the BIH specification of [RFC6535] can be used. It can coexist in a node with the CE function, including if the IPv4-only function is a NAT44 [RFC3022].

4.2. Mapping rules and other Domain parameters

R-2: CEs and BRs MUST be configured with the following Domain parameters:

A. One or several Mapping rules, each one comprising:

1. Rule IPv4 prefix

2. EA-bits length

3. Rule IPv6 prefix

4. WKPs authorized (OPTIONAL)

B. Domain PMTU

C. Hub&spoke topology (Yes or No)

D. Tunnel traffic class (OPTIONAL)

"Rule IPv4 prefix" is used to find, by a longest match, which Mapping rule applies to a 4rd IPv4 address (Section 4.5). A Mapping rule whose Rule IPv4 prefix is longer than /0 is a CE mapping rule. BR and NAT64+ mapping rules, which must apply to all off-domain IPv4 addresses, have /0 as their Rule IPv4 prefixes.

"EA-bits length" is the number of bits that are common to 4rd IPv4 addresses and 4rd IPv6 addresses derived from them. In a CE mapping rule, it is also the number of bits that are common to a CE delegated IPv6 prefix and the 4rd IPv4 prefix derived from it. BR and NAT64+ mapping rules have EA-bits lengths equal to 32.

"Rule IPv6 prefix" is the prefix that is substituted to the Rule IPv4 prefix when a 4rd IPv6 address is derived from a 4rd IPv4 address (Section 4.5). In a BR mapping rule or a NAT64+ mapping rule, it MUST be a /80 prefix whose 64˜79 bits are the 4rd Tag.

"WKPs authorized" may be set for mapping rules that assign shared IPv4 addresses to CEs. (These rules are those whose length of the Rule IPv4 prefix plus the EA-bits length exceeds 32.) If set, well-known ports may be assigned to some CEs having particular IPv6 prefixes. If not set, fairness is privileged: all IPv6 prefixes concerned with the Mapping rule have ports sets having identical values (no port set includes any of the well known ports).

"Domain PMTU" is the IPv6 path MTU that the ISP can guarantee for all its IPv6 paths between CEs and between BRs and CEs. It MUST be at least 1280 [RFC2460].

"Hub&spoke topology", if set to Yes, requires CEs to tunnel all IPv4 packets via BRs. If set to No, CE-to-CE packets take the same routes as native IPv6 packets between the same CEs (mesh topology).

"Tunnel traffic class", if provided, is the IPv6 traffic class that BRs and CEs MUST set in Tunnel packets. In this case, evolutions of the IPv6 traffic class that may occur during Domain traversal are not reflected in TOS fields of IPv4 packets at Domain exit (Section 4.7).

4.3. Reversible Packet Translations at Domain entries and exits

R-3: Domain-entry nodes that receive IPv4 packets with IPv4 options MUST discard these packets, and return ICMPv4 error messages to signal IPv4-option incompatibility (Type = 12, Code = 0, Pointer = 20) [RFC0792]. This limitation is acceptable because there are a lot firewalls in current IPv4 Internet also filter IPv4 packets with IPv4 options.

R-4: Domain-entry nodes that receive IPv4 packets without IPv4 options MUST convert them to Tunnel packets, with or without IPv6 fragment headers depending on what is needed to ensure IPv4 transparency (Figure 2). Domain-exit nodes MUST convert them back to IPv4 packets.

An IPv6 fragmentation header MUST be included at tunnel entry (Figure 2) if, and only if, one or several of the following conditions hold:

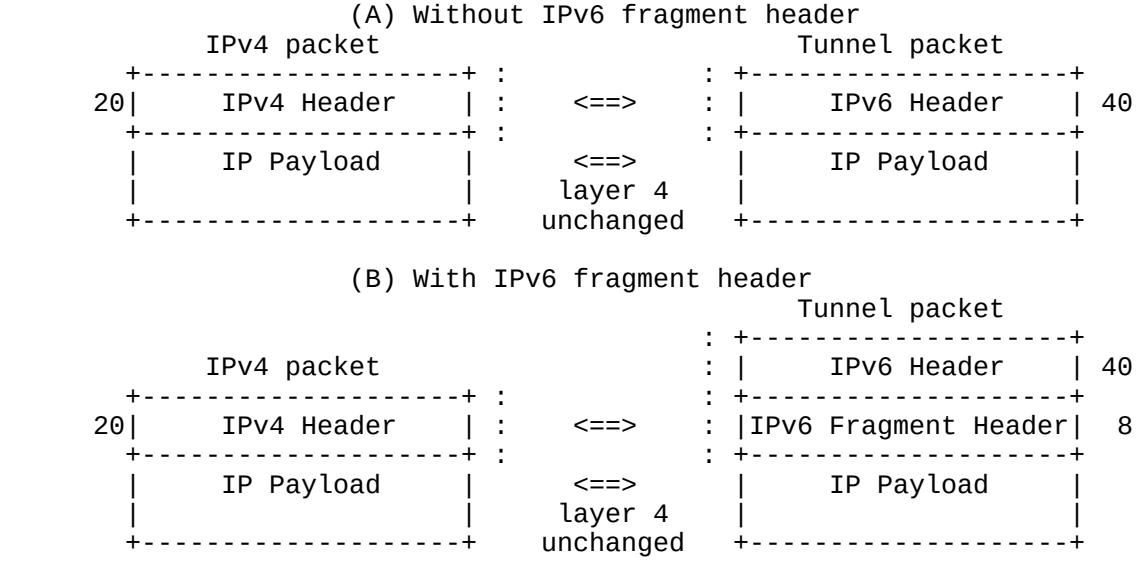
* The Tunnel_traffic_class option applies to the Domain.

* TTL = 1 OR TTL = 255.

* The IPv4 packet is already fragmented, or may be fragmented later on, i.e. if MF=1 OR Offset>0 OR (Total length > 68 AND DF=0).

In order to optimize cases where fragmentation headers are unnecessary, the NAT44 of a CE that has one SHOULD send packets with TTL = 254.

R-5: In Domains whose chosen topology is Hub&spoke, BRs that receive 4rd IPv6 packets whose embedded destination IPv4 addresses match a CE mapping rule MUST do the equivalent of reversibly translating their headers to IPv4 and then reversibly translate them back to IPv6 as though packets would be entering the Domain.



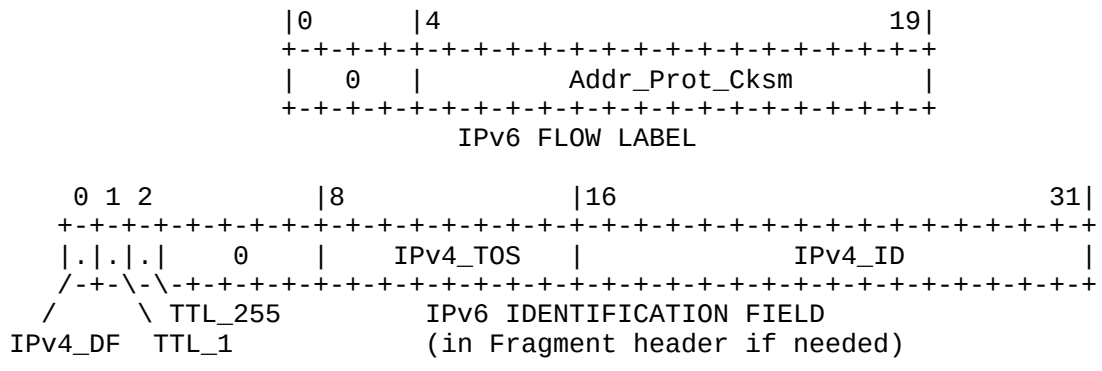
Reversible Packet Translation

Figure 2

R-6: Values to be set in IPv6-header fields at Domain entry are detailed in Table 1 (no-fragment-header case) and Table 2

(fragment-header case). Those to be set in IPv4 header fields at Domain exit are detailed in Table 3 (no-fragment-header case) and Table 4 (fragment-header case).

To convey IPv4-header informations that have no equivalent in IPv6, some ad-hoc fields are placed in IPv6 flow labels and in Identification fields of IPv6 fragment headers, as detailed in Figure 3.



4rd Identification fields of IPv6 Fragment headers

Figure 3

IPv6 FIELD	VALUE (fields from IPv4 header)
Version	6
Traffic class	TOS
Addr_Prot_Cksm	Sum of Addresses and Protocol (Note 1)
Payload length	Total length - 20
Next header	Protocol
Hop limit	Time to live
Source address	See Section 4.5
Destination address	See Section 4.5

IPv4-to-IPv6 Reversible Header Translation (without Fragment header)

Table 1

IPv6 FIELD	VALUE (fields from IPv4 header)
Version	6
Traffic class	TOS OR Tunnel_traffic_class (Section 4.7)
Addr_Prot_Cksm	Sum of Addresses and Protocol (Note 1)
Payload length	Total length - 12
Next header	44 (Fragment header)
Hop limit	IF Time to live = 1 or 255 THEN 254 ELSE Time to live (Note 2)
Source address	See Section 4.5
Dest. address	See Section 4.5
2nd next header	Protocol
Fragment offset	IPv4 Fragment offset
M	More-fragments flag (MF)
IPv4_DF	Don’t-fragment flag (DF)
TTL_1	IF Time to live = 1 THEN 1 ELSE 0 (Note 2)
TTL_255	IF Time to live = 255 THEN 1 ELSE 0 (Note 2)
IPv4_TOS	Type of service (TOS)
IPv4_ID	Identification

IPv4-to-IPv6 Reversible Header Translation (with Fragment header)

Table 2

IPv4 FIELD	VALUE (fields from IPv6 header)
Version	4
Header length	5
TOS	Traffic class
Total Length	Payload length + 20
Identification	0
DF	1
MF	0
Fragment offset	0
Time to live	Hop count
Protocol	Next header
Header checksum	Computed as per [RFC0791] (Note 3)
Source address	Bits 80-111 of source address
Dest. address	Bits 80-111 of source address

IPv6-to-IPv4 Reversible Header Translation (without Fragment header)

Table 3

IPv4 FIELD	VALUE (fields from IPv6 headers)
Version	4
Header length	5
TOS	Traffic class OR IPv4_TOS (Section 4.7)
Total Length	Payload length + 12
Identification	IPv4_ID
DF	IPv4_DF
MF	M
Fragment offset	Fragment offset
Time to live (Note 2)	IF TTL_255 = 1 THEN 255TTL_1 = 1 THEN 1 ELSEIF TTL_1 = 1 THEN 1 ELSE Hop count
Protocol	2nd Next header
Header checksum	Computed as per [RFC0791] (Note 3)
Source address	Bits 80-111 of source address
Destination address	Bits 80-111 of destination address

IPv6 to IPv4 Reversible Header Translation (with Fragment header)

Table 4

NOTE 1: The need to save in the IPv6 header a checksum of both IPv4 addresses and the IPv4 protocol field results from the following facts: (1) Header checksums, present in IPv4 but not in IPv6, protect addresses or protocol integrity; (2) In IPv4, ICMP messages and null-checksum UDP datagram depend on this protection because, unlike other datagrams, they have no other address-and-protocol integrity protection. The sum MUST be performed in ordinary 2’s complement arithmetic.

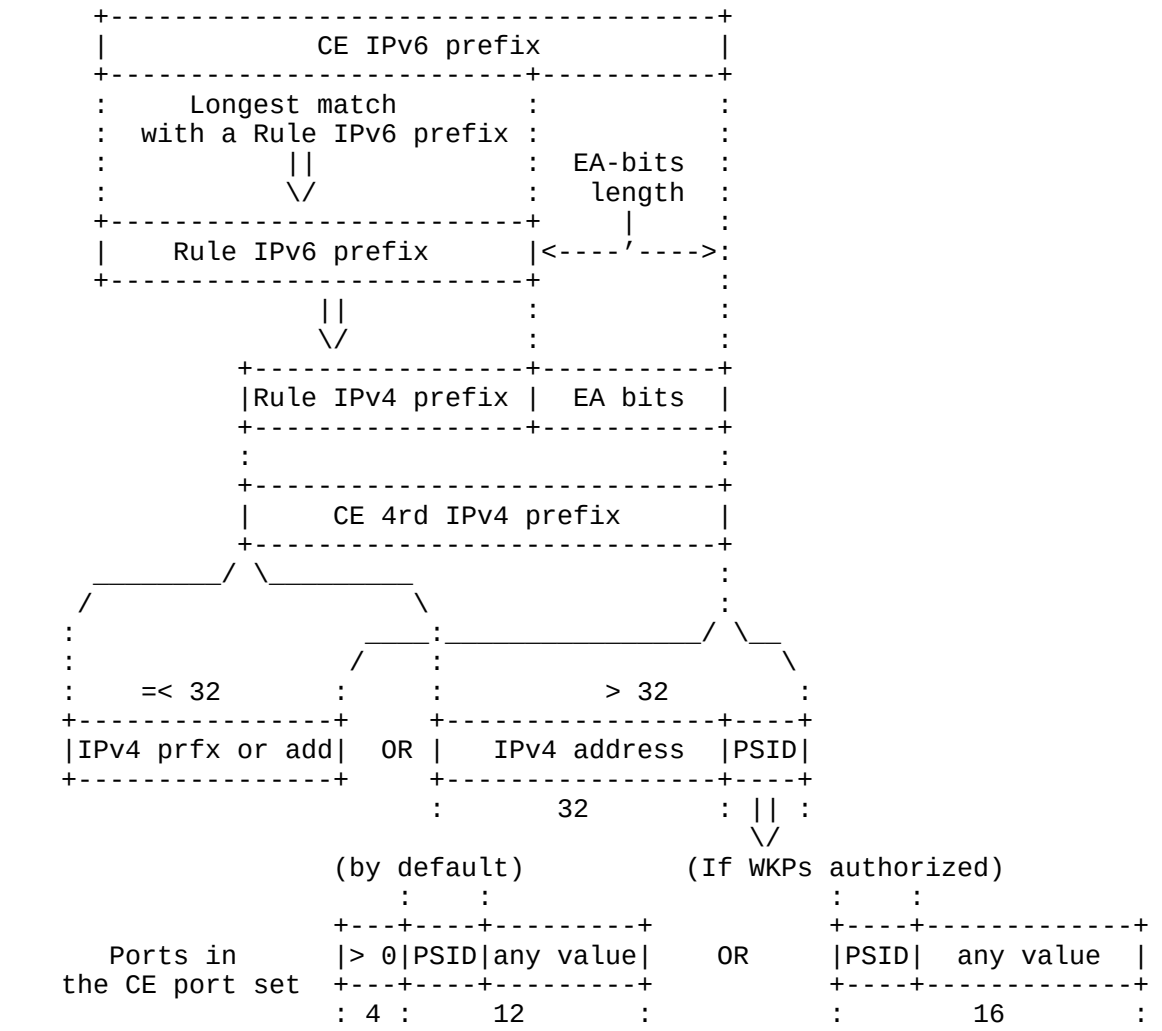
IP-layer Packet length is another field covered by the IPv4 IP-header checksum. It is not included in the saved checksum because: (1) doing so would have conflicted with [RFC6437] (flow labels must be the same in all packets of each flow); (2) ICMPv4 messages have good enough protection with their own checksums; (3) the UDP length field provides to null-checksum UDP datagrams the same level of protection after Domain traversal as without Domain traversal (consistency between IP-layer and UDP-layer lengths can be checked).

NOTE 2: TTL treatment has been chosen to permit adjacency tests between two IPv4 nodes situated at both ends of a 4rd tunnel. TTL values to be preserved for this are TTL=255 and TTL=1. For other values, TTL decrease between to IPv4 nodes is the same as though traversed IPv6 routers would be IPv4 routers.

Effect of this TTL treatment on IPv4 traceroute is specific: (1) the number of routers of the end-to-end path includes traversed IPv6 routers; (2) IPv6 routers of a Domain are listed after IPv4 routers of Domain entry and exit; (3) the IPv4 address shown for an IPv6 router is the IPv6-only dummy IPv4 address of Section 4.8; (4) the response time indicated for an IPv6 router is that of the next router.

NOTE 3: Provided the sum of obtained IPv4 addresses and protocol matches Addr_Prot_Cksm. If not, the packet MUST be silently discarded.

4.4. Address Mapping from CE IPv6 Prefixes to 4rd IPv4 prefixes



From CE IPv6 prefix to 4rd IPv4 address and Port set

Figure 4

R-7: A CE whose delegated IPv6 prefix matches the Rule IPv6 prefix of one or several Mapping rules MUST select the CE mapping rule for which the match is the longest. It then derives its 4rd IPv4 prefix as shown in Figure 4: (1) the CE replaces the Rule IPv6 prefix by the Rule IPv4 prefix. The result is the CE 4rd IPv4 prefix. (2) If this CE 4rd IPv4 prefix has less than 32 bits, the CE takes it as its assigned IPv4 prefix. If it has exactly 32 bits, the CE takes it as its IPv4 address. If it

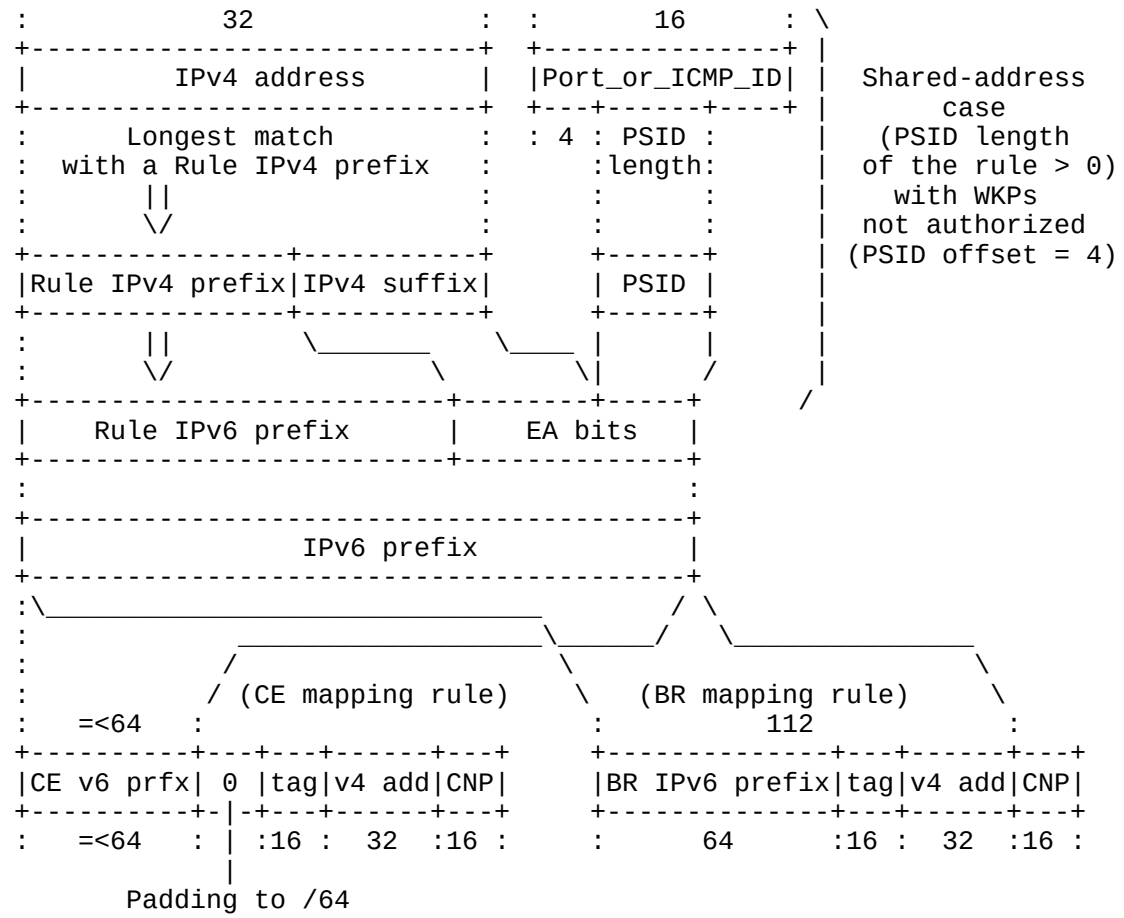
has more than 32 bits, the CE MUST takes the first 32 bits as its shared public IPv4 address, and bits beyond the first 32 as its Port-set identifier (PSID). Ports of its restricted port set are by default those that have any non-zero value in their first 4 bits (the PSID offset), followed by the PSID, and followed by any values in remaining bits. If the WKP authorized option applies to the Mapping rule, there is no 4-bit offset before the PSID so that all ports can be assigned.

NOTE: The choice of the default PSID position in Port fields has been guided by the following objectives: (1) for fairness, avoid having any of the well-known ports 0-1023 in the port set specified by any PSID value; (2) for compatibility RTP/RTCP [RFC4961], include in each port set pairs of consecutive ports; (3) in order to facilitate operation and training, have the PSID at a fixed position in port fields; (4) in order to facilitate documentation in hexadecimal notation, and to facilitate maintenance, have this position nibble aligned. Ports that are excluded from assignment to CEs are 0-4095 instead of just 0-1023 in a trade-off to favor nibble alignment of PSIDs and overall simplicity.

R-8: A CE whose delegated IPv6 prefix has its longest match with the Rule IPv6 prefix of the BR mapping rule MUST take as IPv4 address the 32 bit that, in the delegated IPv6 prefix, follow this Rule IPv6 prefix. If this is the case while the Hub&spoke option applies to the Domain, or if the Rule IPv6 prefix is not a /80, there is a configuration error in the Domain. An implementation-dependent administrative action MAY be taken.

A CE whose delegated IPv6 prefix matches the Rule IPv6 prefix of neither any CE Mapping rule nor the BR mapping rule, and is in a Domain that has a NAT64+ mapping rule, MUST be noted as having the unspecified IPv4 address.

4.5. Address Mapping from 4rd IPv4 addresses to 4rd IPv6 Addresses



From 4rd IPv4 address to 4rd IPv6 address

Figure 5

R-9: BRs, and CEs that are assigned public IPv4 addresses, shared or not, MUST derive 4rd IPv6 addresses from 4rd IPv4 addresses by the steps below or their functional equivalent (Figure 5 details the shared public IPv4 address case):

Note: the rules for forming 4rd specific Interface Identifiers is obey the latest specification of [RFC7136]. "Specifications of forms of 64-bit IID MUST specify how all 64 bits are set". And "the whole IID value MUST be viewed as an opaque bit string by third parties, except possibly in the local context."

(1) If Hub&spoke topology does not apply to the Domain, or if it applies but the IPv6 address to be derived is a source address from a CE or a destination address from a BR, find the CE mapping rule whose Rule IPv4 prefix has the longest match with the IPv4 address.

If no Mapping rule is thus obtained, take the BR mapping rule.

If the obtained Mapping rule assigns IPv4 prefixes to CEs, i.e. if length of the Rule IPv4 prefix plus EA-bits length is 32 - k, with k >= 0, delete the last k bits of the IPv4 address.

Otherwise, i.e. if length of the Rule IPv4 prefix plus EA-bits length is 32 + k, with k > 0, take k as PSID length, and append to the IPv4 address the PSID copied from bits p to p+3 of the Port_or_ICMP_ID field where: (1) p, the PSID offset, is 4 by default, and 0 if the WKPs authorized option applies to the rule; (2) The Port_or_ICMP_ID field is in bits of the IP payload that depend on whether the address is source or destination, on whether the packet is ICMP or not, and, if it is ICMP, whether it is an error message or an echo message. This field is:

a. If the packet Protocol is not ICMP, the port field associated with the address (bits 0-15 for a source address, and bits 16-31 for a destination address).

b. If the packet is an ICMPv4 echo or echo-reply message, the ICMPv4 Identification field (bits 32-47).

c. If the packet is an ICMPv4 error message, the port field associated with the address in the returned packet header (bits 240-255 for a source address, bits 224-239 for a destination address).

NOTE 1: Using Identification fields of ICMP messages as port fields permits to exchange Echo requests and Echo replies between shared-address CEs and IPv4 hosts having exclusive IPv4 addresses. Echo exchanges between two shared-address CEs remain impossible, but this is a limitation inherent to address sharing (one reason among many to use IPv6).

NOTE 2: When the PSID is taken in the port field of the IPv4 payload, it is, to avoid dependency on any particular layer-4 protocol having port fields, without checking that

the protocol is indeed one that has a port field . A packet may consequently go, in case of source mistake, from a BR to a shared-address CE with a protocol that is not supported by this CE. In this case, the CE NAT44 returns an ICMPv4 "protocol unreachable" error message. The IPv4 source is thus appropriately informed of its mistake.

(2) Replace in the result the Rule IPv4 prefix by the Rule IPv6 prefix.

(3) If the result is shorter than a /64, append to the result a null padding up to 64 bits, followed by the 4rd tag (0x0300), and followed by the IPv4 address.

NOTE: The 4rd tag is a 4rd-specific mark. Its function is to ensure that 4rd IPv6 addresses are recognizable by CEs without any interference with the choice of subnet prefixes in CE sites. (These choices may have been done before 4rd is enabled.)

For this, the 4rd tag has its "u" and "g" bits of [RFC4291] both set to 1, so that they maximumly differ from these existing IPv6 address schemas. So far, u=g=1 has not been used in any IPv6 addressing architecture.

With the 4rd tage, IPv6 packets can be routed to the 4rd function within a CE node based on a /80 prefix that no native-IPv6 address can contain.

(4) Add to the result a Checksum-neutrality preserver (CNP). Its value, in one’s complement arithmetic, is the opposite of the sum of 16-bit fields of the IPv6 address other than the IPv4 address and the CNP themselves (i.e. 5 consecutive fields in address-bits 0-79).

NOTE: CNP guarantees that Tunnel packets are valid IPv6 packets for all layer-4 protocols that use the same checksum algorithm as TCP. This guarantee does not depend on where checksum fields of these protocols are placed in IP payloads. (Today, such protocols are UDP [RFC0768], TCP [RFC0793], UDP-Lite [RFC3828], and DCCP [RFC5595]. Should new ones be specified, BRs will support them without needing an update.)

R-10: 4rd-capable CE SHOULD, and 4rd-enbaled CE MUST always prohibit all addresses that use its advertised prefix and have IID

starting with 0x0300 (4rd Tag), by using Duplicate Address Detection [RFC4862].

R-11: A CE that is assigned the unspecified IPv4 address (see Section 4.4) MUST use, for packets tunneled between itself and the Domain NAT64+, addresses as detailed in Figure 6: (a) for its IPv6 source, (b) as IPv6 destinations that depend on IPv4 destinations. A NAT64+, being NAT64 conforming [RFC6146], MUST accept IPv6 packets whose destination conforms to Figure 6 (b) (4rd tag instead of "u" and 0x00 octets). In its Binding Information Base, it MUST remember whether a mapping was created with a "u" or 4rd-tag destination. In the IPv4 to IPv6 direction, it MUST use 4rd tunneling, with source address conforming to Figure 6 (b), when using a mapping that was created with a 4rd-tag destination.

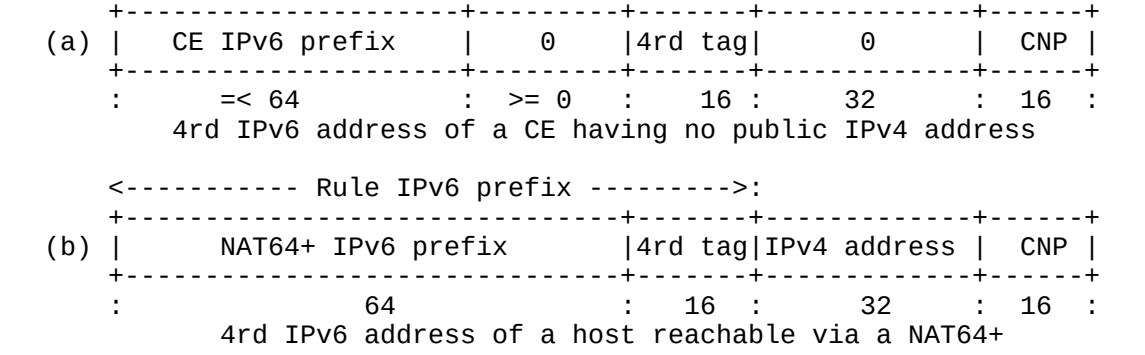


Figure 6

R-12: For anti-spoofing protection, CEs and BRs MUST check that the source address of each received Tunnel packet is that which, according to Section 4.5, is derived from the source 4rd IPv4 address. For this, the IPv4 address used to obtain the source 4rd IPv4 address is that embedded in the IPv6 source address (in its bits 80-111). (This verification is needed because IPv6 ingress filtering [RFC3704] applies only to IPv6 prefixes, without guarantee that Tunnel packets are built as specified in Section 4.5.)

R-13: For additional protection against packet corruption at a link layer that might be undetected at this layer during Domain traversal, CEs and BRs SHOULD verify that source and destination IPv6 addresses have not been modified. This can be done by checking that they remain checksum neutral (see the Note on CNP above).

4.6. Fragmentation Processing

4.6.1. Fragmentation at Domain Entry

R-14: If an IPv4 packet enters a CE or BR with a size such that the derived Tunnel packet would be longer than the Domain PMTU, the packet has to be either discarded or fragmented. The Domain-entry node MUST discard it if the packet has DF=1, with an ICMP error message returned to the source. It MUST fragment it otherwise, with the payload of each fragment not exceeding PMTU - 48. The first fragment has its offset equal to the received offset. Following fragments have offsets increased by lengths of previous-fragments payloads. Functionally, fragmentation is supposed to be done in IPv4 before applying to each fragment the reversible header translation of Section 4.3.

4.6.2. Ports of Fragments addressed to Shared-Address CEs

Because ports are available only in first fragments of IPv4 fragmented packets, a BR needs a mechanism to send to the right shared-address CEs all fragments of fragmented packets.

For this, a BR MAY systematically reassemble fragmented IPv4 packets before tunneling them. But this consumes large memory space, opens denial-of-service-attack opportunities, and can significantly increase forwarding delays. This is the reason for the following requirement:

R-15: BRs SHOULD support an algorithm whereby received IPv4 packets can be forwarded on the fly. The following is an example of such algorithm:

(1) At BR initialization, if at least one CE mapping rule concerns shared public IPv4 addresses (length of Rule IPv4 prefix + EA-bits length > 32), the BR initializes an empty "IPv4-packet table" whose entries have the following items:

- IPv4 source
- IPv4 destination
- IPv4 identification
- Destination port

(2) When the BR receives an IPv4 packet whose matching Mapping rule is one of shared public IPv4 addresses (length of Rule IPv4 prefix + EA-bits length > 32), the BR searches the table for an entry whose IPv4 source, IPv4 destination, and IPv4 Identification, are those of the received packet. The BR then performs actions detailed in Table 5 depending on which conditions hold.

- CONDITIONS -								
First Fragment (offset=0)	Y	Y	Y	Y	N	N	N	N
Last fragment (MF=0)	Y	Y	N	N	Y	Y	N	N
An entry has been found	Y	N	Y	N	Y	N	Y	N
- RESULTING ACTIONS -								
Create a new entry	-	-	-	X	-	-	-	-
Use port of the entry	-	-	-	-	X	-	X	-
Update port of the entry	-	-	X	-	-	-	-	-
Delete the entry	X	-	-	-	X	-	-	-
Forward the packet	X	X	X	X	X	-	X	-

Table 5

(3) The BR performs garbage collection for table entries that remain unchanged for longer than some limit. This limit, normally longer that the maximum time normally needed to reassemble a packet is not critical. It should however not be longer than 15 seconds [RFC0791].

R-16: For the above algorithm to be effective, CEs that are assigned shared public IPv4 addresses MUST NOT interleave fragments of several fragmented packets.

R-17: CEs that are assigned IPv4 prefixes, and are in nodes that route public IPv4 addresses rather than only using NAT44s, MUST have the same behavior as described just above for BRs.

4.6.3. Packet Identifications from Shared-Address CEs

When packets go from CEs that share the same IPv4 address to a common destination, a precaution is needed to guarantee that packet Identifications set by sources are different. Otherwise, packet reassembly at destination could otherwise be confused because it is based only on source IPv4 address and Identification. Probability of

such confusions may in theory be very low but, in order to avoid creating new attack opportunities, a safe solution is needed.

R-18: A CE that is assigned a shared public IPv4 address MUST only use packet Identifications that have the CE PSID in their bits 0 to PSID length - 1.

R-19: A BR or a CE that receives a packet from a shared-address CE MUST check that bits 0 to PSID length - 1 of their packet Identifications are equal to the PSID found in source 4rd IPv4 address.

4.7. TOS and Traffic-Class Processing

IPv4 TOS and IPv6 Traffic class have the same semantic, that of the differentiated-services field, or DS field, specified in [RFC2474] and [RFC6040]. Their first 6 bits contain a differentiated services codepoint (DSCP), and their two last bits can convey explicit congestion notifications (ECNs), which both may evolve during Domain traversal. [RFC2983] discusses how the DSCP can be handled by tunnel end points. The Tunnel traffic class option permits to ignore DS-field evolutions occurring during Domain traversal, if the desired behavior is that of generic tunnels conforming to [RFC2473].

R-20: Unless the Tunnel traffic class option is configured for the Domain, BRs and CEs MUST copy the IPv4 TOS into the IPv6 Traffic class at Domain entry, and copy back the IPv6 Traffic class into the IPv4 TOS at Domain exit.

R-21: If the Tunnel traffic class option is configured for a Domain, BRs and CEs MUST at Domain entry take the configured Tunnel traffic class as IPv6 Traffic class, and copy the received IPv4 TOS into the IPv4_TOS of the fragment header (Figure 3). At Domain exit, they MUST copy back the IPv4_TOS of the fragment header into the IPv4 TOS.

4.8. Tunnel-Generated ICMPv6 Error Messages

If a Tunnel packet is discarded on its way across a 4rd domain because of an unreachable destination, an ICMPv6 error message is returned to the IPv6 source. For the IPv4 source of the discarded packet to be informed of packet loss, the ICMPv6 message has to be converted into an ICMPv4 message.

R-22: If a CE or BR receives an ICMPv6 error message [RFC4443], it MUST synthesize an ICMPv4 error packet [RFC0792]. This packet MUST contain the first 8 octets of the discarded-packet IP

payload. The reserved IPv4 dummy address (TBD, (see Section 6) MUST be used as its source address .

Like in [RFC6145], ICMPv6 Type = 1 and Code = 0 (Destination unreachable, No route to destination") MUST be translated into ICMPv4 Type = 3 and Code = 0 (Destination unreachable, Net unreachable), and ICMPv6 Type = 3 and Code = 0 (Time exceeded, Hop limit exceeded in transit) MUST be translated into ICMPv4 Type = 11 and Code = 0 (Destination unreachable, Net unreachable).

4.9. Provisioning 4rd Parameters to CEs

Domain parameters listed in Section 4.2 are subject to the following constraints:

R-23: Each Domain MUST have a BR mapping rule and/or a NAT64+ mapping rule. (The BR mapping rule is only used by CEs that are assigned public IPv4 addresses, shared or not. The NAT64+ mapping rule is only used by CEs that are assigned the unspecified IPv4 address (Section 4.4), and therefore need an ISP NAT64 to reach IPv4 destinations.

R-24: Each CE and each BR MUST support up to 32 Mapping rules.

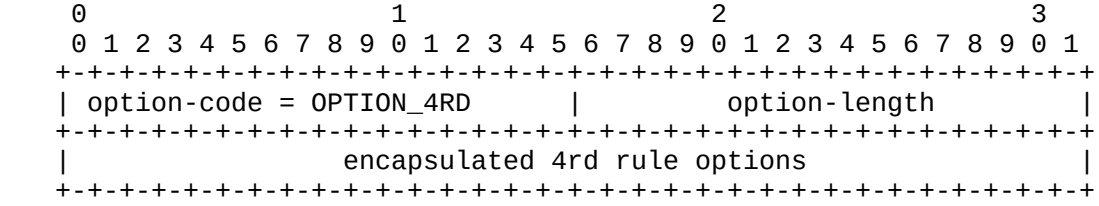
This number of is to ensure that independently acquired CEs an BR nodes can always interwork.

ISPs that need Mapping rules for more IPv4 prefixes than this number SHOULD split their networks into multiple Domains. Communication between these domains can be done in IPv4, or by some implementation-dependent but equivalent other means.

R-25: For mesh topologies, where CE-CE paths don’t go via BRs, all mapping rules of the Domain MUST be sent to all CEs. For hub-and-spoke topologies, where all CE-CE paths go via BRs, each CE MAY be sent only the BR mapping rule of the Domain plus, if different, the CE mapping rule that applies to its CE IPv6 prefix.

R-26: In a Domain where the chosen topology is Hub&spoke, all CEs MUST have IPv6 prefixes that match a CE mapping rule. (Otherwise, packets sent to CEs whose IPv6 prefixes would match only the BR mapping rule would, with longest-match selected routes, be routed directly to these CEs. This would be contrary to the Hub&spoke requirement).

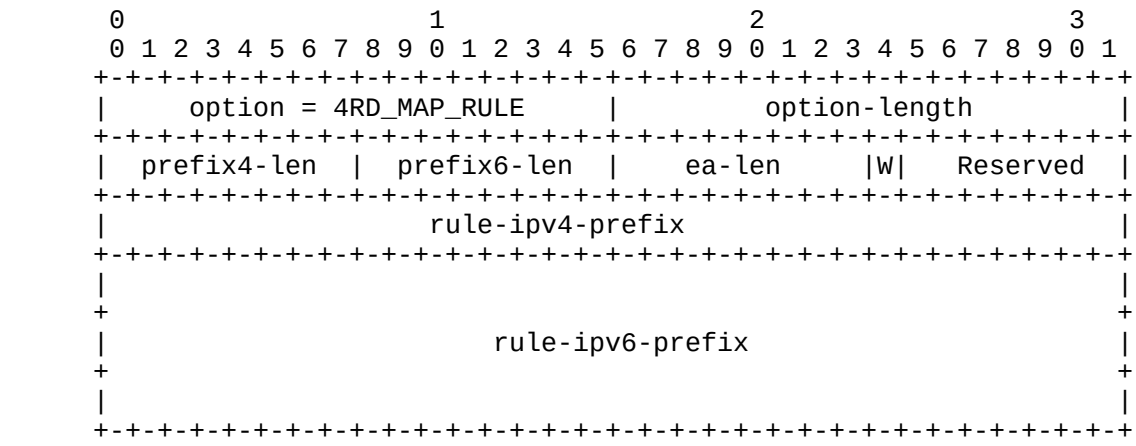
R-27: CEs MUST be able to acquire parameters of 4rd domains (Section 4.2) in DHCPv6 (ref. [RFC2131]). Formats of DHCPv6 options to be used are detailed in Figure 7, Figure 8, and Figure 9 with field values specified after each Figure.



DHCPv6 option for 4rd

Figure 7

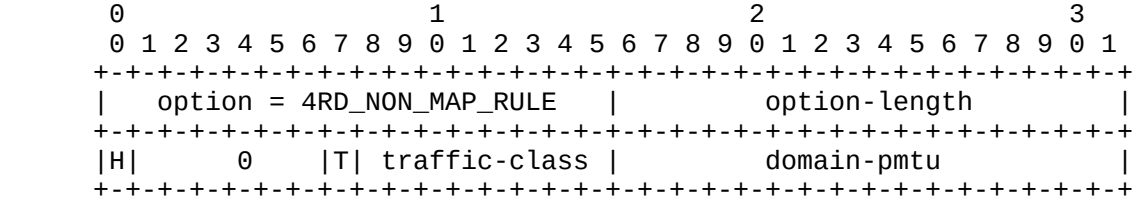
- option-code: TBD1, OPTION_4RD (see Section 6)
- option-length: the length of encapsulated options in octets
- encapsulated 4rd rule options: the 4RD DHCPv6 option contains at least one encapsulated 4RD_MAP_RULE option and maximum one encapsulated 4RD_NON_MAP_RULE option. Since DHCP servers normally send whatever options the operator configures, operators should be advised to configure these options appropriately. DHCP servers MAY check to see that the configuration follows these rules and notify the operator in an implementation-dependent manner if the settings for these options aren’t valid. The length of encapsulated options is in octets.



Encapsulated option for Mapping-rule parameters

Figure 8

- option-code: TBD2, encapsulated 4RD_MAP_RULE option (see Section 6)
- option-length: 20
- prefix4-len: number of bits of the Rule IPv4 prefix
- prefix6-len: number of bits of the Rule IPv6 prefix
- ea-len: EA-bits length
- W: WKP authorized, = 1 if set
- rule-ipv4-prefix: the Rule IPv4 prefix, left aligned
- rule-ipv6-prefix: Rule IPv6 prefix, left aligned



Encapsulated option for non-mapping-rule parameters of 4rd-domains

Figure 9

- option-code: TBD3, encapsulated 4RD_NON_MAP_RULE option (see Section 6)
- option-length: 4
- H: Hub&spoke topology (= 1 if Yes)
- T: Traffic-class flag (= 1 if a Tunnel traffic class is provided)
- traffic-class: Tunnel-traffic class
- domain-pmtu: Domain PMTU (at least 1280)

Other means than DHCPv6 that may prove useful to provide 4rd parameters to CEs are off-scope for this document. The same or similar parameter formats would however be recommended to facilitate training and operation.

5. Security Considerations

Spoofing attacks

With IPv6 ingress filtering effective in the Domain [RFC3704], as required in Section 3 (Figure 1 in particular), and with consistency checks between 4rd IPv4 and IPv6 addresses of Section 4.5, no spoofing opportunity in IPv4 is introduced by 4rd: being able to use as source IPv6 address only one that has been allocated to him, a customer can only provide as source 4rd IPv4 address that which derives this IPv6 address according to Section 4.5, i.e. one that his ISP has allocated to him.

Routing-loop attacks

Routing-loop attacks that may exist in some automatic-tunneling scenarios are documented in [RFC6324]. No opportunity for routing-loop attacks has been identified with 4rd.

Fragmentation-related attacks

As discussed in Section 4.6, each BR of a Domain that assigns shared public IPv4 should maintain a dynamic table for fragmented packets that go to these shared-address CEs.

This opens a BNR vulnerability to a denial of service attack from hosts that would send very large numbers of first fragments and would never send last fragments having the same packet

identifications. This vulnerability is inherent to IPv4 address sharing, be it static or dynamic. Compared to what it is with algorithms that reassemble IPv4 packets in BRs, it is however significantly mitigated by the algorithm of Section 4.6.2 which uses much less memory space.

6. IANA Considerations

IANA is requested to allocate the following:

- One DHCPv6 option codes TBD1 for OPTION_4RD of Section 4.9 respectively (to be added to section 24.3 of [RFC3315]. Encapsulated options of OPTION_4RD, 4RD_MAP_RULE (TBD2) and 4RD_NON_MAP_RULE (TBD3) should also be recorded into the DHCPv6 option code space.

Value	Description	Reference
TBD1	OPTION_4RD	this document
TBD2	4RD_MAP_RULE	this document
TBD3	4RD_NON_MAP_RULE	this document

- A reserved IPv4 address to be used as the "IPv4 dummy address" of Section 4.8. Its proposed value is 192.0.0.8/32 (Section 4.8).

7. Relationship with Previous Works

The present specification has been influenced by many previous IETF drafts, in particular those accessible at http://tools.ietf.org/html/draft-xxxx where xxxx are the following (in order of their first versions):

- bagnulo-behave-nat64 (2008-06-10)
- xli-behave-ivi (2008-07-06)
- despres-sam-scenarios (2008-09-28)
- boucadair-port-range (2008-10-23)
- ymbk-aplusp (2008-10-27)
- xli-behave-divi (2009-10-19)
- thaler-port-restricted-ip-issues (2010-02-28)
- cui-softwire-host-4over6 (2010-05-05)

- xli-behave-divi-pd (2011-07-02)
- dec-stateless-4v6 (2011-03-05)
- matsushima-v6ops-transition-experience (2011-03-07)
- despres-intarea-4rd (2011-03-07)
- deng-aplusp-experiment-results (2011-03-08)
- murakami-softwire-4rd (2011-07-04)
- operators-softwire-stateless-4v6-motivation (2011-05-05)
- murakami-softwire-4v6-translation (2011-07-04)
- despres-softwire-4rd-addmapping (2011-08-19)
- boucadair-softwire-stateless-requirements (2011-09-08)
- chen-softwire-4v6-add-format (2011-10-2)
- mawatari-softwire-464xlat (2011-10-16)
- mdt-softwire-map-dhcp-option (2011-10-24)
- mdt-softwire-mapping-address-and-port (2011-11-25)
- mdt-softwire-map-translation (2012-01-10)
- mdt-softwire-map-encapsulation (2012-01-27)

8. Acknowledgements

This specification has benefited over several years from independent proposals, questions, comments, constructive suggestions, and useful criticisms, coming from numerous IETF contributors.

Authors would like to express recognition to all these contributors, and more especially to the following, in alphabetical order of first names: Brian Carpenter, Behcet Sarikaya, Bing Liu, Cameron Byrne, Congxiao Bao, Dan Wing, Derek Atkins, Erik Kline, Francis Dupont, Gabor Bajko, Gang Chen, Hui Deng, Jan Zorz, Jacni Quin (who was an active co-author of some earlier versions of this specification), James Huang, Jari Arkko, Kathleen Moriarty, Laurent Toutain, Leaf Yeh, Lorenzo Colitti, Mark Townsley, Marcello Bagnulo, Mohamed Boucadair, Nejc Skoberne, Olaf Maennel, Ole Troan, Olivier Vautrin, Peng Wu, Qiong Sun, Rajiv Asati, Ralph Droms, Randy Bush, Satoru

Matsushima, Simon Perreault, Stuart Cheshire, Teemu Savolainen, Tetsuya Murakami, Tomasz Mrugalski, Tina Tsou, Tomasz Mrugalski, Ted Lemon, Suresh Krishnan, Washam Fan, Wojciech Dec, Xiaohong Deng, Xing Li, and Yu Fu.

9. References

9.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[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.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, October 2000.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire Problem Statement", RFC 4925, July 2007.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, October 2007.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, November 2010.

9.2. Informative References

[I-D.ietf-softwire-stateless-4v6-motivation] Boucadair, M., Matsushima, S., Lee, Y., Bonness, O., Borges, I., and G. Chen, "Motivations for Carrier-side Stateless IPv4 over IPv6 Migration Solutions", draft-ietf-softwire-stateless-4v6-motivation-05 (work in progress), November 2012.

[I-D.shirasaki-nat444] Yamagata, I., Shirasaki, Y., Nakagawa, A., Yamaguchi, J., and H. Ashida, "NAT444", draft-shirasaki-nat444-06 (work in progress), July 2012.

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.

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

[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.

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, March 2004.

[RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., and G. Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)", RFC 3828, July 2004.

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

[RFC4961] Wing, D., "Symmetric RTP / RTP Control Protocol (RTCP)", BCP 131, RFC 4961, July 2007.

[RFC5595] Fairhurst, G., "The Datagram Congestion Control Protocol (DCCP) Service Codes", RFC 5595, September 2009.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[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.

[RFC6324] Nakibly, G. and F. Templin, "Routing Loop Attack Using IPv6 Automatic Tunnels: Problem Statement and Proposed Mitigations", RFC 6324, August 2011.

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

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, November 2011.

[RFC6535] Huang, B., Deng, H., and T. Savolainen, "Dual-Stack Hosts Using "Bump-in-the-Host" (BIH)", RFC 6535, February 2012.

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

[RFC7136] Carpenter, B. and S. Jiang, "Significance of IPv6 Interface Identifiers", RFC 7136, February 2014.

Appendix A. Textual representation of Mapping rules

In the next sections, each Mapping rule will be represented as follows, using 0bXXX to represent binary number XXX, and square brackets [ ] for what is optional:

```
   {Rule IPv4 prefix, EA-bits length, Rule IPv6 prefix
      [, WKPs authorized]}

   EXAMPLES:
    {0.0.0.0/0, 32, 2001:db8:0:1:300::/80}
                               a BR mapping rule
    {198.16.0.0/14, 22, 2001:db8:4000::/34}
                               a CE mapping rule
    {0.0.0.0/0, 32, 2001:db8:0:1::/80}
                               a NAT64+ mapping rule)
    {198.16.0.0/14, 22, 2001:db8:4000::/34, Yes}
                               a CE mapping rule and Hub&spoke Topology
```

Appendix B. Configuring multiple Mapping Rules

As far as mapping rules are concerned, the simplest deployment model is that in which the Domain has only one rule (the BR mapping rule). To assign an IPv4 address to a CE in this model, an IPv6 /112 is assigned to it comprising the BR /64 prefix, the 4rd tag, and the IPv4 address. This model has however the following limitations: (1) shared IPv4 addresses are not supported; (2) IPv6 prefixes used for 4rd are too long to be used also for native IPv6 addresses; (3) if the IPv4 address space of the ISP is split with many disjoint IPv4 prefixes, the IPv6 routing plan must be as complex as an IPv4 routing plan based on these prefixes.

With more mapping rules, CE prefixes used for 4rd can be those used for native IPv6. How to choose CE mapping rules for a particular deployment needs not being standardized.

The following is only a particular pragmatic approach that can be used for various deployment scenarios. It is used in some of the use cases that follow.

(1) Select a "Common_IPv6_prefix" that will appear at the beginning of all 4rd CE IPv6 prefixes.

(2) Choose all IPv4 prefixes to be used, and assign one of them to each CE mapping rule i.

(3) For each CE mapping rule i, do the following:

   A. choose the length of its Rule IPv6 prefix (possibly the same for all CE mapping rules).

   B. Determine its PSID_length(i). A CE mapping rule that assigns shared addresses with a sharing ratio 2^Ki, has PSID_length = Ki. A CE mapping rule rule that assigns IPv4

prefixes of length L < 32, is considered to have a negative PSID_length = L - 32.

C. Derive EA-bits length (i) = 32 - L(Rule IPv4 prefix(i)) + PSID_length(i).

D. Derive the length of Rule_code(i), the prefix to be appended to the Common prefix to get the Rule IPv6 prefix of rule i:

```
L(Rule_code(i)) = L(CE IPv6 prefix(i))
                - L(Common_IPv6_prefix]
                - (32 - L(Rule IPv4 prefix(i)))
                - PSID_length(i)
```

E. Derive Rule_code(i) with the following constraints: (1) its length is L(Rule_code(i); it does not overlap with any of the previously obtained Rule codes (for instance, 010, and 01011 do overlap, while 00, 011, and 010 do not); it has the lowest possible value as a fractional binary number (for instance, 0100 < 10 < 11011 < 111). Thus, rules whose Rule_code lengths are 4, 3 , 5, and 2, give Rule_codes 0000, 001, 00010, and 01)

F. Take Rule IPv6 prefix(i)= the Common_IPv6_prefix followed by Rule_code(i).

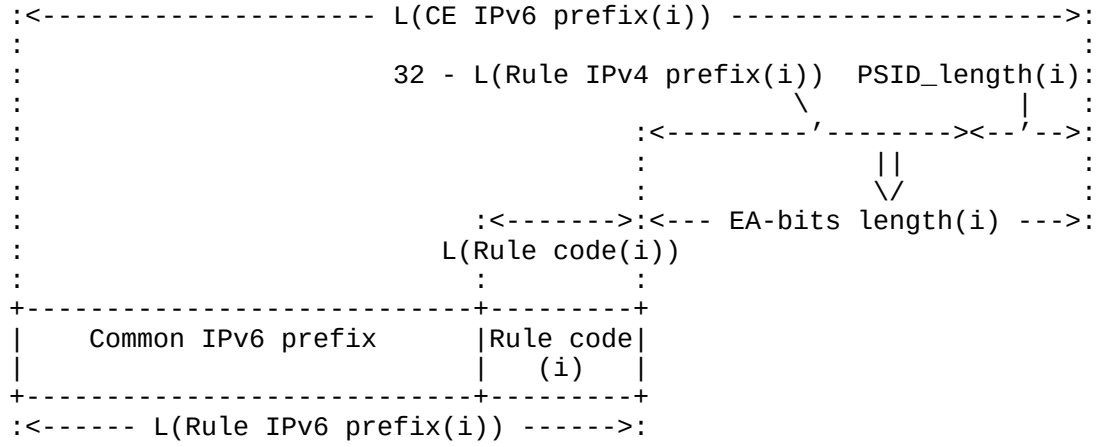


Figure 10

Appendix C. ADDING SHARED IPv4 ADDRESSES TO AN IPv6 NETWORK

C.1. With CEs within CPEs

We consider an ISP that offers IPv6-only service to up to 2^20 customers. Each customer is delegated a /56, starting with common prefix 2001:db8:0::/36. It wants to add public IPv4 service to customers that are 4rd-capable. It prefers to do it with stateless operation in its nodes, but has largely less IPv4 addresses than IPv6 addresses so that a sharing ratio is necessary.

The only IPv4 prefixes it can use are 192.8.0.0/15, 192.4.0.0/16, 192.2.0.0/16, and 192.1.0.0/16 (neither overlapping nor aggregetable). This gives 2^(32-15) + 3*2^(32-16) IPv4 addresses, i.e. 2^18 + 2^16). For the 2^20 customers to have the same sharing ratio, the number of IPv4 addresses to be shared has to be a power of 2. The ISP can therefore renounce to use one /16, say the last one. (Whether it could be motivated to return it to its Internet Registry is off-scope for this document.) The sharing ratio to apply is then 2^20 / 2^18 = 2^2 = 4, giving a PSID length of 2.

Applying principles of Appendix B with L[Common IPv6 prefix] = 36, L[PSID] = 2 for all rules, and L[CE IPv6 prefix(i)] = 56 for all rules, Rule codes and Rule IPv6 prefixes are:

CE Rule IPv4 prefix	EA bits length	Rule-Code length	Code (binary)	CE Rule IPv6 prefix
192.8.0.0/15	19	1	0	2001:db8:0::/37
192.4.0.0/16	18	2	10	2001:db8:800::/38
192.2.0.0/16	18	2	11	2001:db8:c00::/38

Mapping rules are then the following:

```
{192.8.0.0/15, 19, 2001:0db8:0000::/37}
{192.4.0.0/16, 18, 2001:0db8:0800::/38}
{192.2.0.0/16, 18, 2001:0db8:0c00::/38}
{0.0.0.0/0,    32, 2001:0db8:0000:0001:300::/80}
```

The CE whose IPv6 prefix is, for example, 2001:db8:0bbb:bb00::/56, derives its IPv4 address and its port set as follows (Section 4.4):

```
   CE IPv6 prefix     : 2001:0db8:0bbb:bb00::/56
   Rule IPv6 prefix(i): 2001:0db8:0800::/38 (longest match)
   EA-bits length(i)  : 18
   EA bits            : 0b11 1011 1011 1011 1011
   Rule IPv4 prefix(i): 0b1100 0000 0000 0100 (192.4.0.0/16)
   IPv4 address       : 0b1100 0000 0000 0100 1110 1110 1110 1110
                      : 192.4.238.238
   PSID               : 0b11
   Ports              : 0bYYYY 11XX XXXX XXXX
                        with YYYY > 0, and X...X any value
```

An IPv4 packet sent to address 192.4.238.238 and port 7777 is tunneled to the IPv6 address obtained as follows (Section 4.5):

```
   IPv4 address       : 192.4.238.238 (0xC004 EEEE)
                      : 0b1100 0000 0000 0100 1110 1110 1110 1110
   Rule IPv4 prefix(i): 192.4.0.0/16  (longest match)
                      : 0b1100 0000 0000 0100
   IPv4 suffix (i)    : 0b1110 1110 1110 1110
   EA-bits length (i) : 18
   PSID length (i)    : 2  (= 16 + 18 - 32)
   Port field         : 0b 0001 1110 0110 0001 (7777)
   PSID               : 0b11
   Rule IPv6 prefix(i): 2001:0db8:0800::/38
   CE IPv6 prefix     : 2001:0db8:0bbb:bb00::/56
   IPv6 address       : 2001:0db8:0bbb:bb00:300:c004:eeee:YYYY
                        with YYYY = the computed CNP
```

C.2. With some CEs behind Third-party Router CPEs

We now consider an ISP that has the same need as in the previous section except that, instead of using only its own IPv6 infrastructure, it uses that of a third-party provider, and that some of its customers use CPEs of this provider to use specific services it offers. In these CPEs, a non-zero index is used to route IPv6 packets to the physical port to which CEs are attached, say 0x2. Each such CPE delegates to the CE nodes the customer-site IPv6 prefix followed by this index.

The ISP is supposed to have the same IPv4 prefixes as in the previous use case, 192.8.0.0/15, 192.4.0.0/16, and 192.2.0.0/16, and to use the same Common IPv6 prefix, 2001:db8:0::/36.

We also assume that only a minority of customers use third-party CPEs, so that it is sufficient to use only one of the two /16s for them.

Mapping rules, are then (see Appendix C.1):

```
          {192.8.0.0/15, 19, 2001:0db8:0000::/37}
          {192.4.0.0/16, 18, 2001:0db8:0800::/38}
          {192.2.0.0/16, 18, 2001:0db8:0c00::/38}
          {0.0.0.0/0,    32, 2001:0db8:0000:0001:300::/80}
```

CEs that are behind third-party CPEs derive their own IPv4 addresses and port sets as in Appendix C.1.

In a BR, and also in a CE if the topology is mesh, the IPv6 address that is derived from IPv4 address 192.4.238.238 and port 7777 is obtained as in the previous section, except for the two last steps which are modified:

```
  IPv4 address       : 192.4.238.238 (0xC004 EEEE)
                     : 0b1100 0000 0000 0100 1110 1110 1110 1110
  Rule IPv4 prefix(i): 192.4.0.0/16  (longest match)
                     : 0b1100 0000 0000 0100
  IPv4 suffix (i)    : 0b1110 1110 1110 1110
  EA-bits length (i) : 18
  PSID length (i)    : 2  (= 16 + 18 - 32)
  Port field         : 0b 0001 1110 0110 0001 (7777)
  PSID               : 0b11
  Rule IPv6 prefix(i): 2001:0db8:0800::/38
  CE IPv6 prefix       : 2001:0db8:0bbb:bb00::/60
  IPv6 address         : 2001:0db8:0bbb:bb00:300:192.4.238.238:YYYY
                         with YYYY = the computed CNP
```

Appendix D. REPLACING DUAL-STACK ROUTING BY IPv6-ONLY ROUTING

In this use case, we consider an ISP that offers IPv4 service with public addresses individually assigned to its customers. It also offers IPv6 service, having deployed for this dual-stack routing. Because it provides its own CPEs to customers, it can upgrade all its CPEs to support 4rd. It wishes to take advantage of this capability to replace dual-stack routing by IPv6-only routing without changing any IPv4 address or IPv6 prefix.

For this, the ISP can use the single-rule model described at the beginning of Appendix B. If the prefix routed to BRs is chosen to start with 2001:db8:0:1::/64, this rule is:

```
   {0.0.0.0/0, 32, 2001:db8:0:1:300::/80}
```

All what is needed in the network before disabling IPv4 routing is the following:

- In all routers, where there is an IPv4 route toward x.x.x.x/n, add a parallel route toward 2001:db8:0:1:300:x.x.x.x::/(80+n)

- Where IPv4 address x.x.x.x was assigned to a CPE, now delegate IPv6 prefix 2001:db8:0:1:300:x.x.x.x::/112.

NOTE: In parallel with this deployment, or after it, shared IPv4 addresses can be assigned to IPv6 customers. It is sufficient that IPv4 prefixes used for this be different from those used for exclusive-address assignments. Under this constraint, Mapping rules can be set up according to the same principles as those of Appendix C.

Appendix E. ADDING IPv6 AND 4rd SERVICE TO A NET-10 NETWORK

In this use case, we consider an ISP that has only deployed IPv4, possibly because some of its network devices are not yet IPv6 capable. Because it did not have enough IPv4 addresses, it has assigned private IPv4 addresses of [RFC1918] to customers, say 10.x.x.x. It thus supports up to 2^24 customers (a "Net-10" network, using the NAT444 model of [I-D.shirasaki-nat444]).

Now, it wishes to offer IPv6 service without further delay, using for this 6rd [RFC5969]. It also wishes to offer incoming IPv4 connectivity to its customers with a simpler solution than that of PCP [RFC6887].

This appendix describes an example that adds IPv6 (using 6rd) and 4rd services to the "Net-10" private IPv4 network.

The IPv6 prefix to be used for 6rd is supposed to be 2001:db8::/32, and the public IPv4 prefix to be used for shared addresses is supposed to be 198.16.0.0/16 (0xc610). The resulting sharing ratio is 2^24 / 2^(32-16) = 256, giving a PSID length of 8.

The ISP installs one or several BRs, at its border to the public IPv4 Internet. They support 6rd, and 4rd above it. The BR prefix /64 is supposed to be that which is derived from IPv4 address 10.0.0.1 (i.e. 2001:db8:0:100:/64).

In accordance with [RFC5969], 6rd BRs are configured with the following parameters IPv4MaskLen = 8, 6rdPrefix = 2001:db8::/32; 6rdBRIPv4Address = 192.168.0.1 (0xC0A80001).

4rd Mapping rules are then the following:

```
              {198.16.0.0/16, 24, 2001:db8:0:0:300::/80}
              {0.0.0.0/0,      32, 2001:db8:0:100:300:/80,}
```

Any customer device that supports 4rd in addition to 6rd can then use its assigned shared IPv4 address with 240 assigned ports.

If its NAT44 supports port forwarding to provide incoming IPv4 connectivity (statically, or dynamically with UPnP an/or NAT-PMP), it can use it with ports of the assigned port set (a possibility that does not exist in Net-10 networks without 4rd/6rd).

Authors' Addresses

Remi Despres
RD-IPtech
3 rue du President Wilson
Levallois
France

Email: despres.remi@laposte.net

Sheng Jiang (editor)
Huawei Technologies Co., Ltd
Q14, Huawei Campus, No.156 BeiQing Road
Hai-Dian District, Beijing 100095
P.R. China

Email: jiangsheng@huawei.com

Reinaldo Penno
Cisco Systems, Inc.
170 West Tasman Drivee
San Jose, California 95134
USA

Email: repenno@cisco.com

Yiu Lee
Comcast
One Comcast Center
Philadelphia, PA 1903
USA

Email: Yiu_Lee@Cable.Comcast.com

Gang Chen
China Mobile
53A, Xibianmennei Ave.
Xuanwu District, Beijing 100053
China

Email: phdgang@gmail.com

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

Email: fibrib@gmail.com

Softwire Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 17, 2015

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

November 13, 2014

Lightweight 4over6: An Extension to the DS-Lite Architecture
draft-ietf-softwire-lw4over6-13

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 May 17, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Conventions 4
3. Terminology 4
4. Lightweight 4over6 Architecture 5
5. Lightweight B4 Behavior 7
5.1. Lightweight B4 Provisioning with DHCPv6 7
5.2. Lightweight B4 Data Plane Behavior 9
5.2.1. Fragmentation Behaviour 11
6. Lightweight AFTR Behavior 11
6.1. Binding Table Maintenance 11
6.2. lwAFTR Data Plane Behavior 12
7. Additional IPv4 address and Port Set Provisioning Mechanisms 13
8. ICMP Processing 14
8.1. ICMPv4 Processing by the lwAFTR 14
8.2. ICMPv4 Processing by the lwB4 14
9. Security Considerations 15
10. IANA Considerations 15
11. Author List 15
12. Acknowledgement 19
13. References 19
13.1. Normative References 19
13.2. Informative References 20
Authors' Addresses 21

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 an IPv4-in-IPv6 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 is a solution designed specifically for complete independence between IPv6 subnet prefix and IPv4 address with or without IPv4 address sharing. This is accomplished by maintaining state for each softwire (per-subscriber state) in the central lwAFTR and a hub-and-spoke forwarding architecture. [I-D.ietf-softwire-map]

also offers these capabilities or, alternatively, allows for a reduction of the amount of centralized state using rules to express IPv4/IPv6 address mappings. This introduces an algorithmic relationship between the IPv6 subnet and IPv4 address. This relationship also allows the option of direct, meshed connectivity between users.

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 extends the mechanism defined in [RFC7040] by allowing address sharing. The solution in this document 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:

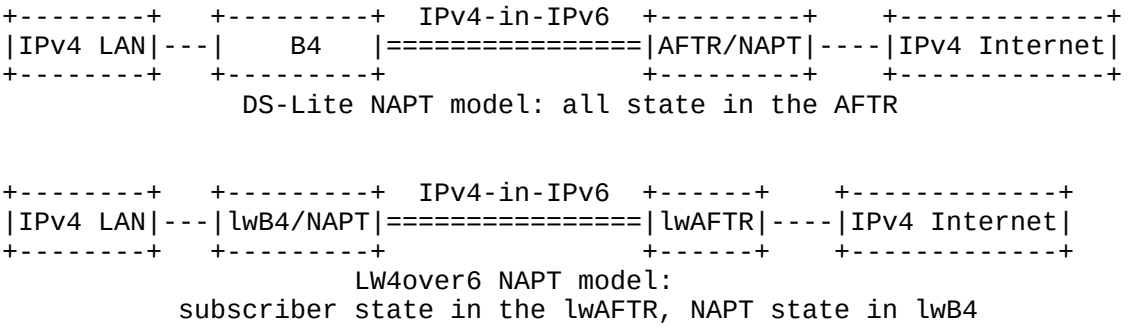


Figure 1 Comparison of DS-Lite and Lightweight 4over6 Data Plane

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

- o The lwB4, which performs the NAPT function and encapsulation/de-capsulation IPv4/IPv6.
- o The lwAFTR, which performs the encapsulation/de-capsulation IPv4/IPv6.
- o 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, Port Control Protocol (PCP, [RFC6887]) 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 precise mechanism whereby this synchronization occurs is out of scope for this document.

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

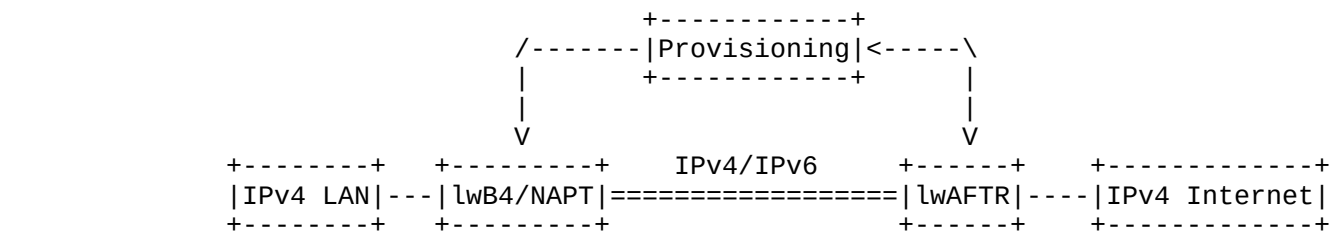


Figure 2 Lightweight 4over6 Provisioning Synchronization

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, 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
- IPv6 Binding Prefix

The lwB4 MUST implement DHCPv6 based configuration using OPTION_S46_CONT_LW as described in section 5.3 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, DHCP 4o6 [RFC7341] MUST be used.

Although it would be possible to extend lw4o6 to have more than one active lw4o6 tunnel configured simultaneously, this document is only concerned with the use of a single tunnel.

The IPv6 binding prefix field is provisioned so that the CE can identify the correct prefix to use as the tunnel source. On receipt of the necessary configuration parameters listed above, the lwB4 performs a longest prefix match between the IPv6 binding prefix and its currently active IPv6 prefixes. The result forms the subnet to

be used for sourcing the lw4o6 tunnel. The full /128 address is then constructed in the same manner as [I-D.ietf-softwire-map].

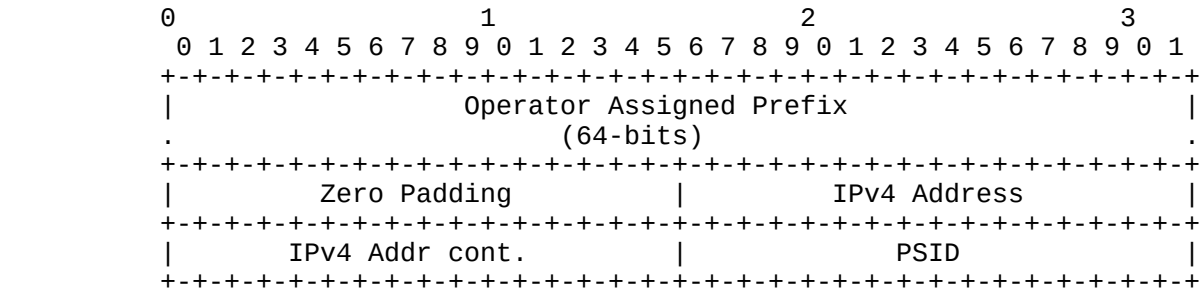


Figure 3 Construction of the lw4o6 /128 Prefix

Operator Assigned Prefix: IPv6 prefix allocated to the client. If the prefix length is less than 64, right padded with zeros to 64-bits.

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 IPv6 encapsulation source address is changed for any reason (such as the DHCPv6 lease expiring), the lwB4's dynamic provisioning process MUST be re-initiated. When the lwB4's public IPv4 address or port set ID is changed for any reason, the lwB4 MUST flush its NAPT table.

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, thus allocating a single contiguous port set to each lwB4.

Provisioning of the lwB4 using DHCPv6 as described here allocates a single PSID to the client. In the event that the client is concurrently using all of the provisioned L4 ports it may be unable to initiate any additional outbound connections. DHCPv6 based provisioning does not provide a mechanism for the client to request more L4 port numbers. Other provisioning mechanisms (e.g. PCP based

provisioning [I-D.ietf-pcp-port-set]) provide this function. Issues relevant to IP address sharing are discussed in more detail in [RFC6269].

Unless an lwB4 is being allocated a full IPv4 address, it is RECOMMENDED that PSIDs containing the system ports (0-1023) are not allocated to lwB4s. The reserved ports are more likely to be reserved by middleware, and therefore we recommend that they not be issued to clients other than as a deliberate assignment. Section 5.2.2 of [RFC6269] provides analysis of allocating system ports to clients with IPv4 address sharing.

In the event that the lwB4 receives an ICMPv6 error message (type 1, code 5) originating from the lwAFTR, the lwB4 interprets this to mean that no matching entry in the lwAFTR's binding table has been found, so the IPv4 payload is not being forwarded by the lwAFTR. The lwB4 MAY then re-initiate the dynamic port-restricted provisioning process. The lwB4's re-initiation policy SHOULD be configurable.

On receipt of such an ICMP error message, the lwB4 MUST validate the source address to be the same as the lwAFTR address that is configured. In the event that these addresses do not match, the lwAFTR MUST discard the ICMP error message.

In order to prevent forged ICMP messages (using the spoofed lwAFTR address as the source) from being sent to lwB4s, the operator can implement network ingress filtering as described in [RFC2827].

The DNS considerations described in Section 5.5 and Section 6.4 of [RFC6333] apply to Lightweight 4over6; lw4o6 implementations MUST comply with all requirements stated there.

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]).
7.1 Tunneling	Covering tunneling and traffic class mapping between IPv4 and IPv6

(referencing [RFC2473] and [RFC2983])

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.

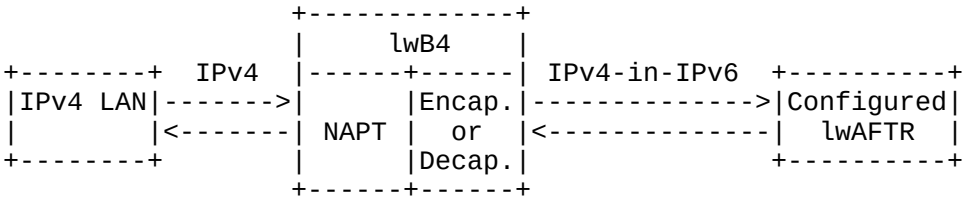


Figure 4 Working Flow of the lwB4

Hosts connected to the customer's network behind the lwB4 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, if necessary. 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. Fragmentation Behaviour

For TCP and UDP traffic the NAPT44 implemented in the lwB4 MUST conform with the behaviour and best current practices documented in [RFC4787], [RFC5508], and [RFC5382]. If the lwB4 supports DCCP, then the requirements in [RFC5597] MUST be implemented.

The NAPT44 in the lwB4 MUST implement ICMP message handling behaviour conforming to the best current practice documented in [RFC5508]. If the lwB4 receives an ICMP error (for errors detected inside the IPv6 tunnel), the node relays the ICMP error message to the original source (the lwAFTR). This behaviour SHOULD be implemented conforming to the section 8 of [RFC2473].

If IPv4 hosts behind different lwB4s sharing the same IPv4 address send fragments to the same IPv4 destination host outside the Lightweight 4over6 domain, those hosts may use the same IPv4 fragmentation identifier, resulting in incorrect reassembly of the fragments at the destination host. Given that the IPv4 fragmentation identifier is a 16-bit field, it could be used similarly to port ranges: A lwB4 could rewrite the IPv4 fragmentation identifier to be within its allocated port-set, if the resulting fragment identifier space is large enough related to the rate fragments are sent. However, splitting the identifier space in this fashion would increase the probability of reassembly collision for all connections through the lwB4. See also Section 5.3.1 of [RFC6864].

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 [RFC2983])

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 from one lwB4 that need to be sent to another lwB4 associated with the same AFTR. 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: [RFC7341] 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. An lwB4 MAY implement [RFC7341] and [I-D.ietf-dhc-dynamic-shared-v4allocation] to retrieve a shared IPv4 address with a set of ports.

- 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 synchronized across the lwB4s, the lwAFTRs and the provisioning server.

To prevent interworking complexity, it is RECOMMENDED that an operator uses a single provisioning mechanism / protocol for their implementation. In the event that more than one provisioning mechanism / protocol needs to be used (for example during a migration to a new provisioning mechanism), the operator SHOULD ensure that each provisioning mechanism has a discrete set of resources (e.g.

IPv4 address/PSID pools and lwAFTR tunnel addresses and binding tables).

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.

Otherwise the lwAFTR MUST discard all inbound ICMPv4 messages.

The ICMP policy SHOULD be configurable.

8.2. ICMPv4 Processing by the lwB4

The lwB4 MUST 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.

The lwAFTR MUST rate limit ICMPv6 error messages (see Section 5.1) to defend against DoS attacks generated by an abuse user.

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

This document describes a number of different protocols which may be used for the provisioning of lw4o6. In each case, the security considerations relevant to the provisioning protocol are also relevant to the provisioning of lw4o6 using that protocol. Lw4o6 does not add any additional provisioning protocol specific security considerations.

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

Axel Clauberg

Deutsche Telekom AG

CTO-ATI

Landgrabenweg 151

Bonn, 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., Farrer, I., Perreault, S., 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-10 (work in progress), November 2014.

[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.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC5382] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.

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

[RFC5597] Denis-Courmont, R., "Network Address Translation (NAT) Behavioral Requirements for the Datagram Congestion Control Protocol", BCP 150, RFC 5597, September 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-dynamic-shared-v4allocation]
Cui, Y., Qiong, Q., Farrer, I., Lee, Y., Sun, Q., and M. Boucadair, "Dynamic Allocation of Shared IPv4 Addresses", draft-ietf-dhc-dynamic-shared-v4allocation-02 (work in progress), September 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-07 (work in progress), November 2014.

[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-11 (work in progress), October 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.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, October 2000.

[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.

[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.

[RFC6864] Touch, J., "Updated Specification of the IPv4 ID Field", RFC 6864, February 2013.

[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.

[RFC7341] Sun, Q., Cui, Y., Siodelski, M., Krishnan, S., and I. Farrer, "DHCPv4-over-DHCPv6 (DHCP 4o6) Transport", RFC 7341, August 2014.

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

Softwire WG
Internet-Draft
Intended status: Standards Track
Expires: September 10, 2015

T. Mrugalski
ISC
O. Troan
Cisco
I. Farrer
Deutsche Telekom AG
S. Perreault
Viagenie
W. Dec
Cisco
C. Bao
Tsinghua University
L. Yeh
CNNIC
X. Deng

March 09, 2015

DHCPv6 Options for configuration of Softwire Address and Port Mapped Clients
draft-ietf-softwire-map-dhcp-12

Abstract

This document specifies DHCPv6 options, termed Softwire46 options, for the provisioning of Softwire46 Customer Edge (CE) devices. Softwire46 is a collective term used to refer to architectures based on the notion of IPv4 Address+Port (A+P) for providing IPv4 connectivity across an IPv6 network.

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 September 10, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Conventions 3
3. Softwire46 Overview 3
4. Common Softwire46 DHCPv6 Options 4
4.1. S46 Rule Option 5
4.2. S46 BR Option 7
4.3. S46 DMR Option 7
4.4. S46 IPv4/IPv6 Address Binding Option 8
4.5. S46 Port Parameters Option 9
5. Softwire46 Containers 10
5.1. Softwire46 MAP-E Container Option 10
5.2. Softwire46 MAP-T Container Option 11
5.3. Softwire46 LightWeight 46 Container Option 11
6. Softwire46 Options Formatting 12
7. DHCPv6 Server Behavior 13
8. DHCPv6 Client Behavior 13
9. Security Considerations 14
10. IANA Considerations 14
11. Acknowledgements 15
12. References 15
12.1. Normative References 15
12.2. Informative References 15
Authors' Addresses 16

1. Introduction

A number of architectural solution proposals discussed in the IETF Softwire Working Group use Address and Port (A+P) as their technology base for providing IPv4 connectivity to end users using CE devices across a Service Provider's IPv6 network, while allowing for shared or dedicated IPv4 addressing of CEs.

An example is Mapping of Address and Port (MAP) defined in [I-D.ietf-softwire-map]. The MAP solution consists of one or more MAP Border Relay (BR) routers, responsible for stateless forwarding between a MAP IPv6 domain and an IPv4 network, and one or more MAP Customer Edge (CE) routers, responsible for forwarding between a user's IPv4 network and the MAP IPv6 network domain. Collectively, the MAP CE and BR form a domain when configured with common service parameters. This characteristic is common to all of the Softwire46 mechanisms.

To function in such a domain, a CE needs to be provisioned with the appropriate A+P service parameters for that domain. These consist primarily of the CE's IPv4 address and transport layer port-range(s). Furthermore, the IPv6 transport mode (i.e. encapsulation or translation) needs to be specified. Provisioning of other IPv4 configuration information not derived directly from the A+P service parameters is not covered in this document. It is expected that provisioning of other IPv4 configuration will continue to use DHCPv4 [RFC2131].

This memo specifies a set of DHCPv6 [RFC3315] options to provision Softwire46 information to CE routers. Although the focus is to deliver IPv4 service to an end-user network (such as a residential home network), it can equally be applied to an individual host acting as a CE. Configuration of the BR is out of scope of this document.

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 RFC 2119 [RFC2119].

3. Softwire46 Overview

This document describes a set of common DHCPv6 options for configuring the MAP-E [I-D.ietf-softwire-map], MAP-T [I-D.ietf-softwire-map-t] and Lightweight 4over6 [I-D.ietf-softwire-lw4over6] mechanisms. For definition of the terminology used in this document please see the relevant terminology sections in the above references.

MAP-E, MAP-T and Lightweight 4over6 are essentially providing the same functionality: IPv4 service to a CE router over an IPv6 only access network. MAP-E and MAP-T may embed parts of the IPv4 address in IPv6 prefixes, thereby supporting many clients with a fixed set of mapping rules and mesh mode (direct CE to CE communication). MAP-E and MAP-T CEs may also be provisioned in hub and spoke mode, and in 1:1 mode (with no embedded address bits). The difference between

MAP-E and MAP-T is that they use different means to connect to the IPv6 domain. MAP-E uses [RFC2473] IPv4 over IPv6 tunnelling, while MAP-T uses NAT64 [RFC6145] based translation. Lightweight 4over6 is a hub and spoke IPv4 over IPv6 tunneling mechanism, with complete independence of IPv4 and IPv6 addressing (zero embedded address bits).

The DHCP options described here tie the provisioning parameters, and hence the IPv4 service itself, to the End-user IPv6 prefix lifetime. The validity of a Softwire46's IPv4 address, prefix or shared IPv4 address, port set and any authorization and accounting are tied to the lifetime of its associated End-user IPv6 prefix.

To support more than one mechanism at a time and to allow for a possibility of transition between them, the DHCPv6 Option Request Option [RFC3315] is used. Each mechanism has a corresponding DHCPv6 container option. A DHCPv6 client can request a particular mechanism by including the option code for a particular container option in its ORO option. The provisioning parameters for that mechanism are expressed by embedding the common format options within the respective container option.

This approach implies that all of the provisioning options MUST appear only within the container options. The client MUST NOT request any of the provisioning options directly within an ORO. MAP-DHCP clients that receive provisioning options that are not encapsulated in container options MUST silently ignore these options. DHCP server administrators are advised to ensure that DHCP servers are configured to send these options in the proper encapsulation.

The document is organized with the common sub-options described first, followed by the three container options. Some sub-options are mandatory in some containers, some are optional and some are not permitted at all. This is shown in Table 1.

4. Common Softwire46 DHCPv6 Options

The DHCPv6 protocol is used for Softwire46 CE provisioning following regular DHCPv6 notions, with the CE assuming the role of a DHCPv6 client, and the DHCPv6 server providing options following DHCPv6 server side policies. The format and usage of the options are defined in the following sub-sections.

Each CE needs to be provisioned with enough information to calculate its IPv4 address, IPv4 prefix or shared IPv4 address. MAP-E and MAP-T use the OPTION_S46_RULE, while Lightweight 4over6 uses the OPTION_S46_V4V6BIND option. A CE that needs to communicate outside of the A+P domain also needs the address or prefix of the BR. MAP-E

and Lightweight 4over6 use the OPTION_S46_BR option to communicate the IPv6 address of the BR. MAP-T forms an IPv6 destination address by embedding an IPv4 destination address into the BR's IPv6 prefix conveyed via the OPTION_S46_DMR option. Optionally, all mechanisms can include OPTION_S46_PORTPARAMS to specify parameters and port sets for the port range algorithm.

Softwire46 options use addresses rather than FQDNs. For rationale behind this design choice, see Section 8 of [RFC7227].

4.1. S46 Rule Option

Figure 1 shows the format of the S46 Rule option (OPTION_S46_RULE) used for conveying the Basic Mapping Rule (BMR) and Forwarding Mapping Rule (FMR).

This option follows behavior described in Sections 17.1.1 and 18.1.1 of [RFC3315]. Clients can insert those options with specific values as hints for the server. Depending on the server configuration and policy, it may accept or ignore the hints. Client MUST be able to process received values that are different than the hints it sent earlier.

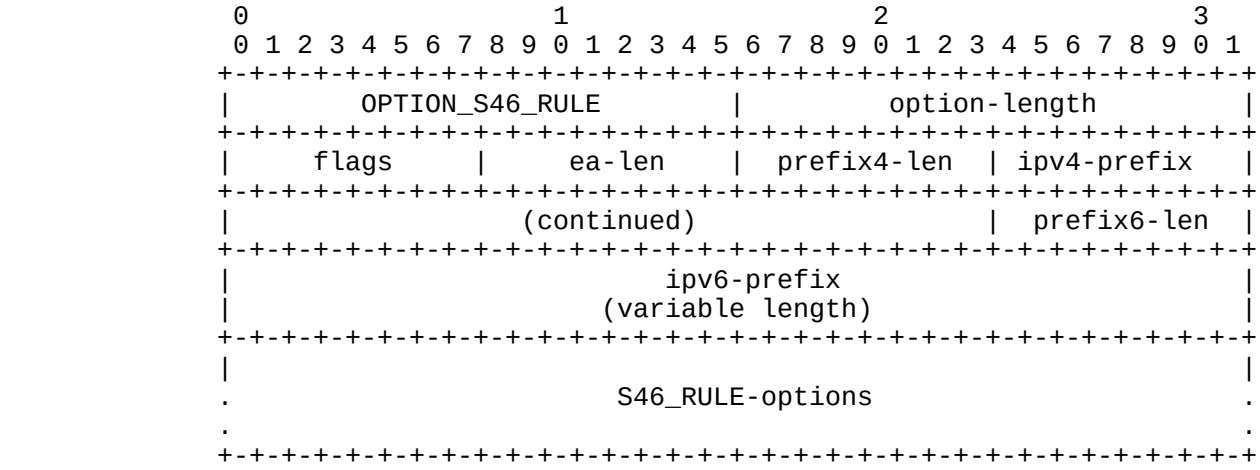


Figure 1: S46 Rule Option

- option-code: OPTION_S46_RULE (TBD1)
- option-length: length of the option, excluding option-code and option-length fields, including length of all encapsulated options, expressed in bytes.

- o flags: 8 bits long field carrying flags applicable to the rule. The meaning of specific bits are explained in Figure 2.

- o ea-len: 8 bits long field that specifies the Embedded-Address (EA) bit length. Allowed values range from 0 to 48.

- o prefix4-len: 8 bits long field expressing the prefix length of the IPv4 prefix specified in the rule-ipv4-prefix field. Valid values 0 to 32.

- o ipv4-prefix: a fixed length 32 bit field that specifies the IPv4 prefix for the S46 rule. The bits in the prefix after prefix4-len number of bits are reserved and MUST be initialized to zero by the sender and ignored by the receiver.

- o prefix6-len: 8 bits long field expressing the length of the IPv6 prefix specified in the rule-ipv6-prefix field.

- o ipv6-prefix: a variable length field that specifies the IPv6 domain prefix for the S46 rule. The field is padded on the right with zero bits up to the nearest octet boundary when prefix6-len is not evenly divisible by 8.

- o S46_RULE-options: a variable field that may contain zero or more options that specify additional parameters for this S46 rule. This document specifies one such option, OPTION_S46_PORTPARAMS.

The Format of the S46 Rule Flags field is:

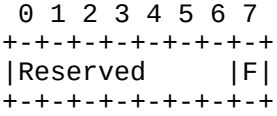


Figure 2: S46 Rule Flags

- o Reserved: 7-bits reserved for future use as flags.

- o F-Flag: 1 bit field that specifies whether the rule is to be used for forwarding (FMR). If set, this rule is used as a FMR, if not set this rule is a BMR only and MUST NOT be used for forwarding. Note: A BMR can also be used as an FMR for forwarding if the F-flag is set. The BMR rule is determined by a longest-prefix match of the Rule-IPv6-prefix against the End-User IPv6 prefix(es).

It is expected that in a typical mesh deployment scenario, there will be a single BMR, which could also be designated as an FMR using the F-Flag.

4.2. S46 BR Option

The S46 BR Option (OPTION_S46_BR) is used to convey the IPv6 address of the Border Relay. Figure 4 shows the format of the OPTION_S46_BR option.

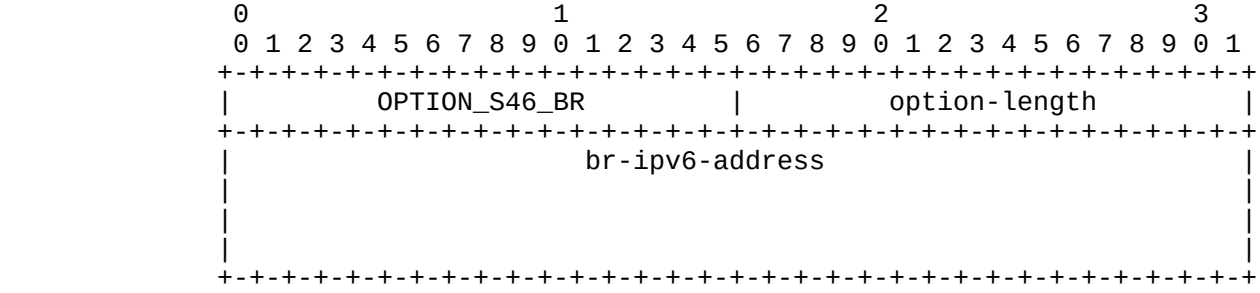


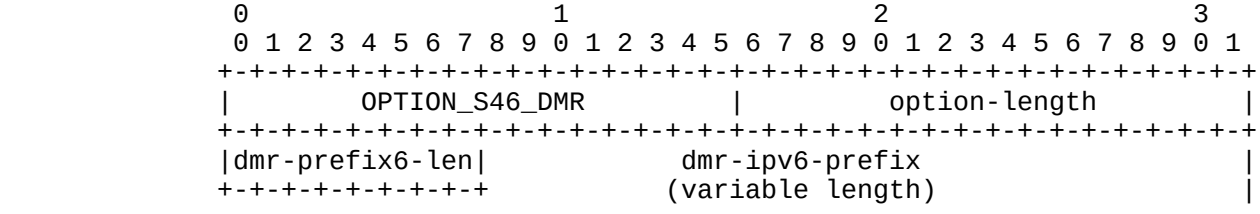
Figure 3: S46 BR Option

- option-code: OPTION_S46_BR (TBD2)
- option-length: 16
- br-ipv6-address: a fixed length field of 16 octets that specifies the IPv6 address for the S46 BR.

BR redundancy can be implemented by using an anycast address for the BR IPv6 address. Multiple OPTION_S46_BR options MAY be included in the container; this document does not further explore the use of multiple BR IPv6 addresses.

4.3. S46 DMR Option

The S46 DMR Option (OPTION_S46_DMR) is used to convey values for the Default Mapping Rule (DMR). Figure 4 shows the format of the OPTION_S46_DMR option used for conveying a DMR.



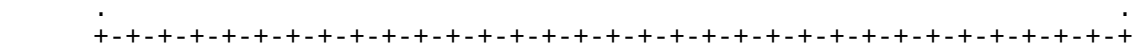


Figure 4: S46 DMR Option

- option-code: OPTION_S46_DMR (TBD3)
- option-length: 1 + length of dmr-ipv6-prefix specified in bytes.
- dmr-prefix6-len: 8 bits long field expressing the bit mask length of the IPv6 prefix specified in the dmr-ipv6-prefix field.
- dmr-ipv6-prefix: a variable length field specifying the IPv6 prefix or address for the BR. This field is right padded with zeros to the nearest octet boundary when dmr-prefix6-len is not divisible by 8.

4.4. S46 IPv4/IPv6 Address Binding Option

The IPv4 address Option (OPTION_S46_V4V6BIND) MAY be used to specify the full or shared IPv4 address of the CE. The IPv6 prefix field is used by the CE to identify the correct prefix to use for the tunnel source.

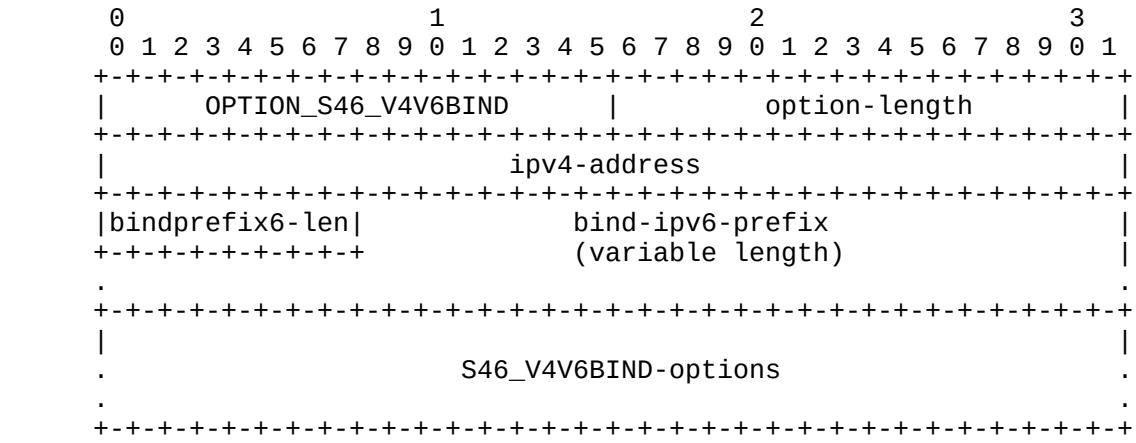


Figure 5: S46 IPv4/IPv6 Address Binding Option

- option-code: OPTION_S46_V4V6BIND (TBD4)
- option-length: length of the option, excluding option-code and option-length fields, including length of all encapsulated options, expressed in bytes.

- ipv4-address: A fixed field of 4 octets specifying an IPv4 address.

- bindprefix6-len: 8 bits long field expressing the bit mask length of the IPv6 prefix specified in the bind-ipv6-prefix field.

- bind-ipv6-prefix: a variable length field specifying the IPv6 prefix or address for the S46 CE. This field is right padded with zeros to the nearest octet boundary when bindprefix6-len is not divisible by 8.

- S46_V4V6BIND-options: a variable field that may contain zero or more options that specify additional parameters. This document specifies one such option, OPTION_S46_PORTPARAMS.

4.5. S46 Port Parameters Option

The Port Parameters Option (OPTION_S46_PORTPARAMS) specifies optional Port Set information that MAY be provided to CEs.

See [I-D.ietf-softwire-map], Section 5.1 for a description of MAP algorithm, explaining all of the parameters in detail.

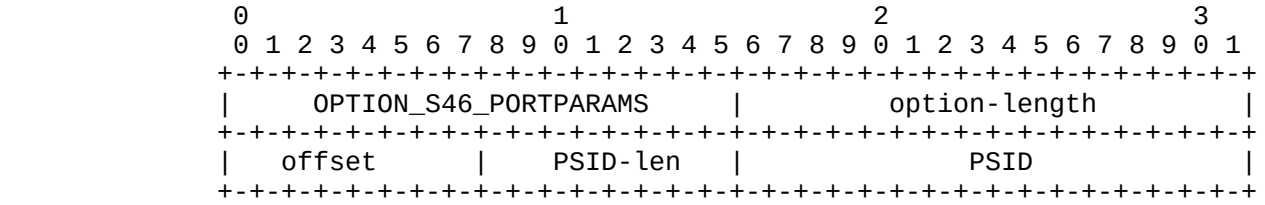


Figure 6: S46 Port Parameters Option

- option-code: OPTION_S46_PORTPARAMS (TBD5)

- option-length: 4

- offset: (PSID offset) 8 bits long field that specifies the numeric value for the S46 algorithm's excluded port range/offset bits (a-bits), as per section 5.1.1 of [I-D.ietf-softwire-map]. Allowed values are between 0 and 15. Default values for this field are specific to the softwire mechanism being implemented and are defined in the relevant specification document.

- PSID-len: Bit length value of the number of significant bits in the PSID field. (also known as 'k'). When set to 0, the PSID field is to be ignored. After the first 'a' bits, there are k bits in the port number representing the value of the Port Set

Identifier (PSID). Consequently, the address sharing ratio would be 2^k.

- PSID: Explicit 16-bit (unsigned word) PSID value. The PSID value algorithmically identifies a set of ports assigned to a CE. The first k bits on the left of this field contain the PSID value. The remaining (16-k) bits on the right are padding zeros.

When receiving the OPTION_S46_PORTPARAMS option with an explicit PSID, the client MUST use this explicit PSID in configuring its softwire interface. The OPTION_S46_PORTPARAMS option with an explicit PSID MUST be discarded if the S46 CE isn't configured with a full IPv4 address (e.g. IPv4 prefix).

The OPTION_S46_PORTPARAMS option with an explicit PSID MUST be discarded if the S46 CE isn't configured with a full IPv4 address (e.g. IPv4 prefix).

The OPTION_S46_PORTPARAMS option is contained within an OPTION_S46_RULE option or an OPTION_S46_V4V6BIND option.

5. Softwire46 Containers

5.1. Softwire46 MAP-E Container Option

The MAP-E Container Option (OPTION_S46_CONT_MAPE) specifies the container used to group all rules and optional port parameters for a specified domain.

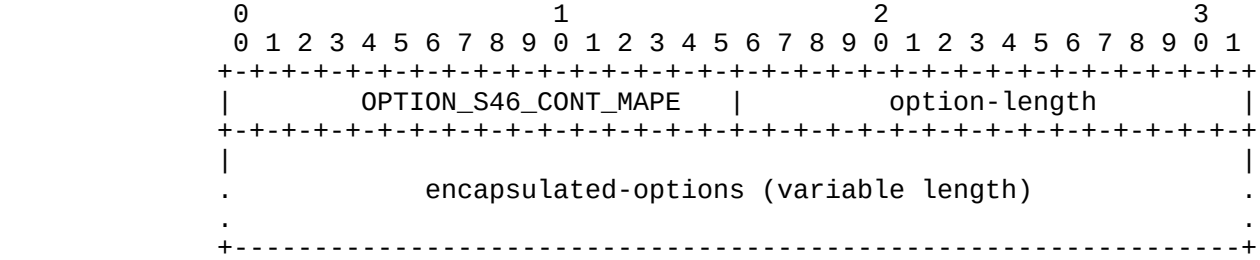


Figure 7: MAP-E Container Option

- option-code: OPTION_S46_CONT_MAPE (TBD6)
- option-length: Length of encapsulated options
- encapsulated-options: options associated with this Softwire46 MAP-E domain.

The encapsulated options field conveys options specific to the OPTION_S46_CONT_MAPE. Currently there are two sub-options specified, OPTION_S46_RULE and OPTION_S46_BR. There MUST be at least one OPTION_S46_RULE option and at least one OPTION_S46_BR option.

Other options applicable to a domain may be defined in the future. A DHCP message MAY include multiple OPTION_S46_CONT_MAPE options (representing multiple domains).

5.2. Softwire46 MAP-T Container Option

The MAP-T Container option (OPTION_S46_CONT_MAPT) specifies the container used to group all rules and optional port parameters for a specified domain.

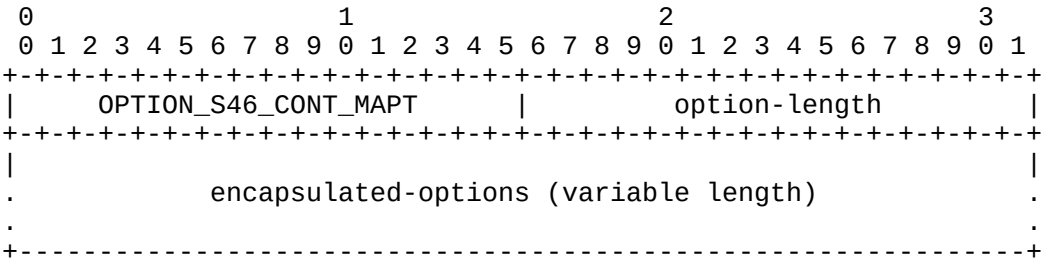


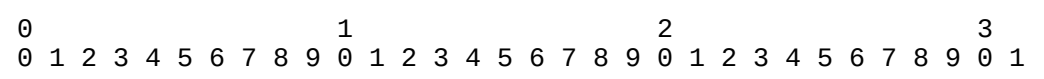
Figure 8: MAP-E Container Option

- option-code: OPTION_S46_CONT_MAPT (TBD7)
- option-length: Length of encapsulated options
- encapsulated-options: options associated with this Softwire46 MAP-T domain.

The encapsulated options field conveys options specific to the OPTION_S46_CONT_MAPT option. Currently there are two options specified, the OPTION_S46_RULE and OPTION_S46_DMR options. There MUST be at least one OPTION_S46_RULE option and exactly one OPTION_S46_DMR option.

5.3. Softwire46 LightWeight 46 Container Option

The LW46 Container option (OPTION_S46_CONT_LW) specifies the container used to group all rules and optional port parameters for a specified domain.



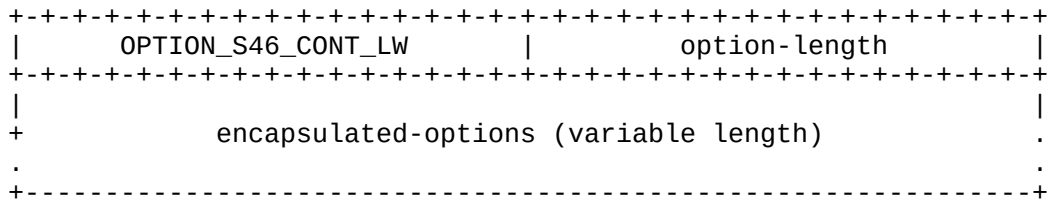


Figure 9: LW46 Container Option

- option-code: OPTION_S46_CONT_LW (TBD8)
- option-length: Length of encapsulated options
- encapsulated-options: options associated with this Softwire46 domain.

The encapsulated options field conveys options specific to the OPTION_S46_CONT_LW option. Currently there are two options specified, OPTION_S46_V4V6BIND and OPTION_S46_BR. There MUST be at most one OPTION_S46_V4V6BIND option and at least one OPTION_S46_BR option.

6. Softwire46 Options Formatting

The below table shows which sub-options are mandatory, optional or not permitted for each defined container option.

Option	MAP-E	MAP-T	Lightweight 4over6
OPTION_S46_RULE	M	M	N/A
OPTION_S46_BR	M	N/A	M
OPTION_S46_PORTPARAMS	O	O	O
OPTION_S46_DMR	N/A	M	N/A
OPTION_S46_V4V6BIND	N/A	N/A	O

M - Mandatory, O - Optional, N/A - Not Applicable

Table 1: Option to Container Mappings

MAP-DHCP clients that receive container options that violate any of the above rules MUST silently ignore such container options.

7. DHCPv6 Server Behavior

[RFC3315] Section 17.2.2 describes how a DHCPv6 client and server negotiate configuration values using the ORO. As a convenience to the reader, we mention here that by default, a server will not reply with a Softwire46 Container Option if the client has not explicitly enumerated one in its Option Request Option.

A CE router may support several (or all) of the mechanisms mentioned here. In the case where a client requests multiple mechanisms in its ORO option, the server will reply with the corresponding Softwire46 Container options for which it has configuration information.

8. DHCPv6 Client Behavior

An S46 CE acting as DHCPv6 client will request S46 configuration parameters from the DHCPv6 server located in the IPv6 network. Such a client MUST request the S46 Container option(s) that it is configured for in its ORO in SOLICIT, REQUEST, RENEW, REBIND and INFORMATION-REQUEST messages.

When processing received S46 container options the following behaviour is expected:

- A client MUST support processing multiple received OPTION_S46_RULE options in a container OPTION_S46_CONT_MAPE or OPTION_S46_CONT_MAPT option

- A client receiving an unsupported S46 option, or an invalid parameter value SHOULD discard that S46 Container option and log the event.

The behavior of a client supporting multiple Softwire46 mechanisms, is out of scope of this document. [I-D.ietf-softwire-unified-cpe] describes client behaviour for the prioritization and handling of multiple mechanisms simultaneously.

Note that system implementing CE functionality may have multiple network interfaces, and these interfaces may be configured differently; some may be connected to networks using a Softwire46 mechanism, and some may be connected to networks that are using normal dual stack or other means. The CE should approach this specification on an interface-by-interface basis. For example, if the CE system is MAP-E capable and is attached to multiple networks that provide the OPTION_S46_CONT_MAPE option, then the CE MUST configure MAP-E for each interface separately.

Failure modes are out of scope for this document. Failure recovery mechanisms may be defined in the future. See Section 5 of [I-D.ietf-softwire-map] for discussion on valid MAP rule combinations. See Section 11 of [RFC7227], Sections 18.1.3, 18.1.4 and 19.1 of [RFC3315] for parameters update mechanisms in DHCPv6 that can be leveraged to update configuration after a failure.

9. Security Considerations

Section 23 of [RFC3315] discusses DHCPv6-related security issues.

As with all DHCPv6-derived configuration state, it is possible that configuration is actually being delivered by a third party (Man In The Middle). As such, there is no basis on which access over MAP or lw4o6 can be trusted. Therefore, softwires should not bypass any security mechanisms such as IP firewalls.

In IPv6-only networks that lack any IPv4 firewalls, a device supporting MAP could be tricked into enabling its IPv4 stack and direct IPv4 traffic to the attacker, thus exposing itself to previously infeasible IPv4 attack vectors.

Section 11 of [I-D.ietf-softwire-map] discusses security issues of the MAP mechanism.

Readers concerned with security of MAP provisioning over DHCPv6 are encouraged to read [I-D.ietf-dhc-sedhcpv6].

10. IANA Considerations

IANA is kindly requested to allocate the following DHCPv6 option codes:

TBD1 for OPTION_S46_RULE

TBD2 for OPTION_S46_BR

TBD3 for OPTION_S46_DMR

TBD4 for OPTION_S46_V4V6BIND

TBD5 for OPTION_S46_PORTPARAMS

TBD6 for OPTION_S46_CONT_MAPE

TBD7 for OPTION_S46_CONT_MAPT

TBD8 for OPTION_S46_CONT_LW

All values should be added to the DHCPv6 option code space defined in Section 24.3 of [RFC3315].

11. Acknowledgements

This document was created as a product of a MAP design team. Following people were members of that team: Congxiao Bao, Mohamed Boucadair, Gang Chen, Maoke Chen, Wojciech Dec, Xiaohong Deng, Jouni Korhonen, Xing Li, Satoru Matsushima, Tomasz Mrugalski, Tetsuya Murakami, Jacni Qin, Necj Scoberne, Qiong Sun, Tina Tsou, Dan Wing, Leaf Yeh and Jan Zorz.

The authors would like to thank Bernie Volz and Tom Taylor for their insightful comments and suggestions.

12. References

12.1. Normative References

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

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

12.2. Informative References

[I-D.ietf-dhc-sedhcpv6]
Jiang, S., Shen, S., Zhang, D., and T. Jinmei, "Secure DHCPv6 with Public Key", draft-ietf-dhc-sedhcpv6-03 (work in progress), June 2014.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Qiong, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-03 (work in progress), November 2013.

[I-D.ietf-softwire-map-t]
Li, X., Bao, C., Dec, W., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", draft-ietf-softwire-map-t-04 (work in progress), September 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-08 (work in progress), August 2013.

[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.

[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.

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

[RFC7227] Hankins, D., Mrugalski, T., Siodelski, M., Jiang, S., and S. Krishnan, "Guidelines for Creating New DHCPv6 Options", BCP 187, RFC 7227, May 2014.

Authors' Addresses

Tomasz Mrugalski
Internet Systems Consortium, Inc.
950 Charter Street
Redwood City, CA 94063
USA

Phone: +1 650 423 1345
Email: tomasz.mrugalski@gmail.com
URI: http://www.isc.org/

Ole Troan
Cisco Systems, Inc.
Philip Pedersens vei 1
Lysaker 1366
Norway

Email: ot@cisco.com

Ian Farrer
Deutsche Telekom AG
CTO-ATI, Landgrabenweg 151
Bonn, NRW 53227
Germany

Email: ian.farrer@telekom.de

Simon Perreault
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Phone: +1 418 656 9254
Email: simon.perreault@viagenie.ca

Wojciech Dec
Cisco Systems, Inc.
The Netherlands

Email: wdec@cisco.com
URI: http://cisco.com

Congxiao Bao
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing 100084
CN

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

Leaf Y. Yeh
CNNIC
4, South 4th Street, Zhong_Guan_Cun
Beijing 100190
P. R. China

Email: leaf.yeh.sdo@gmail.com

Xiaohong Deng
6 Floor, C Block, DaCheng International Center Chaoyang District
Beijing 100124
China

Phone: +61 3858 3128
Email: dxhbupt@gmail.com

Softwires Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 5, 2015

X. Li
C. Bao
CERNET Center/Tsinghua University
W. Dec, Ed.
O. Troan
Cisco Systems
S. Matsushima
SoftBank Telecom
T. Murakami
IP Infusion

December 2, 2014

Mapping of Address and Port using Translation (MAP-T)
draft-ietf-softwire-map-t-08

Abstract

This document specifies the "Mapping of Address and Port" stateless IPv6-IPv4 Network Address Translation (NAT64) based solution architecture for providing shared or non-shared IPv4 address connectivity to and across an IPv6 network.

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 June 5, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Conventions 3
3. Terminology 3
4. Architecture 5
5. Mapping Rules 7
5.1. Destinations outside the MAP domain 7
6. The IPv6 Interface Identifier 7
7. MAP-T Configuration 8
7.1. MAP CE 8
7.2. MAP BR 9
8. MAP-T Packet Forwarding 9
8.1. IPv4 to IPv6 at the CE 9
8.2. IPv6 to IPv4 at the CE 10
8.3. IPv6 to IPv4 at the BR 11
8.4. IPv4 to IPv6 at the BR 11
9. ICMP Handling 11
10. Fragmentation and Path MTU Discovery 12
10.1. Fragmentation in the MAP domain 12
10.2. Receiving IPv4 Fragments on the MAP domain borders . . . 12
10.3. Sending IPv4 fragments to the outside 13
11. NAT44 Considerations 13
12. Usage Considerations 13
12.1. EA-bit length 0 13
12.2. Mesh and Hub and spoke modes 13
12.3. Communication with IPv6 servers in the MAP-T domain . . 14
12.4. Compatibility with other NAT64 solutions 14
13. IANA Considerations 14
14. Security Considerations 14
15. Contributors 15
16. Acknowledgements 16
17. References 16
17.1. Normative References 16
17.2. Informative References 16
Appendix A. Examples of MAP-T translation 19
Appendix B. Port mapping algorithm 22
Authors' Addresses 22

1. Introduction

Experiences from initial service provider IPv6 network deployments, such as [RFC6219], indicate that successful transition to IPv6 can happen while supporting legacy IPv4 users without a full end-to-end dual IP stack deployment. However, due to public IPv4 address exhaustion this requires an IPv6 technology that supports IPv4 users utilizing shared IPv4 addressing, while also allowing the network operator to optimize their operations around IPv6 network practices. The use of double NAT64 translation based solutions is an optimal way to address these requirements, especially in combination with stateless translation techniques that minimize operational challenges outlined in [I-D.ietf-softwire-stateless-4v6-motivation].

The Mapping of Address and Port - Translation (MAP-T) architecture specified in this draft is such a double stateless NAT64 based solution. It builds on existing stateless NAT64 techniques specified in [RFC6145], along with the stateless algorithmic address & transport layer port mapping scheme defined in MAP-E [I-D.ietf-softwire-map]. The MAP-T solution differs from MAP-E in the use of IPv4-IPv6 translation, rather than encapsulation, as the form of IPv6 domain transport. The translation mode is considered advantageous in scenarios where the encapsulation overhead, or IPv6 operational practices (e.g. Use of IPv6 only servers, or reliance on IPv6 + protocol headers for traffic classification) rule out encapsulation. These scenarios are presented in [I-D.maglione-softwire-map-t-scenarios]

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 RFC 2119 [RFC2119].

3. Terminology

MAP-T
: Mapping of Address and Port by means of address Translation.

MAP Customer Edge (CE)
: A device functioning as a Customer Edge (CE) router in a MAP deployment. A typical MAP CE adopting MAP rules will serve a residential site with one WAN side IPv6 addressed interface, and one or more LAN side interfaces addressed using private IPv4 addressing.

MAP Border Relay (BR):	A MAP enabled router managed by the service provider at the edge of a MAP domain. A Border Relay (BR) router has at least an IPv6-enabled interface and an IPv4 interface connected to the native IPv4 network. A MAP BR may also be referred to simply as a "BR" within the context of MAP.
MAP domain:	One or more MAP CEs and BRs connected by means of an IPv6 network and sharing a common set of MAP Rules. A service provider may deploy a single MAP domain, or may utilize multiple MAP domains.
MAP Rule:	A set of parameters describing the mapping between an IPv4 prefix, IPv4 address or shared IPv4 address and an IPv6 prefix or address. Each MAP domain uses a different mapping rule set.
MAP Rule set:	A Rule set is composed out of all the MAP Rules communicated to a device, that are intended for determining the devices' IP+port mapping and forwarding operations. The MAP Rule set is interchangeably referred to in this document as a MAP Rule table or simply Rule table. Two specific types of rules, Basic Mapping Rule (BMR) and Forward Mapping Rule (FMR), are defined in Section 5 of [I-D.ietf-softwire-map]. The Default Mapping Rule (DMR) is defined in this document.
MAP Rule table:	See MAP Rule set.
MAP node:	A device that implements MAP.
Port-set:	Each node has a separate part of the transport layer port space; denoted as a port-set.
Port-set ID (PSID):	Algorithmically identifies a set of ports exclusively assigned to the CE.
Shared IPv4 address:	An IPv4 address that is shared among multiple CEs. Only ports that belong to the assigned port-set can be used for communication. Also known as a Port-Restricted IPv4 address.

End-user IPv6 prefix: The IPv6 prefix assigned to an End-user CE by other means than MAP itself. E.g. Provisioned using DHCPv6 PD [RFC3633], assigned via SLAAC [RFC4862], or configured manually. It is unique for each CE.

MAP IPv6 address: The IPv6 address used to reach the MAP function of a CE from other CEs and from BRs.

Rule IPv6 prefix: An IPv6 prefix assigned by a Service Provider for a MAP rule.

Rule IPv4 prefix: An IPv4 prefix assigned by a Service Provider for a MAP rule.

Embedded Address (EA) bits: The IPv4 EA-bits in the IPv6 address identify an IPv4 prefix/address (or part thereof) or a shared IPv4 address (or part thereof) and a port-set identifier.

4. Architecture

Figure 1 depicts the overall MAP-T architecture, which sees any number of privately addressed IPv4 users (N and M) connected by means of MAP-T CEs to an IPv6 network that is equipped with one or more MAP-T BR. CEs and BRs that share MAP configuration parameters, referred to as MAP rules, form a MAP-T Domain.

Functionally the MAP-T CE and BR utilize and extend some well established technology building blocks to allow the IPv4 users to correspond with nodes on the Public IPv4 network, or IPv6 network as follows:

- A (NAT44) NAPT [RFC2663] function on a MAP CE is extended with support for restricting the allowable TCP/UDP ports for a given IPv4 address. The IPv4 address and port range used are determined by the MAP provisioning process and identical to MAP-E [I-D.ietf-softwire-map].

- A stateless NAT64 function [RFC6145] is extended to allow stateless mapping of IPv4 and transport layer port ranges to IPv6 address space.

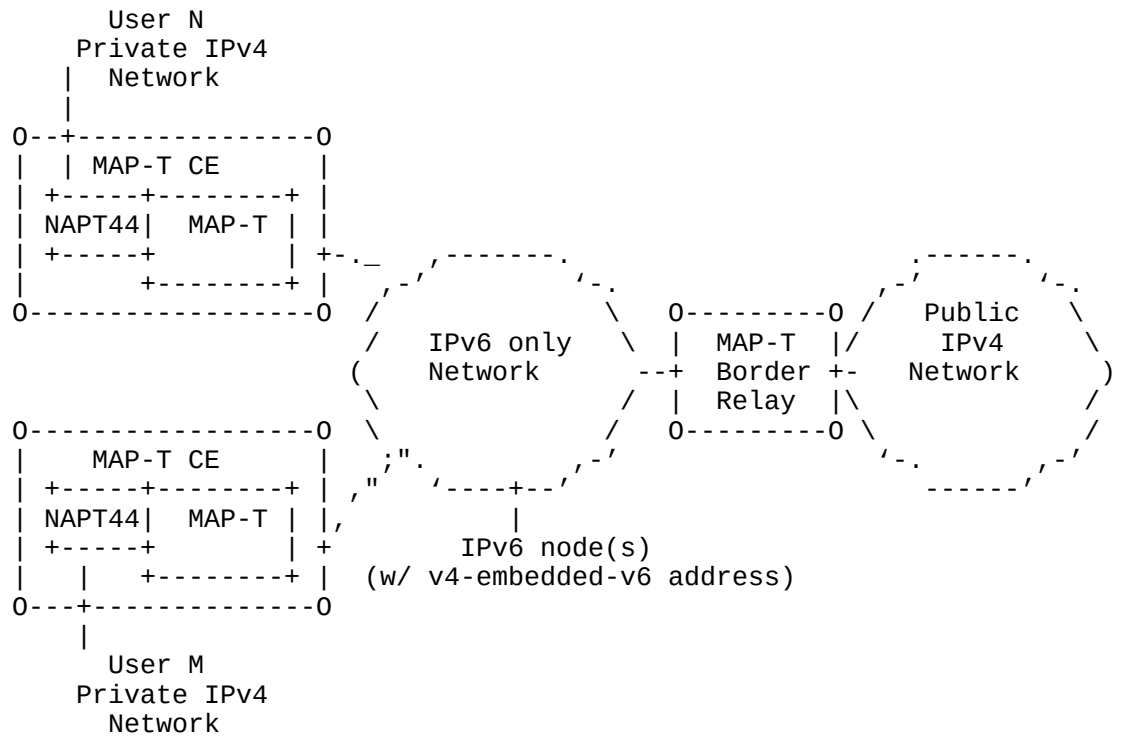


Figure 1: MAP-T Architecture

Each MAP-T CE is assigned with a regular IPv6 prefix from the operator's IPv6 network. This, in conjunction with MAP domain configuration settings and the use of the MAP procedures allows the computation of a MAP IPv6 address and a corresponding IPv4 address. To allow for IPv4 address sharing, the CE may also have be configured with a TCP/UDP port-range that is identified by means of a MAP Port Set Identifier (PSID) value. Each CE is responsible for forwarding traffic between a given user's private IPv4 address space and the MAP domain's IPv6 address space. The IPv4-IPv6 adaptation uses stateless NAT64, in conjunction with the MAP algorithm for address computation.

The MAP-T BR connects one or more MAP-T domains to external IPv4 networks using stateless NAT64 as extended by the MAP-T behaviour described in this document.

In contrast to MAP-E, NAT64 technology is used in the architecture for two purposes. Firstly, it is intended to diminish encapsulation overhead and allow IPv4 and IPv6 traffic to be treated as similarly as possible. Secondly, it is intended to allow IPv4-only nodes to

correspond directly with IPv6 nodes in the MAP-T domain that have IPv4 embedded IPv6 addresses as per [RFC6052]).

The MAP-T architecture is based on the following key properties i) algorithmic IPv4-IPv6 address mapping codified as MAP Rules covered in Section 5 ii) A MAP IPv6 address identifier, described in Section 6 iii) MAP-T IPv4-IPv6 forwarding behavior described in Section 8.

5. Mapping Rules

The MAP-T algorithmic mapping rules are identical to those in Section 5 of the MAP-E specification [I-D.ietf-softwire-map], with the following exception. The forwarding of traffic to and from IPv4 destinations outside a MAP-T domain is to be performed as described here under, instead of Section 5.4 of the MAP-E specification.

5.1. Destinations outside the MAP domain

IPv4 traffic sent by MAP nodes that are all within one MAP domain is translated to IPv6, with the sender’s MAP IPv6 address, derived via the Basic Mapping Rule (BMR), as the IPv6 source address and the recipient’s MAP IPv6 address, derived via the Forward Mapping Rule (FMR), as the IPv6 destination address.

IPv4 addressed destinations outside of the MAP domain are represented by means of IPv4-Embedded IPv6 address as per [RFC6052], using the BR’s IPv6 prefix. For a CE sending traffic to any such destination, the source address of the IPv6 packet will be that of the CE’s MAP IPv6 address, and the destination IPv6 address will be the destination IPv4-embedded-IPv6 address. This address mapping is termed as following the MAP-T Default Mapping Rule (DMR) and is defined in terms of the IPv6 prefix advertised by one or more BRs, which provide external connectivity. A typical MAP-T CE will install an IPv4 default route using this rule. A BR will use this rule when translating all outside IPv4 source addresses to the IPv6 MAP domain.

The DMR IPv6 prefix-length SHOULD be by default 64 bits long, and in any case MUST NOT exceed 96 bits. The mapping of the IPv4 destination behind the IPv6 prefix will by default follow the /64 rule as per [RFC6052]. Any trailing bits after the IPv4 address are set to 0x0.

6. The IPv6 Interface Identifier

The Interface identifier format of a MAP-T node is the same as described in section 6 of [I-D.ietf-softwire-map]. For convenience this is cited below:

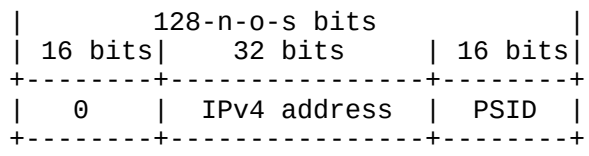


Figure 2

In the case of an IPv4 prefix, the IPv4 address field is right-padded with zeros up to 32 bits. The PSID is zero left-padded to create a 16 bit field. For an IPv4 prefix or a complete IPv4 address, the PSID field is zero.

If the End-user IPv6 prefix length is larger than 64, the most significant parts of the interface identifier is overwritten by the prefix.

7. MAP-T Configuration

For a given MAP domain, the BR and CE MUST be configured with the following MAP parameters. The values for these parameters are identical for all CEs and BRs within a given MAP-T domain.

- The Basic Mapping Rule and optionally the Forwarding Mapping Rules, including the Rule IPv6 prefix, Rule IPv4 prefix, and Length of Embedded Address bits
- Use of Hub and spoke mode or Mesh mode. (If all traffic should be sent to the BR, or if direct CE to CE correspondence should be supported).
- Use of IPv4-IPv6 Translation (MAP-T)
- The BR’s IPv6 prefix used in the DMR

7.1. MAP CE

For a given MAP domain, the MAP configuration parameters are the same across all CEs within that domain. These values may be conveyed and configured on the CEs using a variety of methods, including; DHCPv6, Broadband Forum’s "TR-69" Residential Gateway management interface, Netconf, or manual configuration. This document does not prescribe any of these methods, but recommends that a MAP CE SHOULD implement DHCPv6 options as per [I-D.ietf-softwire-map-dhcp]. Other configuration and management methods may use the data model described by this option for consistency and convenience of implementation on CEs that support multiple configuration methods.

Besides the MAP configuration parameters, a CE requires an IPv6 prefix to be assigned to the CE. This End-user IPv6 prefix is configured as part of obtaining IPv6 Internet access, and is acquired using standard IPv6 means applicable in the network where the CE is located.

The MAP provisioning parameters, and hence the IPv4 service itself, are tied to the End-user IPv6 prefix; thus, the MAP service is also tied to this in terms of authorization, accounting, etc.

A single MAP CE MAY be connected to more than one MAP domain, just as any router may have more than one IPv4-enabled service provider facing interface and more than one set of associated addresses assigned by DHCPv6. Each domain a given CE operates within would require its own set of MAP configuration elements and would generate its own IPv4 address. Each MAP domain requires a distinct End-user IPv6 prefix.

7.2. MAP BR

The MAP BR MUST be configured with the same MAP elements as the MAP CEs operating within the same domain.

For increased reliability and load balancing, the BR IPv6 prefix MAY be shared across a given MAP domain. As MAP is stateless, any BR may be used for forwarding to/from the domain at any time.

Since MAP uses provider address space, no specific IPv6 or IPv4 routes need to be advertised externally outside the service provider’s network for MAP to operate. However, the BR prefix needs to be advertised in the service provider’s IGP.

8. MAP-T Packet Forwarding

The end-to-end packet flow in MAP-T involves an IPv4 or IPv6 packet being forwarded by a CE of BR in one of two directions for each such case. This section presents a conceptual view of the operations involved in such forwarding.

8.1. IPv4 to IPv6 at the CE

A MAP-T CE receiving IPv4 packets SHOULD perform NAPT NAT44 processing, and create any necessary NAPT44 bindings. The source address and source port-range of packets resulting from the NAPT44 processing MUST correspond to the source IPv4 address and source transport port-range assigned to the CE by means of the MAP Basic Mapping Rule (BMR).

The IPv4 packet is subject to a longest IPv4 destination address + port match MAP rule selection, which then determines the parameters for the subsequent NAT64 operation. By default, all traffic is matched to the default mapping rule (DMR), and subject to the stateless NAT64 operation using the DMR parameters for NAT64 Section 5.1. Packets that are matched to (optional) Forward Mapping Rules (FMRs) are subject to the stateless NAT64 operation using the FMR parameters Section 5 for the MAP algorithm. In all cases the CE's MAP IPv6 address Section 6 is used as a source address.

A MAP-T CE MUST support a Default Mapping Rule and SHOULD support one or more Forward Mapping Rules.

8.2. IPv6 to IPv4 at the CE

A MAP-T CE receiving an IPv6 packet performs its regular IPv6 operations (filtering, pre-routing, etc). Only packets that are addressed to the CE's MAP-T IPv6 addresses, and with source addresses matching the IPv6 map-rule prefixes of a DMR or FMR, are processed by the MAP-T CE, with the DMR or FMR being selected based on a longest match. The CE MUST check that each MAP-T received packet's destination transport-layer destination port number is in the range allowed for by the CE's MAP BMR configuration. The CE MUST silently drop any non conforming packet and an appropriate counter incremented. When receiving a packet whose source IP address longest matches an FMR prefix, the CE MUST perform a check of consistency of the source address against the allowed values as per the derived allocated source port-range. If the source port number of a packet is found to be outside the allocated range, the CE MUST drop the packet and SHOULD respond with an ICMPv6 "Destination Unreachable, Source address failed ingress/egress policy" (Type 1, Code 5).

For each MAP-T processed packet, the CE's NAT64 function MUST compute an IPv4 source and destination addresses. The IPv4 destination address is computed by extracting relevant information from the IPv6 destination and the information stored in the BMR as per Section 5. The IPv4 source address is formed by classifying a packet's source as longest matching a DMR or FMR rule prefix, and then using the respective rule parameters for the NAT64 operation.

The resulting IPv4 packet is then forwarded to the CE's NAPT NAPT44 function, where the destination IPv4 address and port number MUST be mapped to their original value, before being forwarded according to the CE's regular IPv4 rules. When the NAPT44 function is not enabled, by virtue of MAP configuration, the traffic from the stateless NAT64 function is directly forwarded according to the CE's IPv4 rules.

8.3. IPv6 to IPv4 at the BR

A MAP-T BR receiving an IPv6 packet MUST select a matching MAP rule based on a longest address match of the packet's source address against the MAP Rules present on the BR. In combination with the Port-Set-Id derived from the packet's source IPv6 address, the selected MAP rule allows the BR to verify that the CE is using its allowed address and port range. Thus, the BR MUST perform a validation of the consistency of the source against the allowed values from the identified port-range. If the packet's source port number is found to be outside the range allowed, the BR MUST drop the packet and increment a counter to indicate the event. The BR SHOULD also respond with an ICMPv6 "Destination Unreachable, Source address failed ingress/egress policy" (Type 1, Code 5).

When constructing the IPv4 packet, the BR MUST derive the source and destination IPv4 addresses as per Section 5 of this document and translate the IPv6 to IPv4 headers as per [RFC6145]. The resulting IPv4 packet is then passed to regular IPv4 forwarding.

8.4. IPv4 to IPv6 at the BR

A MAP-T BR receiving IPv4 packets uses a longest match IPv4 + transport layer port lookup to identify the target MAP-T domain and select the FMR and DMR rules. The MAP-T BR MUST then compute and apply the IPv6 destination addresses from the IPv4 destination address and port as per the selected FMR. The MAP-T BR MUST also compute and apply the IPv6 source addresses from the IPv4 source address as per Section 5.1 (i.e. Using the IPv4 source and the BR's IPv6 prefix it forms an IPv6 embedded IPv4 address). Throughout the generic IPv4 to IPv6 header translation procedures following [RFC6145] apply. The resulting IPv6 packets are then passed to regular IPv6 forwarding.

Note that the operation of a BR when forwarding to/from MAP-T domains that are defined without IPv4 address sharing is the same as that of stateless NAT64 IPv4/IPv6 translation.

9. ICMP Handling

MAP-T CEs and BRs MUST follow ICMP/ICMPv6 translation as per [RFC6145], however additional behavior is also required due to the presence of NAPT44. Unlike TCP and UDP, which provide two transport protocol port fields to represent both source and destination, the ICMP/ICMPv6 [RFC0792], [RFC4443] Query message header has only one ID field which needs to be used to identify a sending IPv4 host. When receiving IPv4 ICMP messages, the MAP-T CE MUST rewrite the ID field to a port value derived from the CE's Port-Set-Id.

A MAP-T BR receiving an IPv4 ICMP packet , which contains an ID field that is bound for a shared address in the MAP-T domain, SHOULD use the ID value as a substitute for the destination port in determining the IPv6 destination address. In all other cases, the MAP-T BR MUST derive the destination IPv6 address by simply mapping the destination IPv4 address without additional port info.

10. Fragmentation and Path MTU Discovery

Due to the different sizes of the IPv4 and IPv6 header, handling the maximum packet size is relevant for the operation of any system connecting the two address families. There are three mechanisms to handle this issue: Path MTU discovery (PMTUD), fragmentation, and transport-layer negotiation such as the TCP Maximum Segment Size (MSS) option [RFC0897]. MAP can use all three mechanisms to deal with different cases.

Note: The NAT64 [RFC6145]mechanism is not lossless. When IPv4 originated communication traverses across a double NAT64 function (a.k.a. NAT464), any IPv4 originated ICMP-independent PathMTU Discovery, as specified in [RFC 4821], ceases to be entirely reliable. This is because the[RFC4821] defined DF=1/MF=1 combination, following a double NAT64 translation, results in DF=0/ MF=1.

10.1. Fragmentation in the MAP domain

Translating an IPv4 packet to carry it across the MAP domain will increase its size typically by 20 bytes. The MTU in the MAP domain should be well managed and the IPv6 MTU on the CE WAN side interface SHOULD be configured so that no fragmentation occurs within the boundary of the MAP domain.

Fragmentation in MAP-T domain SHOULD be handled as described in section 4 and 5 of [RFC6145].

10.2. Receiving IPv4 Fragments on the MAP domain borders

Forwarding of an IPv4 packet received from the outside of the MAP domain requires the IPv4 destination address and the transport protocol destination port. The transport protocol information is only available in the first fragment received. As described in section 5.3.3 of [RFC6346] a MAP node receiving an IPv4 fragmented packet from outside SHOULD reassemble the packet before sending the packet onto the MAP domain. If the first packet received contains the transport protocol information, it is possible to optimize this behavior by using a cache and forwarding the fragments unchanged. A

description of such a caching algorithm is outside the scope of this document.

10.3. Sending IPv4 fragments to the outside

Two IPv4 hosts behind two different MAP CE's with the same IPv4 address sending fragments to an IPv4 destination host outside the domain may happen to use the same IPv4 fragmentation identifier, resulting in incorrect reassembly of the fragments at the destination host. Given that the IPv4 fragmentation identifier is a 16 bit field, it can be used similarly to port ranges. Thus, a MAP CE SHOULD rewrite the IPv4 fragmentation identifier to a value equivalent to a port of its allocated port-set.

11. NAT44 Considerations

The NAT44 implemented in the MAP CE SHOULD conform with the behavior and best current practice documented in [RFC4787], [RFC5508], and [RFC5382]. In MAP address sharing mode (determined by the MAP domain /rule configuration parameters) the operation of the NAT44 MUST be restricted to the available port numbers derived via the basic mapping rule.

12. Usage Considerations

12.1. EA-bit length 0

The MAP solution supports use and configuration of domains where a BMR expresses an EA-bit length of 0. This results in independence between the IPv6 prefix assigned to the CE and the IPv4 address and/ or port-range used by MAP. The k-bits of PSID information may in this case be derived from the BMR.

The constraint imposed is that each such MAP domain be composed of just 1 MAP CE which has a predetermined IPv6 end-user prefix. The BR would be configured with an FMR for each such CPE, where the rule would uniquely associate the IPv4 address + optional PSID and the IPv6 prefix of that given CE.

12.2. Mesh and Hub and spoke modes

The hub and spoke mode of communication, whereby all traffic sent by a MAP-T CE is forwarded via a BR, and the mesh mode, whereby a CE is directly able to forward traffic to another CE, are governed by the activation of Forward Mapping Rule that cover the IPv4-prefix destination, and port-index range. By default, a MAP CE configured only with a BMR, as per this specification, will use it to configure its IPv4 parameters and IPv6 MAP address without enabling mesh mode.

12.3. Communication with IPv6 servers in the MAP-T domain

By default, MAP-T allows communication between both IPv4-only and any IPv6 enabled devices, as well as with native IPv6-only servers provided that the servers are configured with an IPv4-mapped IPv6 address. This address could be part of the IPv6 prefix used by the DMR in the MAP-T domain. Such IPv6 servers (e.g. An HTTP server, or a web content cache device) are thus able to serve both IPv6 users as well as IPv4-only users alike utilizing IPv6. Any such IPv6-only servers SHOULD have both A and AAAA records in DNS. DNS64 [RFC6147] become required only when IPv6 servers in the MAP-T domain are expected themselves to initiate communication to external IPv4-only hosts.

12.4. Compatibility with other NAT64 solutions

The MAP-T CEs NAT64 function is by default compatible for use with [RFC6146] stateful NAT64 devices that are placed in the operator's network. In such a case the MAP-T CE's DMR prefix is configured to correspond to the NAT64 device prefix. This in effect allows the use of MAP-T CEs in environments that need to perform statistical multiplexing of IPv4 addresses, while utilizing stateful NAT64 devices, and can take the role of a CLAT as defined in [RFC6877].

13. IANA Considerations

This specification does not require any IANA actions.

14. Security Considerations

Spoofing attacks: With consistency checks between IPv4 and IPv6 sources that are performed on IPv4/IPv6 packets received by MAP nodes, MAP does not introduce any new opportunity for spoofing attacks that would not already exist in IPv6.

Denial-of-service attacks: In MAP domains where IPv4 addresses are shared, the fact that IPv4 datagram reassembly may be necessary introduces an opportunity for DOS attacks. This is inherent to address sharing, and is common with other address sharing approaches such as DS-Lite and NAT64/DNS64. The best protection against such attacks is to accelerate IPv6 support in both clients and servers.

Routing-loop attacks: This attack may exist in some automatic tunneling scenarios are documented in [RFC6324]. They cannot exist with MAP because each BRs checks that the IPv6 source address of a received IPv6 packet is a CE address based on Forwarding Mapping Rule.

Attacks facilitated by restricted port-set: From hosts that are not subject to ingress filtering of [RFC2827], some attacks are possible by an attacker injecting spoofed packets during ongoing transport connections ([RFC4953], [RFC5961], [RFC6056]. The attacks depend on guessing which ports are currently used by target hosts, and using an unrestricted port-set is preferable, i.e. Using native IPv6 connections that are not subject to MAP port-range restrictions. To minimize this type of attacks when using a restricted port set, the MAP CE's NAT44 filtering behavior SHOULD be "Address-Dependent Filtering". Furthermore, the MAP CEs SHOULD use a DNS transport proxy function to handle DNS traffic, and source such traffic from IPv6 interfaces not assigned to MAP-T. Practicalities of these methods are discussed in Section 5.9 of [I-D.dec-stateless-4v6].

ICMP Flooding Given the necessity to process and translate ICMP and ICMPv6 messages by the BR and CE nodes, a foreseeable attack vector is that of a flood of such messages leading to a saturation of the node's ICMP computing resources. This attack vector is not specific to MAP, and its mitigation lies a combination of policing the rate of ICMP messages, policing the rate at which such messages can get processed by the MAP nodes, and of course identifying and blocking off the source(s) of such traffic.

[RFC6269] outlines general issues with IPv4 address sharing.

15. Contributors

The following individuals authored major contributions to this document, and made the document possible:

Chongfeng Xie (China Telecom) Room 708, No.118, Xizhimennei Street Beijing 100035 CN Phone: +86-10-58552116 Email: xiechf@ctbri.com.cn

Qiong Sun (China Telecom) Room 708, No.118, Xizhimennei Street Beijing 100035 CN Phone: +86-10-58552936 Email: sunqiong@ctbri.com.cn

Rajiv Asati (Cisco Systems) 7025-6 Kit Creek Road Research Triangle Park NC 27709 USA Email: rajiva@cisco.comc

Gang Chen (China Mobile) 53A,Xibianmennei Ave. Beijing 100053 P.R.China Email: chengang@chinamobile.com

Wentao Shang (CERNET Center/Tsinghua University) Room 225, Main Building, Tsinghua University Beijing 100084 CN Email: wentaoshang@gmail.com

Guoliang Han (CERNET Center/Tsinghua University) Room 225, Main Building, Tsinghua University Beijing 100084 CN Email: bupthgl@gmail.com

Yu Zhai CERNET Center/Tsinghua University Room 225, Main Building, Tsinghua University Beijing 100084 CN Email: jacky.zhai@gmail.com

16. Acknowledgements

This document is based on the ideas of many. In particular Remi Despres, who has tirelessly worked on generalized mechanisms for stateless address mapping.

The authors would also like to thank Mohamed Boucadair, Guillaume Gottard, Dan Wing, Jan Zorz, Nejc Scoberne, Tina Tsou, Gang Chen, Maoke Chen, Xiaohong Deng, Jouni Korhonen, Tomasz Mrugalski, Jacni Qin, Chunfa Sun, Qiong Sun, Leaf Yeh, Andrew Yourtchenko, Roberta Maglione and Hongyu Chen for their review and comments.

17. References

17.1. Normative References

[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-12 (work in progress), November 2014.

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

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

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

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

17.2. Informative References

[I-D.dec-stateless-4v6] Dec, W., Asati, R., and H. Deng, "Stateless 4Via6 Address Sharing", draft-dec-stateless-4v6-04 (work in progress), October 2011.

[I-D.ietf-softwire-map-dhcp]
Mrugalski, T., Troan, O., Farrer, I., Perreault, S., 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-11 (work in progress), November 2014.

[I-D.ietf-softwire-stateless-4v6-motivation]
Boucadair, M., Matsushima, S., Lee, Y., Bonness, O., Borges, I., and G. Chen, "Motivations for Carrier-side Stateless IPv4 over IPv6 Migration Solutions", draft-ietf-softwire-stateless-4v6-motivation-05 (work in progress), November 2012.

[I-D.maglione-softwire-map-t-scenarios]
Maglione, R., Dec, W., Leung, I., and E. Mallette, "Use cases for MAP-T", draft-maglione-softwire-map-t-scenarios-05 (work in progress), October 2014.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[RFC0897] Postel, J., "Domain name system implementation schedule", RFC 897, February 1984.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC4953] Touch, J., "Defending TCP Against Spoofing Attacks", RFC 4953, July 2007.

[RFC5382] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.

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

[RFC5961] Ramaiah, A., Stewart, R., and M. Dalal, "Improving TCP's Robustness to Blind In-Window Attacks", RFC 5961, August 2010.

[RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", BCP 156, RFC 6056, January 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.

[RFC6147] Bagnulo, M., Sullivan, A., Matthews, P., and I. van Beijnum, "DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, April 2011.

[RFC6219] Li, X., Bao, C., Chen, M., Zhang, H., and J. Wu, "The China Education and Research Network (CERNET) IVI Translation Design and Deployment for the IPv4/IPv6 Coexistence and Transition", RFC 6219, May 2011.

[RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, June 2011.

[RFC6324] Nakibly, G. and F. Templin, "Routing Loop Attack Using IPv6 Automatic Tunnels: Problem Statement and Proposed Mitigations", RFC 6324, August 2011.

[RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC 6877, April 2013.

Appendix A. Examples of MAP-T translation

Example 1 - Basic Mapping Rule:

Given the following MAP domain information and IPv6 end-user prefix assigned to a MAP CE:

```
End-user IPv6 prefix:  2001:db8:0012:3400::/56
Basic Mapping Rule:    {2001:db8:0000::/40 (Rule IPv6 prefix),
                       192.0.2.0/24 (Rule IPv4 prefix),
                       16 (Rule EA-bits length)}
PSID length:           (16 - (32 - 24) = 8. (Sharing ratio of 256)
PSID offset:           6 (default)
```

A MAP node (CE or BR) can via the BMR, or equivalent FMR, determine the IPv4 address and port-set as shown below:

```
EA bits offset:        40
IPv4 suffix bits (p):  Length of IPv4 address (32) - IPv4 prefix
                       length (24) = 8
IPv4 address:          192.0.2.18 (0xc0000212)
PSID start:            40 + p = 40 + 8 = 48
PSID length (q):       o - p = (End-user prefix len -
                       rule IPv6 prefix len) - p
                       = (56 - 40) - 8 = 8
PSID:                  0x34
```

```
Available ports (63 ranges): 1232-1235, 2256-2259, ..... ,
                                  63696-63699, 64720-64723
```

The BMR information allows a MAP CE to determine (complete) its IPv6 address within the indicated end-user IPv6 prefix.

```
IPv6 address of MAP CE:  2001:db8:0012:3400:0000:c000:0212:0034
```

Example 2 - BR:

Another example can be made of a MAP-T BR,
configured with the following FMR when receiving a packet
with the following characteristics:

```
IPv4 source address:       10.2.3.4 (0x0a020304)
TCP source port:           80
IPv4 destination address:  192.0.2.18 (0xc0000212)
TCP destination port:      1232

Forwarding Mapping Rule:  {2001:db8::/40 (Rule IPv6 prefix),
                           192.0.2.0/24 (Rule IPv4 prefix),
                           16 (Rule EA-bits length)}

MAP-T BR Prefix (DMR):    2001:db8:ffff::/64
```

The above information allows the BR to derive as follows
the mapped destination IPv6 address for the corresponding
MAP-T CE, and also the source IPv6 address for
the mapped IPv4 source address.

```
IPv4 suffix bits (p):  32 - 24 = 8 (18 (0x12))
PSID length:           8
PSID:  0               x34 (1232)
```

The resulting IPv6 packet will have the following header fields:

```
IPv6 source address:      2001:db8:ffff:0:000a:0203:0400::
IPv6 destination address: 2001:db8:0012:3400:0000:c000:0212:0034
TCP source Port:          80
TCP destination Port:     1232
```

Example 3- FMR:

An IPv4 host behind a MAP-T CE (configured as per the previous examples) corresponding with an IPv4 host 10.2.3.4 will have its packets converted into IPv6 using the DMR configured on the MAP-T CE as follows:

```
Default Mapping Rule:          {2001:db8:ffff::/64 (Rule IPv6 prefix),
                                0.0.0.0/0 (Rule IPv4 prefix)}

IPv4 source address:           192.0.2.18
IPv4 destination address:      10.2.3.4
IPv4 source port:              1232
IPv4 destination port:         80
MAP-T CE IPv6 source address:  2001:db8:0012:3400:0000:c000:0212:0034
IPv6 destination address:      2001:db8:ffff:0:000a:0203:0400::
```

Example 4 - Rule with no embedded address bits and no address sharing

```
End-user IPv6 prefix:     2001:db8:0012:3400::/56
Basic Mapping Rule:       {2001:db8:0012:3400::/56 (Rule IPv6 prefix),
                          192.0.2.1/32 (Rule IPv4 prefix),
                          0 (Rule EA-bits length)}
PSID length:              0 (Sharing ratio is 1)
PSID offset:              n/a
```

A MAP node can via the BMR or equivalent FMR, determine the IPv4 address and port-set as shown below:

```
EA bits offset:           0
IPv4 suffix bits (p):     Length of IPv4 address - IPv4 prefix
                          length = 32 - 32 = 0
IPv4 address:             192.0.2.18 (0xc0000212)
PSID start:               0
PSID length:              0
PSID:                     null
```

The BMR information allows a MAP CE also to determine (complete) its full IPv6 address by combining the IPv6 prefix with the MAP interface identifier (that embeds the IPv4 address).

```
IPv6 address of MAP CE:  2001:db8:0012:3400:0000:c000:0201:0000
```

Example 5 - Rule with no embedded address bits and address sharing (sharing ratio 256)

```
End-user IPv6 prefix:    2001:db8:0012:3400::/56
Basic Mapping Rule:      {2001:db8:0012:3400::/56 (Rule IPv6 prefix),
                         192.0.2.18/32 (Rule IPv4 prefix),
                         0 (Rule EA-bits length)}
PSID length:             (16 - (32 - 24)) = 8. Sharing ratio of 256.
                         Provisioned with DHCPv6.
PSID offset:             6 (default)
PSID:                    0x20 (Provisioned with DHCPv6)
```

A MAP node can via the BMR determine the IPv4 address and port-set as shown below:

```
EA bits offset:          0
IPv4 suffix bits (p):    Length of IPv4 address - IPv4 prefix
                         length = 32 -32 = 0
IPv4 address             192.0.2.18 (0xc0000212)
PSID start:              0
PSID length:             8
PSID:                    0x34

Available ports (63 ranges) : 1232-1235, 2256-2259, ..... ,
                              63696-63699, 64720-64723
```

The BMR information allows a MAP CE also to determine (complete) its full IPv6 address by combining the IPv6 prefix with the MAP interface identifier (that embeds the IPv4 address and PSID).

```
IPv6 address of MAP CE:  2001:db8:0012:3400:0000:c000:0212:0034
```

Note that the IPv4 address and PSID is not derived from the IPv6 prefix assigned to the CE, but provisioned separately using for example MAP options in DHCPv6.

Appendix B. Port mapping algorithm

The driving principles and the mathematical expression of the mapping algorithm used by MAP can be found in Appendix B of [I-D.ietf-softwire-map]

Authors' Addresses

Xing Li
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing 100084
CN

Email: xing@cernet.edu.cn

Congxiao Bao
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing 100084
CN

Email: congxiao@cernet.edu.cn

Wojciech Dec (editor)
Cisco Systems
Haarlerbergpark Haarlerbergweg 13-19
Amsterdam, NOORD-HOLLAND 1101 CH
Netherlands

Email: wdec@cisco.com

Ole Troan
Cisco Systems
Oslo
Norway

Email: ot@cisco.com

Satoru Matsushima
SoftBank Telecom
1-9-1 Higashi-Shinbashi, Munato-ku
Tokyo
Japan

Email: satoru.matsushima@tm.softbank.co.jp

Tetsuya Murakami
IP Infusion
1188 East Arques Avenue
Sunnyvale
USA

Email: tetsuya@ipinfusion.com

Softwire WG
Internet-Draft
Intended status: Standards Track
Expires: April 3, 2017

M. Boucadair
Orange
I. Farrer
Deutsche Telekom

September 30, 2016

Unified IPv4-in-IPv6 Softwire CPE: A DHCPv6-based Prioritization Mechanism
draft-ietf-softwire-unified-cpe-08

Abstract

In IPv6-only provider networks, transporting IPv4 packets encapsulated in IPv6 is a common solution to the problem of IPv4 service continuity. A number of differing functional approaches have been developed for this, each having their own specific characteristics. As these approaches share a similar functional architecture and use the same data plane mechanisms, this memo specifies a DHCPv6 option whereby a single CPE can interwork with all of the standardized and proposed approaches to providing encapsulated IPv4 in IPv6 services by providing a prioritization mechanism.

Requirements Language

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 RFC 2119 [RFC2119].

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 April 3, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Terminology 4
1.2. Rationale 4
1.3. DHCPv6 S46 Priority Option 4
1.4. DHCPv6 Client Behavior 5
1.5. DHCPv6 Server Behavior 6
2. Operator Deployment Considerations for Deploying Multiple Sotfwire Mechanisms 6
2.1. Client Address Planning 6
2.2. Backwards Compatability with Existing Softwire Clients . 7
3. Security Considerations 7
4. IANA Considerations 7
4.1. S46 Mechanisms and their Identifying Option Codes 8
5. Acknowledgements 8
6. References 8
6.1. Normative References 8
6.2. Informative References 9
Authors' Addresses 10

1. Introduction

IPv4 service continuity is one of the major technical challenges which must be considered during IPv6 migration. Over the past few years, a number of different approaches have been developed to assist with this problem (e.g., [RFC6333], [RFC7596], or [RFC7597]). These approaches, referred to as 'S46 mechanisms' in this document, exist in order to meet the particular deployment, scaling, addressing and other requirements of different service provider's networks.

A common feature shared between all of the differing modes is the integration of softwire tunnel end-point functionality into the

Customer Premise Equipment (CPE) router. Due to this inherent data plane similarity, a single CPE may be capable of supporting several different approaches. Users may also wish to configure a specific mode of operation.

A service provider's network may also have more than one S46 mechanism enabled in order to support a diverse CPE population with differing client functionality, such as during a migration between mechanisms, or where services require specific supporting softwire architectures.

For softwire based services to be successfully established, it is essential that the customer end-node, the service provider end-node and provisioning systems are able to indicate their capabilities and preferred mode of operation.

A number of DHCPv6 options for the provisioning of softwires have been standardized:

RFC6334 Defines DHCPv6 option 64 for configuring Basic Bridging BroadBand (B4, [RFC6333]) elements with the IPv6 address of the Address Family Transition Router (AFTR, [RFC6333]).
RFC7341 Defines DHCPv6 option 88 for configuring the address of a DHCPv4 over DHCPv6 server, which can then be used by a softwire client for obtaining further configuration.
RFC7598 Defines DHCPv6 options 94, 95 and 96 for provisioning Mapping of Address and Port with Encapsulation (MAP-E, [RFC7597]), Mapping of Address and Port using Translation (MAP-T, [RFC7599]), and Lightweight 4over6 [RFC7596] respectively.

This document describes a DHCPv6 based prioritization method whereby a CPE which supports several S46 mechanisms and receives configuration for more than one can prioritise which mechanism to use. The method requires no server side logic to be implemented and only uses a simple S46 mechanism prioritization to be implemented in the CPE.

The prioritization method as described here does not provide redundancy between S46 mechanisms for the client. I.e. If the highest priority S46 mechanism which has been provisioned to the client is not available for any reason, the means for identifying this and falling back to the S46 mechanism with the next highest priority is not in the scope of this document.

1.1. Terminology

This document makes use of the following terms:

- Address Family Transition Router (AFTR): is the IPv4-in-IPv6 tunnel termination point and the NAT44 function deployed in the operator's network [RFC6333].
- Border Relay (BR): a MAP-enabled router managed by the service provider at the edge of a MAP domain. A BR has at least an IPv6-enabled interface and an IPv4 interface connected to the native IPv4 network [RFC7597].
- Customer Premise Equipment (CPE): denotes the equipment at the customer edge that terminates the customer end of an IPv6 transitional tunnel. In some documents (e.g., [RFC7597]), this functional entity is called CE (Customer Edge).

1.2. Rationale

The following rationale has been adopted for this document:

(1) Simplify solution migration paths: Define unified CPE behavior, allowing for smooth migration between the different s46 mechanisms.
(2) Deterministic CPE co-existence behavior: Specify the behavior when several S46 mechanisms co-exist in the CPE.
(3) Deterministic service provider co-existence behavior: Specify the behavior when several modes co-exist in the service providers network.
(4) Re-usability: Maximize the re-use of existing functional blocks including tunnel end-points, port restricted NAPT44, forwarding behavior, etc.
(5) Solution agnostic: Adopt neutral terminology and avoid (as far as possible) overloading the document with solution-specific terms.
(6) Flexibility: Allow operators to compile CPE software only for the mode(s) necessary for their chosen deployment context(s).
(7) Simplicity: Provide a model that allows operators to only implement the specific mode(s) that they require without the additional complexity of unneeded modes.

1.3. DHCPv6 S46 Priority Option

The S46 Priority Option is used to convey a priority order of IPv4 service continuity mechanisms. Figure 1 shows the format of the S46 Priority Option.

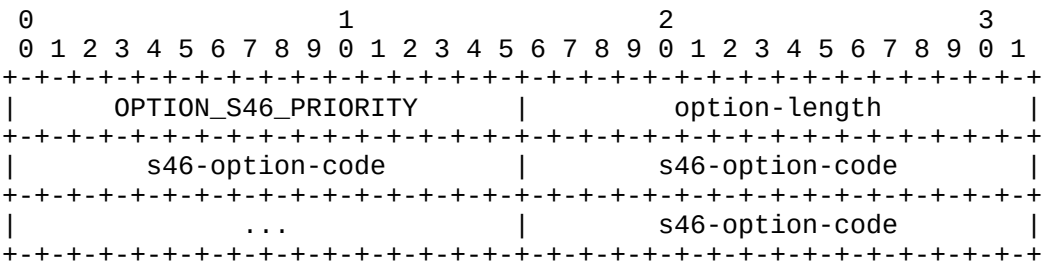


Figure 1: S46 Priority Option

- option-code: OPTION_S46_PRIORITY (TBD)
- option-length: >=2 and a multiple of 2, in octets.
- s46-option-code: 16-bits long IANA registered option code of the DHCPv6 option which is used to identify the softwire mechanism. S46 mechanism are prioritized in the appearance order in the S46 Priority Option.

Codes in OPTION_S46_PRIORITY are processed in order; in the event that a client receives more than one s46-option-code with a particular value, this should be considered as invalid. DHCP servers MAY validate the list of s46-option-code values to detect invalid values and duplicates. The option MUST contain at least one s46-option-code.

1.4. DHCPv6 Client Behavior

Clients MAY request option OPTION_S46_PRIORITY, as defined in [RFC3315], Sections 17.1.1, 18.1.1, 18.1.3, 18.1.4, 18.1.5, and 22.7. As a convenience to the reader, we mention here that the client includes requested option codes in the Option Request Option.

Upon receipt of a DHCPv6 Advertise message from the server containing OPTION_S46_PRIORITY the client performs the following steps:

1. Check the contents of the DHCPv6 message for options containing valid S46 mechanism configuration. A candidate list of possible S46 mechanisms is created from these option codes.
2. Check the contents of OPTION_S46_PRIORITY for the DHCPv6 option codes contained in the included s46-option-code fields. From this, an S46 mechanism priority list is created, ordered from highest to lowest following the appearance order.
3. Sequentially check the priority list against the candidate list until a match is found.
4. When a match is found, the client MUST configure the resulting S46 mechanism.

In the event that no match is found between the priority list and the candidate list, the client MAY proceed with configuring one or more of the provisioned S46 softwire mechanism(s). In this case, which mechanism(s) are chosen by the client is implementation-specific and not defined here.

If an invalid OPTION_S46_PRIORITY option is received, the client MAY proceed with configuring the provisioned S46 mechanisms as if OPTION_S46_PRIORITY had not been received.

If an unknown option code is received in OPTION_S46_PRIORITY option, the client MUST skip it and continue processing other listed option codes if they exist. The initial option codes that are allowed to be included in a OPTION_S46_PRIORITY option are listed in Section 4.1.

1.5. DHCPv6 Server Behavior

Sections 17.2.2 and 18.2 of [RFC3315] govern server operation in regards to option assignment. As a convenience to the reader, we mention here that the server will send a particular option code only if configured with specific values for that option code and if the client requested it.

Option OPTION_S46_PRIORITY is a singleton. Servers MUST NOT send more than one instance of the OPTION_S46_PRIORITY option.

2. Operator Deployment Considerations for Deploying Multiple Sotfwire Mechanisms

The following sub-sections describe some considerations for operators who are planning on implementing multiple softwire mechanisms in their network (e.g., during a migration between mechanisms).

2.1. Client Address Planning

As an operator’s available IPv4 resources are likely to be limited, it may be desirable to use a common range of IPv4 addresses across all of the active Softwire mechanisms. However, this is likely to result in difficulties in routing ingress IPv4 traffic to the correct Border Relay (BR)/AFTR instance which is actively serving a given CE. For example, a client which is configured to use MAP-E may send its traffic to the MAP-E BR, but on the return path, the ingress IP traffic gets routed to a MAP-T BR. The resulting translated packet that gets forwarded to the MAP-E client will be dropped.

Therefore, operators are advised to use separate IPv4 pools for each of the different mechanisms to simplify planning and IPv4 routing.

For IPv6 planning there is less of a constraint as the BR/AFTR elements for the different mechanisms can contain configuration for overlapping client's IPv6 addresses, providing only one mechanism is actively serving a given client at a time. However, the IPv6 address that is used as the tunnel concentrator's endpoint (BR/AFTR address) needs to be different for each mechanisms to ensure correct operation.

2.2. Backwards Compatability with Existing Softwire Clients

Deployed clients which can support multiple softwire mechanisms, but do not implement the prioritization mechanism described here may require additional planning. In this scenario, the CPE would request configuration for all of the supported softwire mechanisms in its DHCPv6 Option Request Option (ORO), but would not request OPTION_S46_PRIORITY. By default, the DHCPv6 server will respond with configuration for all of the requested mechanisms which could result in unpredictable and unwanted client configuration.

In this scenario, it may be necessary for the operator to implement logic within the DHCPv6 server to identify such clients and only provision them with configuration for a single softwire mechanism. It should be noted that this can lead to complexity and reduced scalability in the DHCPv6 server implementation due to the addition DHCPv6 message processing overhead.

3. Security Considerations

Security considerations discussed in [RFC6334] and [RFC7598] apply for this document.

Misbehaving intermediate nodes may alter the content of the S46 Priority Option. This may lead to setting a different IPv4 service continuity mechanism than the one initially preferred by the network side. Also, a misbehaving node may alter the content of the S46 Priority Option and other DHCPv6 options (e.g., DHCPv6 Option #64 or #90) so that the traffic is intercepted by an illegitimate node. Those attacks are not unique to the S46 Priority Option but are applicable to any DHCPv6 option that can be altered by a misbehaving intermediate node.

4. IANA Considerations

IANA is kindly requested to allocate the following DHCPv6 option code:

TBD for OPTION_S46_PRIORITY

All values should be added to the DHCPv6 option code space defined in Section 24.3 of [RFC3315].

4.1. S46 Mechanisms and their Identifying Option Codes

This document requests that IANA create a new registry entitled "Option Codes permitted in the S46 Priority Option". This registry will enumerate the set of DHCPv6 Option Codes that can be included in OPTION_S46_PRIORITY option. Options may be added to this list using the IETF Review process described in Section 4.1 of [RFC5226].

The following table shows the option codes which are currently defined and the S46 mechanisms which they represent. The contents of this table shows the format and the initial values for the new registry. Option codes that have not been requested to be added according to the stated procedure should not be mentioned at all in the table, and should not be listed as "reserved" or "unassigned". The valid range of values for the registry is the range of DHCPv6 Option Codes (1-65535).

Option Code	S46 Mechanism	Reference
64	DS-Lite	[RFC6334]
88	DHCPv4 over DHCPv6	[RFC7341]
94	MAP-E	[RFC7598]
95	MAP-T	[RFC7598]
96	Lightweight 4over6	[RFC7598]

Table 1: DHCPv6 Option to S46 Mechanism Mappings

5. Acknowledgements

Many thanks to O. Troan, S. Barth. A. Yourtchenko, B. Volz, T. Mrugalski, J. Scudder, P. Kyzivat, F. Baker, and B. Campbell for their input and suggestions.

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC6334] Hankins, D. and T. Mrugalski, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite", RFC 6334, DOI 10.17487/RFC6334, August 2011, <http://www.rfc-editor.org/info/rfc6334>.

[RFC7341] Sun, Q., Cui, Y., Siodelski, M., Krishnan, S., and I. Farrer, "DHCPv4-over-DHCPv6 (DHCP 4o6) Transport", RFC 7341, DOI 10.17487/RFC7341, August 2014, <http://www.rfc-editor.org/info/rfc7341>.

[RFC7598] Mrugalski, T., Troan, O., Farrer, I., Perreault, S., Dec, W., Bao, C., Yeh, L., and X. Deng, "DHCPv6 Options for Configuration of Softwire Address and Port-Mapped Clients", RFC 7598, DOI 10.17487/RFC7598, July 2015, <http://www.rfc-editor.org/info/rfc7598>.

6.2. Informative References

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <http://www.rfc-editor.org/info/rfc6333>.

[RFC7596] Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the Dual-Stack Lite Architecture", RFC 7596, DOI 10.17487/RFC7596, July 2015, <http://www.rfc-editor.org/info/rfc7596>.

[RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", RFC 7597, DOI 10.17487/RFC7597, July 2015, <http://www.rfc-editor.org/info/rfc7597>.

[RFC7599] Li, X., Bao, C., Dec, W., Ed., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", RFC 7599, DOI 10.17487/RFC7599, July 2015, <http://www.rfc-editor.org/info/rfc7599>.

Authors' Addresses

Mohamed Boucadair
Orange
Rennes
France

Email: mohamed.boucadair@orange.com

Ian Farrer
Deutsche Telekom
Germany

Email: ian.farrer@telekom.de

Softwire Working Group
Internet-Draft
Intended status: Informational
Expires: January 16, 2014

Y. Lee
Comcast
Q. Sun
China Telecom
C. Liu
Tsinghua University
July 15, 2013

Simple Failover Mechanism for Lightweight 4over6
draft-lee-softwire-lw4over6-failover-01

Abstract

This memo specifies a simple mechanism for Lightweight AFTR (lwAFTR) to notify Lightweight B4 (lwB4) to initiate the recreation of the binding when lwAFTR does not have the subscriber mapping in the mapping table. This often happens at failover the backup lwAFTR does not have the subscriber mapping information to process the packets between lwB4 and external IPv4 host.

Requirements Language

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].

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 January 16, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Background 3
2. Failover Use Case 3
3. Failover Setup Considerations 4
3.1. Backup lwAFTR Discovery Consideration 4
3.2. lwB4 IPv4 Prefix Management Consideration 5
3.3. lwB4 IPv4 Address Provisioning 6
3.3.1. DHCPv4-over-DHCPv6 6
3.3.2. Port Control Protocol 6
4. Failover Trigger Mechanisms 7
5. Control Message Trigger Failover 7
5.1. Tunnel Concentrator Behavior 7
5.2. Tunnel Initiator Behavior 8
6. Data Packet Trigger Failover 8
6.1. Tunnel Concentrator Behavior 8
6.2. Tunnel Initiator Behavior 8
7. IANA Considerations 8
8. Security Considerations 9
9. Acknowledgements 9
10. References 9
10.1. Normative References 9
10.2. Informative References 9
Authors' Addresses 10

1. Background

Lightweight 4over6 [I-D.ietf-softwire-lw4over6] defines that Lightweight AFTR (lwAFTR) stores per subscriber binding. The subscriber binding entry is usually created when the Lightweight B4 (lwB4) successfully requested IPv4 resource from the provisioning system. lwAFTR could be in the provisioning path between lwB4 and the provisioning system. This allows lwAFTR to listen to the provisioning messages and create the binding on demand. The exact mechanism is out of scope.

The AFTR's subscriber binding table is used to map the subscriber's IPv6 address to the IPv4 resource (i.e., full IPv4 address or restricted IPv4 address). Due to security reason, entries in the table are usually created after a successful lwB4 provisioning. This means the network knows the lwB4 and authorizes the lwAFTR to provide lightweight 4over6 services. Consider when the primary lwAFTR failed, the Backup lwAFTR might not have the binding entry in the table because the Backup lwAFTR was not in the original provisioning path. This requires the Backup lwAFTR to notify lwB4 to trigger provisioning request so that the Backup lwAFTR can create the binding entry. This memo defines two simple mechanisms to let the Backup lwAFTR to create the subscriber binding after failover.

2. Failover Use Case

Consider a typical deployment model that a set of lwAFTRs were all provisioned with the same IPv6 anycast address. When lwB4 booted up, it sent a dhcpv4 request over dhcpv6 [I-D.ietf-dhc-dhcpv4-over-dhcpv6] to the primary lwAFTR (e.g., lwAFTR1). lwAFTR1 created the subscriber binding and started providing lightweight 4over6 service. At some point lwAFTR1 failed. Network converged and the Backup lwAFTR (e.g., lwAFTR2) became the active lwAFTR serving the lwB4s previously served by lwAFTR1. However, when lwAFTR2 received an IPv6 packet sourced from the lwB4, lwAFTR2 would fail to perform decapsulation and forward the IPv4 packet because lwAFTR2 didn't have the subscriber binding in the table.

In this use case there are four assumptions:

1. lwAFTR in the same failover group use the same IPv6 anycast address for the Softwire interface

2. The subscriber binding entry is created on-demand upon successfully lwB4 provisioning

3. IPv4 provisioning mechanism is dynamically required by the lwB4

4. IPv4 address used by the lwB4 is either static or dynamic

In this memo, we only consider the failover scenario in deployments with these assumptions. Other deployments such as using static provisioning are out of scope.

3. Failover Setup Considerations

To provide minimal impact to users during failover, there are some considerations:

- Backup lwAFTR Discovery
- lwB4 IPv4 Prefix Management
- lwB4 IPv4 Address Provisioning

3.1. Backup lwAFTR Discovery Consideration

During failover, fast service recovery relies on how fast the lwB4 to detect and discover the Backup lwAFTR. In this memo, we suppose lwAFTR serving a failover group will all use the same IPv6 anycast address for the softwire interface. When the Primary lwAFTR fails, lwB4 will rely on IP routing to discover the closest Backup lwAFTR. This mechanism does not require any pre-configuration in the lwB4. The time lwB4 to discover the Backup lwAFTR relies on how fast the routing protocol converges.

When mulitple Primary (or Backup) lwAFTRs using the same anycast address, the intermediate routers can using Equal Cost Multi-Path (ECMP) to load-balancing session among the lwAFTR. [RFC6437] proposes to use IPV6 flow label for the load-balancing entropy, but this requires the lwB4 to generate the flow label. Since most existing routers support load-balancing by hashing of the three-tuple of IP header, we recommend to use this as entropy field for the load-balancing. Somebody may argue 3-tuple may not seem unique enough to randomize session in IPv4. But IPv6 address is 4 times longer than IPv4 address, it guarantees way better uniqueness for the hash.

lwAFTR must stop announcing the anycast address when it no longer provides lightweight 4over6 service. This is critical to prevent lwB4’s packets from reaching the failed lwAFTR.

3.2. lwB4 IPv4 Prefix Management Consideration

Given service agreement, some users may expect their IPv4 addresses will not change due to failover. This is particular important for server applications which require to accept external connections using a given static IPv4 address. Others may accept dynamic IPv4 address which may change after failover. In reality, an operator may have a mixed scheme for both static and dynamic IPv4 prefixes. The business decision of IPv4 prefix management is out of scope of this memo. However, the decision will have impact in failover design.

For the same failover group, operators usually have two choices to manage the IPv4 prefix for lwAFTR:

Case 1: Each lwAFTR is given different IPv4 prefixes

Case 2: All lwAFTR are given identical IPv4 prefixes

Case 1 supports dynamic IPv4 scenario. lwB4 does not require to use the same IPv4 address after failover. Hence, both Primary and Backup lwAFTRs are advertising their own IPv4 prefixes. When Backup lwAFTR receives a packet sourcing from an unknown IPv6 address (i.e., fail to find a match in the subscriber binding table), it will silently drop the packet. Backup lwAFTR is not required to know the status of Primary lwAFTRs. In fact, both Primary and Backup lwAFTRs are running autonomously.

Case 2 supports the static IPv4 scenario. lwB4 expects the IPv4 will stay unchanged during failover. When the Primary lwAFTR fails, the Backup lwAFTR will take over the IPv4 prefix and start accepting packets designated to that IPv4 prefix. This requirement implies the following steps:

1. Primary and Backup lwAFTRs are running dynamic routing protocol

2. Primary lwAFTR set the routing matrix higher than the Backup lwAFTR does for the serving IPv4 prefix

3. When Primary detects problem, it stops advertising the IPv4 prefix

4. Routing protocol converges and the Backup lwAFTR is the router announcing the IPv4 prefix

The above steps requires the Primary and Backup lwAFTR must run dynamic routing protocol. At any given time, only the serving lwAFTR is the next-hop router of the IPv4 prefix. This implies the operator must manually configure the routing matrix of the IPv4 prefix so that

the Backup lwAFTR will be the next-hop only if the Primary lwAFTR withdraws announcing the prefix.

In both cases, when the Backup lwAFTR receives the IPv4 packet, the Backup lwAFTR must identify the lw4over6 B4 and send the packet to the lwB4. This requires the Backup lwAFTR to know the subscriber binding. Section 4 will discuss more in details.

3.3. lwB4 IPv4 Address Provisioning

When lwB4 starts up, it will need to acquire IPv4 resources. There are multiple ways to acquire IPv4 address and restricted port-set. In this memo, we make no assumption how to obtain the IPv4 resources. Given a provisioning method, there are implications when failover occurs. In this memo, we discuss failover impacts to DHCPv4-over-DHCPv6 [I-D.ietf-dhc-dhcpv4-over-dhcpv6] and PCP Port-Set [I-D.ietf-pcp-port-set].

3.3.1. DHCPv4-over-DHCPv6

Operator may use DHCPv4 to provision IPv4 address to the lwB4. Since the access network is IPv6, the DHCPv4 messages must be encapsulated into DHCPv6 message to deliver between DHCP server and lwB4. In a typical deployment, the DHCP server is a centralized DHCP server and lwAFTR is the DHCP relay agent to relay the dhcp messages to the server over unicast. Rarely DHCP server will collocate with the lwAFTR to provision IPv4 resources to the lwB4. We consider the collocated DHCP server is out of scope.

DHCPv6 client uses a link-scoped multicast [RFC3315] to communicate with neighboring relay agents and servers. If the Primary and Backup lwAFTRs are the lwB4’s next-hop IPv6 routers, they must act as a dhcpv6 relay agent and listen to the DHCP multicast request. If they are not the next-hop IPv6 router, the next-hop router must relay the dhcp packet over unicast to the lwAFTR’s anycast address. This will allow the Backup lwAFTR to receive the dhcp message and create the subscriber binding at failover.

3.3.2. Port Control Protocol

Operator may also use PCP Port-set Option [I-D.ietf-pcp-port-set] to provision IPv4 address and port-set to the lwB4. In a typical deployment, PCP server [RFC6887] will collocate with lwAFTR, and the subscriber binding can be determined by the lwAFTR. The PCP request should be sent to the lwAFTR’s anycast address. It is uncommon that PCP server will be centralized deployed in which the lwAFTR is the PCP proxy to relay PCP requests. We consider the centralized PCP server is out of scope in this document.

If the Primary and Backup lwAFTR are the lwB4's next-hop IPv6 routers, the PCP requests can be sent in the plain mode. However, if the lwAFTRs are not the lwB4's next-hop IPv6 routers and multiple Primary lwAFTRs are using anycast address to achieve ECMP load-balancing. When using EMCP load-balancing, it is possible that intermediate routers will perform 3-tuple hash on the plain PCP packets while doing 5-tuple hash for subsequent softwire traffic. This may result inconsistent path selection (e.g., PCP request may arrive in one Primary lwAFTR while softwire packets may arrive in a different Primary lwAFTR). Therefore, the PCP request SHOULD also been encapsulated into IPv6 tunnel and apply the same 3-tuple hash on the outer IPv6 header. This guarantees the same hash will be used for both PCP and Softwire packets.

4. Failover Trigger Mechanisms

For Control Message Trigger Failover, when a lwAFTR receives an IPv6 packet from an unknown lwB4 from its tunnel interface, it sends an ICMP error message to the lwB4. When lwB4 receives the ICMP error message, it must send the provisioning request to the network to trigger the subscriber entry creation in the lwAFTR.

For Data Packet Trigger Failover, when a lwAFTR receives a packet which contains an unknown lwB4 from its tunnel interface, it must validate the source IPv4 address whether it is assigned by the provisioning system to the user. The validation mechanism is deployment specific. If the lwAFTR is next-hop of the lwB4, DHCP Lease Query may be used to validate the IPv4 address. Other methods such as proprietary out-of-band verification may be used. After successfully validation, lwAFTR will create the binding entry.

5. Control Message Trigger Failover

5.1. Tunnel Concentrator Behavior

When lwAFTR receives a packet in its tunnel interface:

1. It must check its subscriber binding table against the IPv6 source address of the encapsulated packet.

2. If an entry is found, forward the packet.

3. If an entry is not found, send an ICMPv6 Error Message (Type 1 Code 0)

5.2. Tunnel Initiator Behavior

When lwB4 receives an ICMPv6 Error Message (Type 1 Code 0), it must start the provisioning mechanism to request IPv4 resource.

lwB4 may be setup to receive external initiated sessions. This is important for the lwB4 to periodically verify the binding entry in the lwAFTR. Therefore, lwB4 must send packets (e.g. PING) periodically to the lwAFTR.

6. Data Packet Trigger Failover

6.1. Tunnel Concentrator Behavior

When lwAFTR receives a packet in its tunnel interface:

1. It must check its subscriber binding table against the IPv6 source address of the encapsulated packet.

2. If an entry is found, forward the packet.

3. If an entry is not found, extract the IPv4 source address from the encapsulated packet.

4. Validate the IPv4 address is the IPv4 address provisioned to the user. The validation mechanism is out of scope.

5. Upon successful validation, create an entry in the subscriber binding table.

6.2. Tunnel Initiator Behavior

lwB4 is transparent to the failover. lwB4 will continue to send packets to the backup lwAFTR.

lwB4 may be setup to receive external initiated sessions. This is important for the lwB4 to periodically verify the binding entry in the lwAFTR. Therefore, lwB4 must send packets (e.g. PING) periodically to the lwAFTR.

7. IANA Considerations

8. Security Considerations

TBD

9. Acknowledgements

TBD

10. References

10.1. Normative References

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 2013.

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

10.2. Informative References

[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-dhcpv6-00 (work in progress), April 2013.

[I-D.ietf-pcp-port-set]
Sun, 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-02 (work in progress), July 2013.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, November 2011.

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

Authors' Addresses

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

Email: yiu_lee@cable.comcast.com

Qiong Sun
China Telecom
Room 708, No.118, Xizhimennei Street
Beijing 100084
P.R. China

Email: sunqiong@ctbri.com.cn

Cong Liu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6278-5822
Email: gnocuil@gmail.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 13, 2014

Z. Li
S. Zhuang
H. Ni
Huawei Technologies
July 12, 2013

Connecting IPv6 Multicast Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)
draft-li-idr-mcast-6pe-00

Abstract

This document defines a new Network Layer Reachability Information (NLRI), called as the MCAST-6PE NLRI. The MCAST-6PE NLRI is used to interconnect IPv6 C-Multicast islands over a Multiprotocol Label Switching (MPLS)-enabled IPv4 cloud. This approach relies on IPv6 Provider Edge routers (6PE), which can exchange the IPv6 C-Multicast reachability information transparently over the core using the Multiprotocol Border Gateway Protocol (MP-BGP) over IPv4. This document describes the BGP encodings and procedures for exchanging the information elements required by IPv6 Multicast in 6PE. MPLS-based Service Providers may use the 6PE Multicast mechanism to provide IPv6 Multicast service for customers.

Requirements Language

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 RFC 2119 [RFC2119].

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 January 13, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Terminology 3
3. MCAST-6PE NLRI 4
3.1. Intra-AS 6PE I-PMSI A-D Route 5
3.2. Inter-AS 6PE I-PMSI A-D route 6
3.3. 6PE S-PMSI A-D Route 6
3.4. 6PE Leaf A-D Route 7
3.5. 6PE Source Active A-D Route 7
3.6. 6PE C-Multicast Route 8
4. PMSI Tunnel Attribute 9
5. Source AS Extended Community 9
6. Route Import Extended Community 9
7. PE Distinguisher Labels Attribute 10
8. Operations 10
8.1. BGP-Based MCAST-6PE Membership Auto-Discovery 10
8.1.1. Intra-AS Operations 10
8.1.2. Inter-AS Operations 11
8.2. PE-PE Transmission of IPv6 C-Multicast Routing 11
8.2.1. Selecting the Upstream Multicast Hop (UMH) 11
8.2.2. Signaling P-tunnel 12
8.2.3. Use of BGP for Carrying IPv6 C-Multicast Routing . . 12
8.2.4. Propagating IPv6 C-Multicast Routes by an ASBR . . . 14
8.3. Using 6PE S-PMSI A-D Routes to Bind C-Trees to P-Tunnels 14
9. IANA Considerations 14
10. Security Considerations 14
11. References 14
11.1. Normative References 14
11.2. Informative References 15
Authors' Addresses 15

1. Introduction

6PE[RFC4798] defines a mechanism to interconnect IPv6 islands over an MPLS-enabled IPv4 cloud using the IPv6 Provider Edge routers (6PE) approach. In this document an 'IPv6 island' is a network running native IPv6 as per [RFC2460]. A typical example of an IPv6 island would be a customer's IPv6 site connected via its IPv6 Customer Edge (CE) router to one (or more) Dual Stack Provider Edge router(s) of a Service Provider. These IPv6 Provider Edge routers (6PE) are connected to an IPv4 MPLS core network. This document defines a new Network Layer Reachability Information (NLRI), called as the MCAST-6PE NLRI. The MCAST-6PE NLRI is used to interconnect IPv6 C-Multicast islands over a Multiprotocol Label Switching (MPLS)-enabled IPv4 cloud. This approach relies on IPv6 Provider Edge routers (6PE), which can exchange the IPv6 C-Multicast reachability information transparently over the core using the Multiprotocol Border Gateway Protocol (MP-BGP) over IPv4. This document describes the BGP encodings and procedures for exchanging the information elements required by IPv6 Multicast in 6PE. MPLS-based Service Providers may use the 6PE Multicast mechanism to provide IPv6 Multicast service for customers.

2. Terminology

This document uses terminology from [RFC4798], [RFC6513], [RFC6514].

Term Definition

---------- --

6PE: IPv6 Provider Edge routers

A-D: auto-discovery

BGP: Border Gateway Protocol

CE: customer edge

C-G: customer multicast group address

C-join: customer join message

C-multicast: customer multicast

C-PIM: customer PIM

C-RP: customer rendezvous point

C-RPT: customer RP Tree

C-S: customer multicast source address

I-PMSI: inclusive PMSI

LSP: label switched path

MCAST: multicast

mLDP: multipoint Label Distribution Protocol

MP2MP: multipoint to multipoint

MVPN: multicast VPN

NG MVPN: next-generation multicast VPN

NLRI: Network Layer Reachability Information

OIL: outgoing interface list

P2MP: point to multipoint PE: provider edge

PIM: Protocol Independent Multicast

PMSI: Provider Multicast Service Interface

P-group: Provider multicast group

P-join: Provider join message

P-PIM: Provider PIM

P-RP: Provider Rendezvous Point

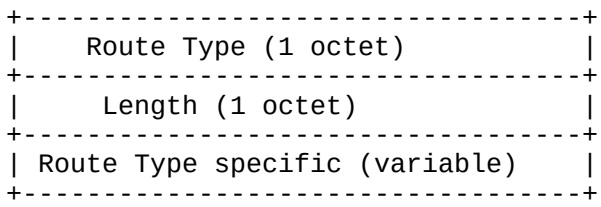
SAFI: Subsequent Address Family Identifier

S-PMSI: Selective PMSI

UMH: Upstream Multicast Hop

3. MCAST-6PE NLRI

This document defines a new BGP NLRI, called as the MCAST-6PE NLRI. Following is the format of the MCAST-6PE NLRI:



The Route Type field defines the encoding of the rest of MCAST-6PE NLRI (Route Type specific MCAST-6PE NLRI). The Length field indicates the length in octets of the Route Type specific field of the MCAST-6PE NLRI. This document defines the following Route Types for A-D routes:

+ 1 - Intra-AS 6PE I-PMSI A-D route;

+ 2 - Inter-AS 6PE I-PMSI A-D route;

+ 3 - 6PE S-PMSI A-D route;

+ 4 - 6PE Leaf A-D route;

+ 5 - 6PE Source Active A-D route.

This document defines the following Route Types for IPv6 C-multicast routes:

+ 6 - 6PE Shared Tree Join route;

+ 7 - 6PE Source Tree Join route;

The MCAST-6PE NLRI is carried in BGP using BGP Multiprotocol Extensions [RFC4760] with an AFI of 2 (IPv6 AFI), and a SAFI of MCAST-6PE [To be assigned by IANA]. The NLRI field in the MP_REACH_NLRI / MP_UNREACH_NLRI attribute contains the MCAST-6PE NLRI (encoded as specified above). The following sections describe the format of the Route Type specific MCAST-6PE NLRI for various Route Types defined in this document.

3.1. Intra-AS 6PE I-PMSI A-D Route

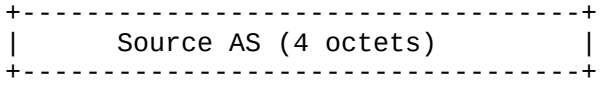
An Intra-AS 6PE I-PMSI A-D Route Type specific MCAST-6PE NLRI consists of the following:

Originating Router's IP Addr filed set to the IP address of the MCAST 6PE router originating this route, which is typically the primary loopback address of the MCAST 6PE router.

All MCAST 6PE routers create and advertise a Type 1 intra-AS 6PE I-PMSI A-D route for IPv6 MCAST service to which they are connected.

3.2. Inter-AS 6PE I-PMSI A-D route

An Inter-AS 6PE I-PMSI A-D Route Type specific MCAST-6PE NLRI consists of the following:



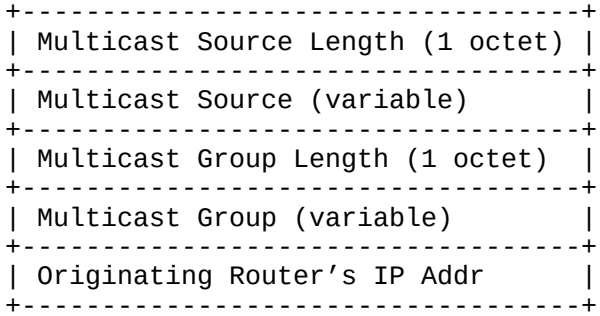
The Source AS contains an Autonomous System Number (ASN), 4 octets.

Two-octet ASNs are encoded in the two low-order octets of the Source AS field, with the two high-order octets set to zero.

Type 2 routes are used for MCAST 6PE membership discovery between MCAST-6PE routers that belong to different ASes.

3.3. 6PE S-PMSI A-D Route

A 6PE S-PMSI A-D Route Type specific MCAST-6PE NLRI consists of the following:



For MCAST-6PE, the Multicast Source field contains the C-S address i.e. the address of the multicast source, which is an IPv6 address, then the value of the Multicast Source Length field is 128 bits.

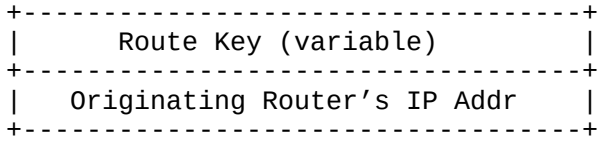
For MCAST-6PE, the Multicast Group field contains the C-G address i.e. the address of the multicast group, which is an IPv6 address, then the value of the Multicast Group Length field is 128 bits.

The Originating Router's IP Addr filed set to the IP address of the MCAST-6PE router originating this route, which is typically the primary loopback address of the MCAST-6PE router.

A sender MCAST-6PE that initiates a selective P-tunnel is required to originate a Type 3 6PE S-PMSI A-D route with the appropriate PMSI attribute.

3.4. 6PE Leaf A-D Route

A 6PE Leaf A-D Route Type specific MCAST-6PE NLRI consists of the following:



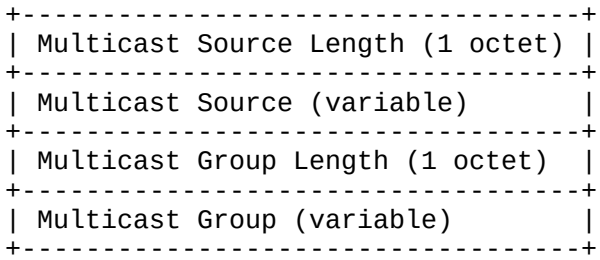
The Route Key field contains the original Type 3 route received. The Originating Router's IP Addr filed set to the IP address of the MCAST-6PE originating the 6PE leaf A-D route, typically the primary loopback address.

A 6PE Leaf A-D routes may be originated as a result of processing a received Inter-AS 6PE I-PMSI A-D route [Type 2] or 6PE S-PMSI A-D route [Type 3]. A 6PE Leaf A-D route is originated in these situations only if the received route has a PMSI Tunnel attribute whose "Leaf Information Required" bit is set to 1.

Typically a receiver MCAST-PE router responds to a Type 3 route by originating a Type 4 6PE leaf A-D route if it has local receivers interested in the traffic transmitted on the selective P-tunnel. The Type 4 route informs the sender MCAST-6PE of the leaf MCAST-6PE routers.

3.5. 6PE Source Active A-D Route

A 6PE Source Active A-D Route Type specific MCAST-6PE NLRI consists of the following:



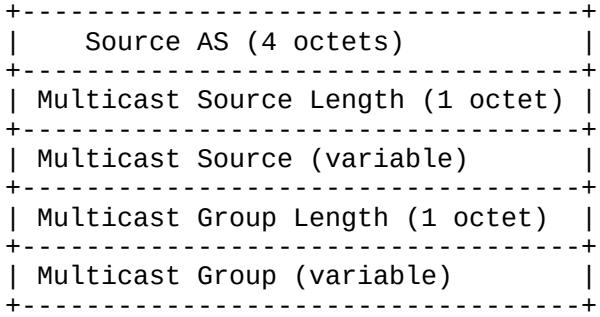
For MCAST-6PE, the Multicast Source field contains the C-S address i.e. the address of the multicast source, which is an IPv6 address, then the value of the Multicast Source Length field is 128 bits.

For MCAST-6PE, the Multicast Group field contains the C-G address i.e. the address of the multicast group, which is an IPv6 address, then the value of the Multicast Source Length field is 128 bits.

Type 5 6PE Source Active A-D routes carry information about active IPv6 Multicast sources and the groups to which they are transmitting data. These routes can be generated by any MCAST-6PE router that becomes aware of an active source.

3.6. 6PE C-Multicast Route

A 6PE Shared Tree Join Route and a 6PE Source Tree Join Route Type specific MCAST-6PE NLRI consists of the following:



The Source AS contains an ASN, 4 octets. Two-octet ASNs are encoded in the low-order two octets of the Source AS field.

For MCAST-6PE, the Multicast Source field contains an IPv6 address, then the value of the Multicast Source Length field is 128 bits. For a 6PE Shared Tree Join Route, the Multicast Source field contains the C-RP address; for a 6PE Source Tree Join Route, the Multicast Source field contains the C-S address.

For MCAST-6PE, the Multicast Group field contains an IPv6 address, then the value of the Multicast Group Length field is 128 bits. The Multicast Group field contains the C-G address.

The 6PE C-Multicast Routes exchange between MCAST-6PE routers refers to the propagation of C-joins from receiver MCAST-6PEs to the sender MCAST-6PEs.

In a 6PE MCAST Network, IPv6 C-joins received by MCAST-6PE Router from the CEs are encoded as BGP 6PE C-Multicast Routes and advertised via 6PE C-Multicast Routes towards the sender MCAST-6PEs. Two types of 6PE C-Multicast Routes are specified. The Type 6 6PE C-Multicast Routes are used in representing information contained in a shared tree (C-*, C-G) join. The Type 7 6PE C-Multicast Routes are used in representing information contained in a source tree (C-S, C-G) join.

4. PMSI Tunnel Attribute

The usage of PMSI Tunnel Attribute is described in [RFC6514].

5. Source AS Extended Community

The Source AS is an AS-specific Extended Community, of an extended type, and is transitive across AS boundaries [RFC4360]. The Global Administrator field of this Community MUST be set to the ASN of the MCAST-6PE router. The Local Administrator field of this Community MUST be set to 0.

The usage of a received Source AS Extended Community in MCAST 6PE is the same as described in [RFC6514].

6. Route Import Extended Community

This document defines a new BGP Extended Community called "Route Import", type value is to be assigned by IANA.The Route Import Extended Community is an IP-address-specific extended community that is used for importing IPv6 C-Multicast routes in the active sender MCAST-6PE router’s MCAST-6PE routing table to which the source is attached. For MCAST-6PE Network case, for constructing IPv6 C-Multicast Import RT, the Local Administrator is set to 0 and the Global Administrator field MUST be set to an IP address of the MCAST-6PE router.

7. PE Distinguisher Labels Attribute

The usage of PE Distinguisher Labels Attribute is described in [RFC6513].

8. Operations

8.1. BGP-Based MCAST-6PE Membership Auto-Discovery

This section specifies procedures for the auto-discovery of MCAST-6PE memberships and the distribution of information used to instantiate I-PMSIs.

There are two MCAST-6PE auto-discovery mechanisms, dubbed "intra- AS" and "inter-AS" respectively. The intra-AS mechanisms provide auto-discovery within a single AS. The inter-AS mechanisms provide auto-discovery across multiple ASes when segmented inter-AS tunnels are being used.

BGP-Based MCAST-6PE Membership Auto-Discovery is done by means of a new address family, the MCAST-6PE address family. Any PE that attaches to a MCAST-6PE service MUST issue a BGP Update message containing a NLRI in this address family, along with a specific set of attributes.

8.1.1. Intra-AS Operations

This section describes exchanges of Type 1 Intra-AS 6PE I-PMSI A-D routes originated/received by PEs within the same AS.

To participate in the MCAST-6PE auto-discovery, a PE router that provides MCAST 6PE service MUST originate an Intra-AS 6PE I-PMSI A-D route and advertises this route in IBGP. The route is constructed as follows.

The route carries a single MCAST-6PE NLRI with the Originating Router’s IP Addr field set to the IP address of the MCAST 6PE router originating this route. Note that the <Originating Router’s IP Addr> uniquely identifies a given MCAST-6PE router.

The route carries the PMSI Tunnel attribute if and only if an I-PMSI is used for the MCAST-6PE (the conditions under which an I-PMSI is used can be found in [RFC6513]). Depending on the technology used for the P-tunnel for the MCAST-6PE on the PE, the PMSI Tunnel attribute of the Intra-AS 6PE I-PMSI A-D route is the same as described in [RFC6514].

The Next Hop field of the MP_REACH_NLRI attribute of the route MUST be set to the same IP address as the one carried in the Originating Router's IP Addr field.

When PE-PE Type 1 intra-AS 6PE I-PMSI A-D routes are exchanged among all provider routers, every PE can know the MCAST-6PE neighbors to itself.

8.1.2. Inter-AS Operations

This section applies only to the case where segmented inter-AS tunnels are used.

Type 2 routes are used for MCAST 6PE membership discovery between MCAST-6PE routers that belong to different ASs.

If an ASBR is configured to support MCAST 6PE service, the ASBR MUST participate in the intra-AS MCAST 6PE auto-discovery procedures, for that MCAST 6PE within the ASBR's own AS, as specified in Section ;"Intra-AS Operations".

A Type 2 Inter-AS 6PE I-PMSI A-D route for MCAST 6PE originated by an ASBR within a given AS is propagated via BGP to other ASes.

The route carries a single MCAST-6PE NLRI with the Source AS field set to the ASBR's own AS.

When re-advertising an Inter-AS 6PE I-PMSI A-D route, the ASBR MUST set the Next Hop field of the MP_REACH_NLRI attribute to a routable IP address of the ASBR.

8.2. PE-PE Transmission of IPv6 C-Multicast Routing

IPv6 C-Multicast Routing Information is exchanged among PEs by using 6PE C-multicast routes that are carried using an MCAST-6PE NLRI. These routes are originated and propagated as follows.

8.2.1. Selecting the Upstream Multicast Hop (UMH)

Section 5.1 of [RFC6513] describes the method of Selecting the Upstream Multicast Hop (UMH). Constructing the C-Multicast Import RT as specified in Section 7 of [RFC6514].

For a PE as the MCAST-6PE sender, issues the UMH route through a 6PE UCAST route carrying Route Import Extended Community and Source AS Extended Community.

The Route Import Extended Community is an IP-address-specific extended community that is used for importing IPv6 C-Multicast routes in the active sender MCAST-6PE's MCAST-6PE routing table to which the source is attached. For MCAST-6PE Network case, for constructing IPv6 C-Multicast Import RT, the Local Administrator is set to 0 and the Global Administrator field MUST be set to an IP address of the MCAST-6PE router.

8.2.2. Signaling P-tunnel

The PMSI tunnel attribute carries information about the P-tunnel. In a MCAST-6PE Network, the sender PE router sets up the P-tunnel, and therefore is responsible for originating the PMSI tunnel attribute. The PMSI tunnel attribute can be attached to Type 1, Type 2, and Type 3 routes.

The MCAST-6PE sender router, attaches a PMSI tunnel attribute to Type 1 Intra-AS 6PE I-PMSI A-D Route, begins to signal P-tunnel for MCAST-6PE Network.

MCAST-6PE sender sends Type 1 route to other PEs, when other PEs receive the Type 1 route with PMSI tunnel attribute from MCAST-6PE sender, then join the P-tunnel.

8.2.3. Use of BGP for Carrying IPv6 C-Multicast Routing

Part of the procedures for constructing MCAST-6PE NLRI depends on the multicast routing protocol between CE and PE (C-multicast protocol).

8.2.3.1. PIM as the C-Multicast Protocol

Whenever (a) a C-PIM instance on a particular PE creates a new (C-S,C-G) state, and (b) the selected upstream PE for C-S (see [RFC6513]) is not the local PE, then the local PE MUST originate a C-multicast route of type Source Tree Join. The Multicast Source field in the MCAST-6PE NLRI of the route is set to C-S; the Multicast Group field is set of C-G.

This C-multicast route is said to "correspond" to the C-PIM (C-S,C-G) state.

The semantics of the route are such that the PE has one or more receivers for (C-S,C-G) in the sites connected to the PE (the route has the (C-S,C-G) Join semantics).

Whenever a C-PIM instance on a particular PE deletes a (C-S,C-G) state, the corresponding C-multicast route MUST be withdrawn. (The withdrawal of the route has the (C-S,C-G) Prune semantics). The

MCAST-6PE NLRI of the withdrawn route is carried in the MP_UNREACH_NLRI attribute.

8.2.3.1.1. Source Tree Join (C-S, C-G)

When receiver PE receives a source tree join (C-S, C-G) from CE, it does a route look up for C-S. If there is more than one route, the receiver PE chooses a single forwarder PE. The procedures used for choosing a single forwarder are outlined in [RFC6514]. When the C-S route has been selected, the receiver PE will originate a Type 7 route, carrying Route Import attribute extracting from the C-S route, and sends this Type 7 route to other PEs.

When sender PE receives a Type 7 route, if RT-Import of this route belongs to itself, it translates this Type 7 route back into a C-join message and sends it to its CE.

8.2.3.1.2. Shared Tree Join (C-*, C-G)

When receiver PE receives a shared tree join (C-*, C-G) from CE, it does a route look up for C-RP. If there is more than one route, the receiver PE chooses a single forwarder PE. The procedures used for choosing a single forwarder are outlined in [RFC6514].

When the C-RP route has been selected, the receiver PE will create a Type 6 route. If this PE has not received a Type 5 route, it will not advertise it.

When source connected to CE is active, register message is sent to the sender PE. The sender PE originates a Type 5 route, and sends to other MCAST-6PE routers.

When receiver PE receives the Type 5 route from the remote PE, it will originate a Type 7 route based on Type 5 and Type 6, then it sends the Type 7 route carrying Route Import attribute extracting from the C-RP route, and sends this Type 7 route to other PEs.

When sender PE receives the Type 7 routes, compares local RT-Import to RT received with Type 7 routes. If match, it imports the Type 7 routes, then translates the Type 7 route back into a C-join message and passes the C-join messages to CE.

8.2.3.2. mLDP as the C-Multicast Protocol

The construction of the MCAST-6PE NLRI of C-multicast routes for the case where the C-multicast protocol is mLDP [mLDP] is described in [RFC6514].

8.2.4. Propagating IPv6 C-Multicast Routes by an ASBR

The mechanisms for IPv6 C-Multicast Routes by an ASBR are the same as the MVPN case described in section 11.2 of [RFC6514].

8.3. Using 6PE S-PMSI A-D Routes to Bind C-Trees to P-Tunnels

BGP-based procedures for using 6PE S-PMSIs A-D routes to bind (C-S,C-G) trees to P-tunnels are the same as the MVPN case described in section 12 of [RFC6514].

9. IANA Considerations

This document defines a new BGP Extended Community called "Route Import" (Type value is to be assigned by IANA). This Community is IP address specific, of an extended type, and is transitive.

This document defines a new NLRI, called as MCAST-6PE NLRI, to be carried in BGP using multiprotocol extensions. It requires assignment of a new SAFI. This is to be assigned by IANA.

10. Security Considerations

This document raises no new security issues. Security considerations for the base protocol are covered in [RFC6513] and [RFC6514].

11. References

11.1. Normative References

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

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, February 2006.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.

[RFC4798] De Clercq, J., Ooms, D., Prevost, S., and F. Le Faucheur, "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, February 2007.

[RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.

[RFC6514] Aggarwal, R., Rosen, E., Morin, T., and Y. Rekhter, "BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs", RFC 6514, February 2012.

11.2. Informative References

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, January 2006.

[RFC4610] Farinacci, D. and Y. Cai, "Anycast-RP Using Protocol Independent Multicast (PIM)", RFC 4610, August 2006.

[RFC5331] Aggarwal, R., Rekhter, Y., and E. Rosen, "MPLS Upstream Label Assignment and Context-Specific Label Space", RFC 5331, August 2008.

[RFC6388] Wijnands, IJ., Minei, I., Kompella, K., and B. Thomas, "Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths", RFC 6388, November 2011.

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: lizhenbin@huawei.com

Shunwan Zhuang
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: zhuangshunwan@huawei.com

Hui Ni
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: nihui@huawei.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 15, 2015

C. Liu
Q. Sun
J. Wu
Tsinghua University
October 12, 2014

Dynamic IPv4 Provisioning for Lightweight 4over6
draft-liu-softwire-lw4over6-dhcp-deployment-05

Abstract

Lightweight 4over6 [I-D.ietf-softwire-lw4over6] is an IPv4 over IPv6 hub and spoke mechanism that provides overlay IPv4 services in an IPv6-only access network. Provisioning IPv4 addresses and port set to customers is the core function of Lightweight 4over6 control plane. [I-D.ietf-softwire-lw4over6] illustrates how to use DHCPv6 for deterministic IPv4 provisioning. This document discusses how to provision IPv4 parameters by using dynamic IPv4 provisioning protocols such as DHCPv4 over DHCPv6 [RFC7341]. This document describes a dynamic IPv4 provisioning mode for Lightweight 4over6 that uses DHCPv4 over DHCPv6 [RFC7341] for IPv4 address provisioning.

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 April 15, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Advantage of Dynamic IPv4 Provisioning 3
4. Using DHCPv4 over DHCPv6 for Lw4over6 Provisioning 4
4.1. IP Addressing 4
4.2. DHCPv6 Configuration 4
4.3. DHCPv4 over DHCPv6 Function 4
4.4. Port Set Consideration 5
4.5. lwAFTR Binding Table Maintenance 5
5. Security Considerations 6
6. IANA Considerations 6
7. References 6
7.1. Normative References 6
7.2. Informative References 7
Authors' Addresses 7

1. Introduction

Lightweight 4over6 [I-D.ietf-softwire-lw4over6] provides IPv4 access over IPv6 network in hub-and-spoke softwire architecture. In Lightweight 4over6, each Lightweight B4 (lwB4) is assigned with a port-restricted public IPv4 address or a full public IPv4 address to be used for IPv4 communication. Provisioning IPv4 address, port set and other IPv4 parameters to lwB4 is the core function of the Lightweight 4over6 control plane. It can be achieved by several protocols, such as DHCPv6 [RFC3315] [I-D.ietf-softwire-map-dhcp], DHCPv4 over DHCPv6 [RFC7341] , and PCP [RFC6887].

[I-D.ietf-softwire-lw4over6] illustrates how to use DHCPv6 for deterministic IPv4 provisioning. The IPv4 address and port set ID (PSID) are carried in DHCPv6 options defined in [I-D.ietf-softwire-map-dhcp]. However, the deterministic IPv4 provisioning adds some restrictions for addressing and deployment: the IPv4 address's life time needs to be bound to the IPv6 lease time; the IPv4 address and PSID need to be embedded into clients' /128 IPv6 address so the client can not use arbitrary /128 IPv6 address as tunnel source address; a customer network that is provisioned with a unique IPv6 prefix can only set up one tunnel instance.

This document describes how to deploy Lightweight 4over6 using DHCPv4 over DHCPv6 for dynamic IPv4 address provisioning. Since pure DHCPv4 is unable to directly work in native IPv6 network, DHCPv4 over DHCPv6 [RFC7341] is proposed to support DHCPv4 functionality in IPv6 network by transporting DHCPv4 messages over DHCPv6 message. [I-D.ietf-dhc-dynamic-shared-v4allocation] describes how to allocate port set to clients using DHCPv4 over DHCPv6. [I-D.fsc-softwire-dhcp4o6-saddr-opt] defines options for lwB4 to report its IPv6 tunnel source address to the server. This document does not define a new provisioning method, but describes how these existing specifications are organized to support IPv4 provisioning for Lightweight 4over6.

2. Terminology

Terminology defined in [RFC7341] and [I-D.ietf-softwire-lw4over6] is used extensively in this document.

3. Advantage of Dynamic IPv4 Provisioning

[I-D.ietf-softwire-lw4over6] describes the behavior of lwB4 and lwAFTR using DHCPv6 as provisioning protocol. It is based on a pre-determined binding relationship between IPv6 prefix and IPv4 address + PSID. With dynamic IPv4 provisioning, there is no restriction on how the lwB4's IPv6 address is generated. Since in the DHCPv4 over DHCPv6 process the lwB4 is able to tell the server which IPv6 address it intends to use, the lwB4 can run SLAAC, DHCPv6 or other mechanism to achieve and generate its IPv6 address that is used for IPv6 tunnel source address. It is different from the deterministic provisioning mode that IPv4 address are pre-binded to IPv6 prefix and multiple lwB4s sourced behind the same IPv4 prefix can not be supported, and generally lwB4 can not run SLAAC to generate its IPv6 address for tunnel.

From the IPv4 address life time view, dynamic IPv4 provisioning allows IPv4 address to have a independent IPv4 life time. This is helpful that the in some case the IPv4 provisioning server may not be able to know the lwB4's IPv6 address life time. It may be because that the IPv4 provisioning server may not also be the IPv6 provisioning server for the lwB4, or even the lwB4's IPv6 address does not have a life time at all, thus to bound the IPv4 address life time to IPv6 address life time may cause a waste of IPv4 addresses that the provisioning server is unable to recycle IPv4 address. The dynamic provisioning schema is suitable for operators that has restricted IPv4 address recourses.

4. Using DHCPv4 over DHCPv6 for Lw4over6 Provisioning

This section describes how DHCPv4 over DHCPv6 is used for Lightweight 4over6 configuration. In the remaining of this section, "lwB4" without explicitly written as "stateless lwB4" will refer to stateful lwB4 that runs DHCPv4 over DHCPv6 for dynamic IPv4 provisioning.

4.1. IP Addressing

Before starting DHCPv4 over DHCPv6 to achieve IPv4 configuration, lwB4 MUST be configured with an IPv6 address. There's no restrictions on how IPv6 address is provisioned. The configured IPv6 address is used for IPv6 tunneling and DHCPv4 over DHCPv6 process. The address that lwB4 chooses MUST be routable to the lwAFTR and DHCP 4o6 server, e.g. a link-local address must not be used.

The softwire provider is free to provide any IPv4 address for a lwB4. There's no restrictions on IPv6/IPv4 addressing, e.g. scattered IPv4 addresses can be used, and there's no need for embedding IPv4 address/PSID into IPv6 address.

4.2. DHCPv6 Configuration

Before stateful lwB4 runs DHCPv4 over DHCPv6 to acquire IPv4 address and port set, lwB4 MUST run DHCPv6 to achieve the DHCP 4o6 server's IPv6 address. The DHCPv6 server provides the DHCP 4o6 server's IPv6 address by OPTION_DHCP4_O_DHCP6_SERVER as defined in [RFC7341].

A stateful lwB4 may also be compatible with [I-D.ietf-softwire-map-dhcp] and thus will require both OPTION_DHCP4_O_DHCP6_SERVER and OPTION_S46_CONT_LW. The DHCPv6 server decides whether supply OPTION_S46_CONT_LW and OPTION_S46_V4V6BIND directly or indicate the client to run DHCPv4 over DHCPv6 by supplying OPTION_DHCP4_O_DHCP6_SERVER according to its policy. The lwB4 should implement a local logic to decide which one it prefers. The strategy of how to decide preferences between the provisioning modes is out of the scope of the document.

4.3. DHCPv4 over DHCPv6 Function

The DHCPv4 over DHCPv6 function in lwB4 is disabled by default, and enabled by OPTION_DHCP4_O_DHCP6_SERVER in DHCPv6 server's response. Once enabled, lwB4 runs stateful DHCPv4 over DHCPv6 to acquire IPv4 address and port set. lwB4 provides one of its IPv6 address as IPv6 tunnel source address to the DHCP 4o6 server, and get the lwAFTR's tunnel address through DHCPv4 over DHCPv6. The DHCPv4 over DHCPv6 message flow is described in section 4 of [I-D.fsc-softwire-dhcp4o6-saddr-opt] and MUST be followed.

4.4. Port Set Consideration

lwB4 gets its PSID through DHCPv4 over DHCPv6 along with its IPv4 address. [I-D.ietf-dhc-dynamic-shared-v4allocation] describes how to provision PSID to lwB4 through DHCPv4 over DHCPv6.

When sending a DHCPDISCOVER over DHCPv6 message, lwB4 MUST include OPTION_V4_PORTPARAMS in the Parameter Request List. If the server decides to reply a port-restricted address, it MUST reply OPTION_V4_PORTPARAMS to lwB4. if the server decides to reply a full IPv4 address, it SHOULD NOT reply OPTION_V4_PORTPARAMS in the response. When lwB4 receives DHCPv4 over DHCPv6 response without OPTION_V4_PORTPARAMS, it configures itself with the full IPv4 address as regular DHCPv4 client does. When lwB4 receives a shared IPv4 address, the address is used for NAPT and MUST NOT be used to identify the lwB4.

4.5. lwAFTR Binding Table Maintenance

lwAFTR maintains its binding table as per section 6.1 of [I-D.ietf-softwire-lw4over6]. Unless the binding table is fixed and pre-determined, it is synchronized with DHCPv4 over DHCPv6 process. The following DHCPv4 over DHCPv6 messages triggers binding table modification:

- DHCPACK: Generated by DHCP server, triggers lwAFTR to add a new entry or modify an existing entry.

- DHCPRELEASE: Generated by lwB4, triggers lwAFTR to delete an existing entry.

When lwAFTR receives a DHCPACK event, it looks up the binding table using the lwB4’s IPv4 address and PSID as index. If there is an existing entry found, the lwAFTR updates the IPv6 address and lifetime fields of the entry; otherwise the lwAFTR creates a new entry accordingly. When lwAFTR receives a DHCPRELEASE event, it looks up the binding table using the lwB4’s IPv6 address, IPv4 address and PSID as index. The lwAFTR deletes the entry either by removing it from the binding table or mark the lifetime field to an invalid value (e.g. 0).

When lwAFTR is co-located with the DHCP server, it listens all DHCPv4 over DHCPv6 messages generated or received by the DHCP server and updates the bindings through valid messages. When lwAFTR is not co-located with the DHCP server, the DHCP server informs the lwAFTR about the binding updates through other protocols. DHCP active lease query [I-D.ietf-dhc-dhcpv4-active-leasequery] [I-D.ietf-dhc-dhcpv4-active-leasequery] could be used to do this.

The lwAFTR works as a requestor to get every lwB4's IPv4 address + PSID (from DHCPv4 lease), and IPv6 address (from DHCPv6 option). Since current DHCPv4 active lease query doesn't support carrying DHCPv6 options, and DHCPv6 active lease query doesn't support carrying DHCPv4 lease information, it may require extensions to current DHCPv4/DHCPv6 active lease protocols but out of the scope of this document.

5. Security Considerations

Security considerations in [I-D.ietf-softwire-lw4over6] and [RFC7341] should be considered.

The DHCP message triggered binding table maintenance may be used by an attacker to send faked DHCP messages to lwAFTR. The operator network should deploy [RFC2827] to prevent this kind of attack.

6. IANA Considerations

This document does not include an IANA request.

7. References

7.1. Normative References

[I-D.fsc-softwire-dhcp4o6-saddr-opt]
Farrer, I., Sun, Q., and Y. Cui, "DHCPv4 over DHCPv6 Source Address Option", draft-fsc-softwire-dhcp4o6-saddr-opt-01 (work in progress), September 2014.

[I-D.ietf-dhc-dynamic-shared-v4allocation]
Cui, Y., Qiong, Q., Farrer, I., Lee, Y., Sun, Q., and M. Boucadair, "Dynamic Allocation of Shared IPv4 Addresses", draft-ietf-dhc-dynamic-shared-v4allocation-02 (work in progress), September 2014.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Qiong, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-10 (work in progress), June 2014.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC7341] Sun, Q., Cui, Y., Siodelski, M., Krishnan, S., and I. Farrer, "DHCPv4-over-DHCPv6 (DHCP 4o6) Transport", RFC 7341, August 2014.

7.2. Informative References

[I-D.ietf-dhc-dhcpv4-active-leasequery]
Kinnear, K., Stapp, M., Volz, B., and N. Russell, "Active DHCPv4 Lease Query", draft-ietf-dhc-dhcpv4-active-leasequery-01 (work in progress), June 2014.

[I-D.ietf-dhc-dhcpv6-active-leasequery]
Dushyant, D., Kinnear, K., and D. Kukrety, "DHCPv6 Active Leasequery", draft-ietf-dhc-dhcpv6-active-leasequery-01 (work in progress), March 2014.

[I-D.ietf-softwire-map-dhcp]
Mrugalski, T., Troan, O., Farrer, I., Perreault, S., 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-09 (work in progress), October 2014.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

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

Authors' Addresses

Cong Liu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: gnocuil@gmail.com

Qi Sun
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: sunqi@csnet1.cs.tsinghua.edu.cn

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

Phone: +86-10-6278-5983
Email: jianping@cernet.edu.cn

softwire
Internet-Draft
Intended status: Informational
Expires: April 16, 2016

R. Maglione, Ed.
W. Dec
Cisco Systems
I. Leung
Rogers Communications
E. Mallette
Bright House Networks

October 14, 2015

Use cases for MAP-T
draft-maglione-softwire-map-t-scenarios-06

Abstract

The Softwire working group standardized both encapsulation and translation based stateless IPv4/IPv6 solutions in order to be able to provide IPv4 connectivity to customers in an IPv6-Only environment.

The purpose of this document is to describe some operational use cases that would benefit from a translation based approach and highlights the operational benefits that a translation based solution would allow.

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 April 16, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Operational Service Policy Use Cases 3
2.1. Network/Transport Layer Classification classifiers . . . 4
2.2. Device Configuration (DOCSIS) 5
2.3. Service Flow management using Deep Packet Inspection . . 5
2.4. Service Flow Redirection Policies (Web-redirection) . . . 6
2.5. Service Flow Caching 7
3. Technological Considerations 7
3.1. Encapsulation and Translation Overhead 8
3.2. Efficient Utilization of the Access Network 8
3.2.1. Jumbo Frame Support in the Access 8
3.2.2. Operator Added Packet Overhead and Service Level Agreements 9
4. Conclusions 10
5. Acknowledgements 10
6. IANA Considerations 10
7. Security Considerations 10
8. Informative References 10
Authors' Addresses 10

1. Introduction

The Softwire working group standardized both encapsulation [RFC7597] and translation [RFC7599] based stateless IPv4/IPv6 solutions developed for the purposes of offering IPv4 connectivity to the customers in an IPv6-Only environment.

There are deployment scenarios that may benefit equally from an encapsulated or translated form of an IPv4/IPv6 stateless addressing solution. There are, however, use cases where using a translation approach could lead to significant operational benefits and potential savings for the operators.

This document describes some use cases that would take advantage of a translation based solution, by highlighting the operational benefits that a translation based approach would allow.

2. Operational Service Policy Use Cases

In Broadband Networks it is common practice for Operators to apply per-subscriber policies on subscriber traffic at the network edge such as a BNG (Broadband Network Gateway), CMTS (Cable Modem Termination System), PGW (PDN Gateway) or like device. Various services may require the application of different policies at these services edges.

Typically a policy would include a classification function and an action function.

- Service flow classification may occur based on any combination of the following:
  - Layer-3 identifiers such as source, destination address, protocol or next header, DSCP or Traffic Class;
  - Layer-4 identifiers such as source or destination port;
  - service type/destination (i.e. Internet, network service, or other service)
- Actions may be provisioned against the classified traffic; the following are some examples of actions:
  - application of different QoS treatment (could be rate-limit, drop, redirect,.. etc) based on Layer 3 or higher layer (Layer 4-7) classification from devices like deep packet inspection appliances;
  - Service flow redirection on selected types of traffic (i.e. Web portal);
  - Service flow caching on selected types of traffic.

The rationale for applying such policy at the network edge is based on how tightly coupled this layer of the network is with many key systems within the operators network such as RADIUS, DHCP, access technology awareness and ability to implement subscriber awareness.

In many common deployments today, the customer’s policies are maintained in RADIUS server or enforced through other provisioned data in co-operation with service activation such as DHCP and bootstrap configuration. In a cable operator network, while much of the heavily lifting of subscriber management is embedded on the CMTS or OLT, the reality is that classification is shared across CMTS and cable-modem (CM) or across OLT and optical network unit (ONU.) The

CM and ONU classification capabilities are not as robust and flexible as the upstream CMTS, OLT and/or assisting edge router. The implications of that are that the CPE may need to be replaced with a device that has the capability to classify on a larger packet header.

An additional point to consider is that the edge network nodes are also often fitted with, or co-located with higher functioning appliances that employ Deep Packet Inspection and distributed caches used to enhance service performance.

2.1. Network/Transport Layer Classification classifiers

Most of the policies described in Section 2 require the use of network and transport layer classification and filtering mechanisms such as classifiers at the network edge. The application of classifiers and other network layer classification functions on selected subscriber flows are often applied by a AAA server, gleaned from configuration information, provisioned from per-CM DOCSIS configuration files generated from the operator OSS, or sent by a policy control function (PCRF, PCMM, etc).

This section will explain why the application of some types of classifiers (like Layer 3 destination based classifiers and - Layer 3 plus Layer 4 - classifiers,) can be deployed in a more simplistic fashion when using a translated form of a stateless IPv4/IPv6 transition technology such as MAP-T [RFC7599].

A key characteristic of MAP-T is the mapping of the IPv4 address of any destination into the IPv6 destination address, by means of IPv4 to IPv6 mapping rules. This mapping means that the subscriber flows are native IPv6 flows within the operators network. The ability to use a standard IPv6 classifier to identify interesting traffic for classification is well aligned with traditional traffic identification capabilities using IPv4 based classifiers. Such classifiers can be easily applied at the access edge as a standard function commonly available on most platforms deployed.

In contrast, a solution utilizing an IP tunnel based transport (MAP-E [RFC7597] or DS-Lite [RFC6333]), effectively hides the payload’s IP layer information, making it difficult to identify by means of an IPv6 classifier . The operator in the latter case (tunneled option) would need additional functionality to classify the same subscriber flows which may not be available on the deployed platforms.

The classifier use case is further extended when considering that many traffic classifications are made using transport layer (Layer 4) information. This is common in operator networks that often apply differential traffic treatment to different services that typically

operate using well defined TCP/UDP ports. In the MAP-T deployment case, these ports are available for classification matching using the same standard access edge node capabilities using IPv6 classifiers. In the case where tunneled forms of a solution are used, these higher layer ports are hidden from the network (base IP layer) and special functionality to correctly classify these service flows is required.

The ability to apply classifiers at the access edge node allows the operator to not only use standard IPv6 classifier functionality, but also use same mechanisms (RADIUS interface parameters/system, or DOCSIS configuration classifier parameters) for applying such classifiers. I.e. custom RADIUS interface extensions or custom DOCSIS classifier extensions to deal with the classifier semantics of an IP tunnel based transport are not required.

2.2. Device Configuration (DOCSIS)

Some access technologies, like DOCSIS, require a modem configuration file for network operation. These configuration files often contain access control and classification information that uses IPv4 and/or IPv6 network and transport layer information.

MAP-T allows use of standard IPv6 classifiers within these configuration files permitting the continued use of the well-known service architecture. Translation technologies which use tunneling may require the operator to update how services are managed as information needed to enforce policy is not longer viewable by the Cable Modem or upstream CMTS. The operator in this case may need to build new service capabilities higher up in the network after the network translator to apply the full range of polices for the subscriber base.

2.3. Service Flow management using Deep Packet Inspection

Several Service Providers today use Deep Packet Inspection devices located at the network edge (such as a BNG) in order to inspect the subscriber’s traffic for different purposes: profiling the user’s behavior, and classifying the traffic based on higher layer information and/or traffic signatures.

Deep packet inspection devices available today in the market and, more importantly, those already deployed in operator’s network may not be able to analyze encapsulated traffic, like IPinIP, and to correlate the inner packet’s contents to the outer packet’s "subscriber" context - this limitation is consistent across multiple vendors. In order to overcome this limitation when using IP tunnel based transports, without resorting to costly network upgrades, dedicated DPI devices need to be applied at a point in the network

where the IP tunnel transport has been stripped and the payload is directly available for native processing. This not only changes the network architecture, but it increases the number of DPI's devices required: one for IPv6 traffic at the access edge, the other at a location where the IPv4 traffic is exposed (typically a separate Location). In addition the operator would need to enforce policies at two architecturally separate places in the network. Furthermore, even with these changes enacted, there remains a critical problem of correlating traffic to a given subscriber: in encapsulation based solutions, the IPv4 address information in the payload is not sufficient to uniquely identify a subscriber given that an IPv4 address will not be unique. As such, additional mechanisms and changes to the accounting infrastructure need to be introduced which when combined with all the previous aspects makes this solution operationally complex.

With MAP-T operators can continue using the current architectural model with DPI devices installed at the access edge; the only requirement would be to have the same device able to recognize specific applications on the native IPv6 transport, which DPI devices based on application signatures are capable of doing. Thus with MAP-T it doesn't matter that an operator might provision the same IPv4 address across multiple subscribers. In addition with MAP-T the access edge would remain the single enforcement point for all user's policies for all traffic. This would allow the operators to continue using a consistent architecture and set of accounting tools for their network.

2.4. Service Flow Redirection Policies (Web-redirection)

Redirecting the user's traffic to web portal is a common practice in Service Provider networks. For example, it is common for operators to inform users about new services, service advisories and/or access to account changes using web-reduction techniques activated on http traffic. In current deployments web-redirection occurs at the Edge node level, where the subscriber's traffic first hits the IP network. The activation/de- activation of redirection policy on selected subscribers may be driven by the AAA/RADIUS through specific RADIUS attributes. In current deployments web-redirection occurs at the Edge node level, where the subscriber's traffic first hits the IP network. The activation/de- activation of redirection policy on selected subscribers may be driven by the AAA/RADIUS through specific RADIUS attributes.

If MAP-T is used the redirection of both IPv6 and IPv4 traffic can be kept at the Edge of the network with the same configuration currently used and by simply translating the Server's address in IPv6 with known mapping rules. In case of tunnel based solution the

redirection of IPv6 and IPv4 cannot occur in a single place, because the redirection of IPv4 traffic must be implemented at or after the v4/v6 gateway responsible for de-encapsulating the traffic. This approach not only would require deploying two separate infrastructures located in different places in order to achieve the redirection for both IPv6 and IPv4 traffic, but also it would not allow continuing using the AAA/RADIUS Server infrastructure in order to enforce the redirect policy at the subscriber's session.

2.5. Service Flow Caching

With the continuing growing of video traffic, especially considering the increase of http video traffic (YouTube like,) it is useful for the Service Providers to be able to cache the video stream at the Edge of the network in order to save bandwidth on upstream links. Using cache devices together with tunnel solutions would introduce similar challenges/issues as the ones described for DPI scenarios, in particular it would require applying caching functionality after the decapsulation point. Obviously this would not eliminate the benefits of the cache. Instead a MAP-T approach would allow caching the subscriber traffic at the edge of the network and gaining the bandwidth savings introduced by the caching. Crucially, any native IPv6 web-caches would be capable of processing IPv6 MAP-T traffic as fully native traffic.

In addition in some deployments today, Web Cache Control Protocol (WCCP) feature is used in order to redirect subscriber's traffic to the cache devices. When a subscriber requests a page from a web server (located in the Internet, in this case), the network node where the WCCP is active, sends the request to a Cache Engine. If the cache engine has a copy of the requested page in storage, the engine sends the user that page. Otherwise, the engine gets the requested page and the objects on that page from the web server, stores a copy of the page and its objects (caches them), and forwards the page and objects to the user. WCCP is another example of web redirect thus, the same considerations described in section Section 2.4 and the benefits introduced by MAP-T also apply here.

3. Technological Considerations

There are additional technological considerations which need to be analyzed by the operator when choosing which transition technology option they would like to deploy. This section describes some of those considerations.

3.1. Encapsulation and Translation Overhead

MAP-E adds an encapsulation tax of 40 bytes, while MAP-T adds a translation tax of 20 bytes (translating from a 20-byte IPv4 header to a 40-byte IPv6 header.) In the downstream direction (from network toward the CPE), with an average packet size of 1000-1100 bytes, the added encapsulation is under 4% in the case of MAP-E. In the case of MAP-T that encapsulation tax drops to about 2%.

In the upstream direction, with an average packet size of ~400 bytes, the effects of the encapsulation tax is more pronounced with an added 10% overhead for MAP-E and 5% additional overhead for MAP-T. As the upstream direction tends to be both (a) more heavily oversubscribed than is the downstream and (b) of lower performance, the greater the header tax the more it upsets the precariously balanced upstream/ downstream network loading models.

3.2. Efficient Utilization of the Access Network

Point-to-Multipoint access networks are common across network operators - DOCSIS (1.0, 1.1, 2.0, 3.0), EPON, 10G-EPON, GPON, XGPON,XGPON2, etc. This network type has been incredibly successful, as attested to by all the variants of point-to-multipoint networks deployed, primarily because of their cost effectiveness.

There are a couple challenges that are introduced by adding a significant amount of encapsulation overhead. These challenges affect MAP-T and MAP-E similarly; the effects from MAP-E are simply more pronounced.

The first challenge is that, commonly, point-to-multipoint networks have limited support for jumbo frames. The second challenge is one that results in reduction in effective capacity on the wire, which yields higher cost.

3.2.1. Jumbo Frame Support in the Access

Some access technologies natively support fragmentation, and as a result, can support "jumbo frames" up to a point. A max size IPv4 packet that fits into the payload of a standard-compliant Ethernet frame is 1500 bytes. In the context of this discussion a "jumbo frame" is any Ethernet frame that has more than 1500 bytes in the Ethernet payload. IEEE Std. 802.3 now specifies a larger frame size of up to 2000 bytes, referred to as an envelope frame, where the envelope frame, quoting from IEEE Std.802.3-2012 "is intended to allow inclusion of additional prefixes and suffixes required by higher layer encapsulation protocols. The encapsulation protocols may use up to 482 octets."

In the network access space, particularly one filled with legacy access products which may be 10 years old (or perhaps older), it is not uncommon to find products that just only support a max 1500 byte Ethernet payload. Some may support up to 1532 byte payload (1550 byte Ethernet frame), some 1582 byte payload (1600 byte Ethernet frame), though there's certainly not a uniform supported frame size past the 1500 byte payload.

Since MTU discovery isn't typically used for IPv4 in operator networks and since MTU discovery for IPv6 is not implemented on the IPv4 host stack requesting the communication, there's no effective way to tell the host stack to reduce the size of its IPv4 frame to accommodate the MAP-T or MAP-E overhead with the MTU frame size limitation of the specific access products. There are tools like Maximum Segment Size rewrite that can be implemented to help address the issue for a TCP payload but UDP payload will continue to be impaired.

Thus MAP-T is preferred as there are more deployed access products that could support a 1534-byte or 1538-byte Ethernet frame than can support a 1554-byte or 1558-byte Ethernet frame, which mandates fewer access product replacements.

3.2.2. Operator Added Packet Overhead and Service Level Agreements

One of the traditional challenges with adding additional packet overhead to a customer frame is that it becomes more challenging to provide customer the last-mile bandwidth in their SLA. This is a very simple overprovisioning problem when the maximum size frame is used, as the overhead in that case is a fixed ~1.5% or ~3% for MAP-T and MAP-E respectively.

However in the case of variable packet sizes, the added overhead from either MAP-T or MAP-E can become very significant - from a worse case of ~31% (MAP-T) and ~63% (MAP-E) to the ~1.5% or ~3%. This means that to provide the customer what they purchased operators will either provision more than the customer SLA to account for the added overhead or abide by the "not guaranteed" bandwidth response.

With the average upstream packet sizes being smaller, the 5% (MAP-T) or 10% (MAP-E) added overhead for the average upstream packet size could find itself in an overprovisioned QoS profile.

Many customers, particularly business customers, are very savvy and have a strong belief that when a network operator offers them an SLA, it's not an SLA at a specific packet size. This can be a significant operational difficulty for network operators, one with a real operational cost.

4. Conclusions

The use cases described in this document have highlighted a clear need for a MAP-T solution based on Service Providers' operational requirements.

This document showed that a MAP-T approach is not a duplication of any other existing IPv4/IPv6 migration mechanisms based on IP tunneling, but actually has capabilities to solve Service Provider's problems.

5. Acknowledgements

The authors would like to thank Victor Kuarsingh for his valuable comments and inputs to this document.

6. IANA Considerations

This document does not require any action from IANA.

7. Security Considerations

This document has no additional security considerations beyond those already identified in section 11 of [RFC7599]

8. Informative References

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <http://www.rfc-editor.org/info/rfc6333>.

[RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", RFC 7597, DOI 10.17487/RFC7597, July 2015, <http://www.rfc-editor.org/info/rfc7597>.

[RFC7599] Li, X., Bao, C., Dec, W., Ed., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", RFC 7599, DOI 10.17487/RFC7599, July 2015, <http://www.rfc-editor.org/info/rfc7599>.

Authors' Addresses

Roberta Maglione (editor)
Cisco Systems
Via Torri Bianche 8
Vimercate 20871
Italy

Email: robmgl@cisco.com

Wojciech Dec
Cisco Systems
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands

Email: wdec@cisco.com

Ida Leung
Rogers Communications
8200 Dixie Road
Brampton, ON L6T 0C1
CANADA

Email: Ida.Leung@rci.rogers.com

Edwin Mallette
Bright House Networks
4145 S. Faulkenburg Road
Riverview, Florida 33578
USA

Email: edwin.mallette@gmail.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 24, 2016

N. Matsuhira
Fujitsu Limited

July 23, 2015

SA46T Prefix Resolution (SA46T-PR)
draft-matsuhira-sa46t-pr-spec-05

Abstract

This document specifies SA46T Prefix Resolution (SA46T-PR) specification. SA46T-PR is almost same as SA46T, however method of generation of outer IPv6 address is different. SA46T is backbone network based approach, however SA46T-PR is stub network based approch.

Requirements Language

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 RFC 2119 [RFC2119].

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 January 24, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Basic Network Configuration 3
3. Basic Function of SA46T-PR 4
3.1. IPv4 over IPv6 Encapsulation / Decapsulation 4
3.2. SA46T-PR Address Format 5
3.3. Resolving SA46T-PR address 6
4. Mode of SA46T-PR 7
4.1. Router mode 7
4.2. Host mode 7
5. Sample configuration 7
6. Comparison with SA46T 9
6.1. difference with SA46T 9
6.2. Compatibility with SA46T 9
7. IANA Considerations 9
8. Security Considerations 9
9. Acknowledgements 9
10. References 9
10.1. Normative References 9
10.2. Informative References 10
Author's Address 10

1. Introduction

This document provide SA46T Prefix Resolution (SA46T-PR) specification.

The basic strategy for IPv6 deployment is dual stack. However, because of exhaustion of IPv4 address, there will be no IPv4 addresses for configuring dual stack in near future. That means there will be IPv6 only networks automatically.

However, there are many IPv4 only networks still exist and those seems continuous use in near future. That means methods continuous use of IPv4 network over IPv6 only network will be required.

SA46T [I-D.draft-matsuhira-sa46t-spec] provide such methods. In addition, SA46T-PR also provide such methots. SA46T is backbone network based approach, on the other hand, SA46T-PR is stub network based approach.

2. Basic Network Configuration

Figure 1 shows network configuration with SA46T-PR. The network consists of three parts, backbone network, stub network, and SA46T-PR.

Backbone network can be operated with IPv6 only. Stub network has three cases, IPv4 only, Dual Stack (both IPv4 and IPv6), and IPv6 only.

SA46T connects backbone network and stub network in case IPv4 still works in that stub network. If stub network is IPv6 only, SA46T-PR is not needed.

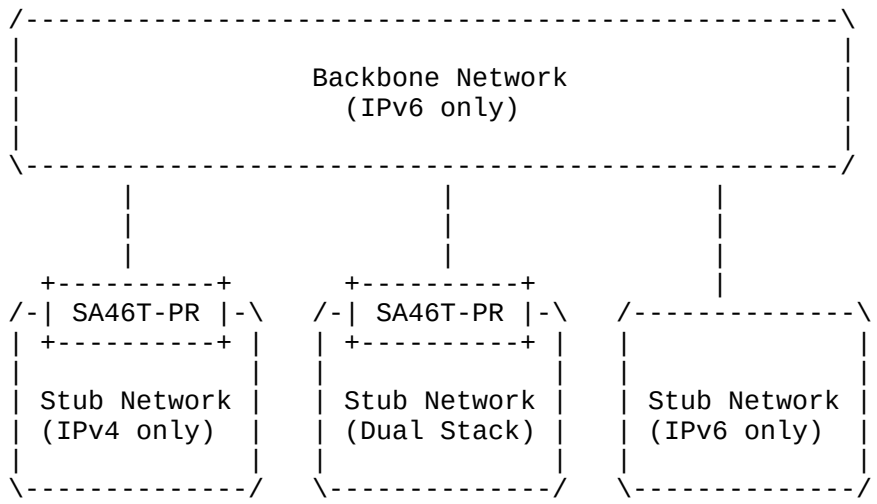


Figure 1

3. Basic Function of SA46T-PR

SA46T-PR has mainly two function. One is IPv4 over IPv6 Encapsulation / Decapsulation, and another is generate a table where IPv4 stub network belong to IPv6 network.

3.1. IPv4 over IPv6 Encapsulation / Decapsulation

SA46T-PR excapsulates IPv4 packet to IPv6 from stub network to backbone network, and decapsulates IPv6 packet to IPv4 from backbone network to stub network. Figure 2 shows packet format on both backbone network and stub network.

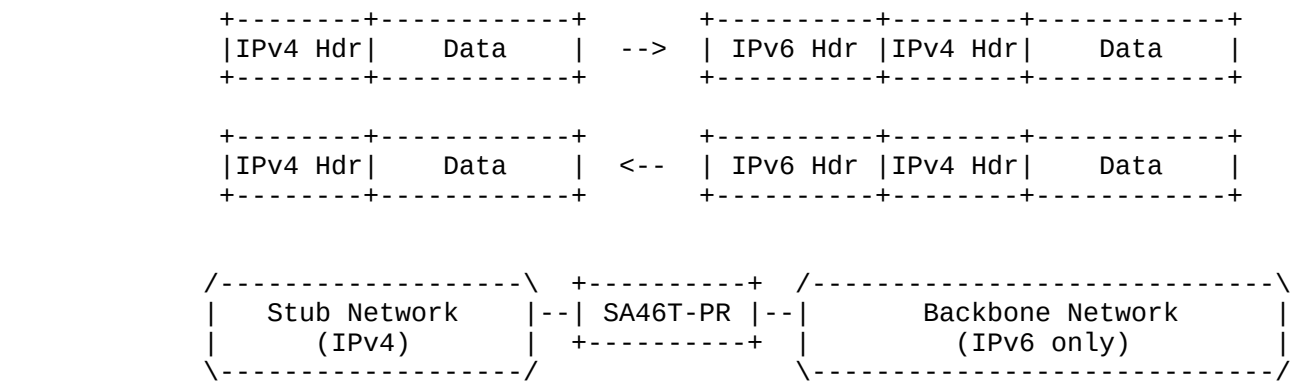


Figure 2

3.2. SA46T-PR Address Format

SA46T-PR address is a IPv6 address used in outer IPv6 header which encapsulate IPv4 packet by SA46T-PR. Figure 3 shows SA46T-PR address format.

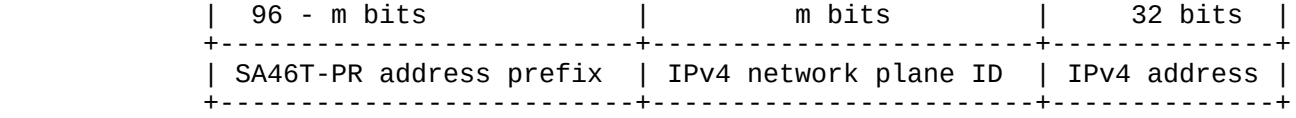


Figure 3

SA46T address consists of three parts as follows.

SA46T-PR address prefix

SA46T-PR address prefix is the IPv6 network prefix of stub network which contain IPv4 network of the IPv4 network plane.

IPv4 network plane ID

IPv4 network plane ID is an identifier of IPv4 network stack over IPv6 backbone network.

IPv4 address

IPv4 address in inner IPv4 packet.

3.3. Resolving SA46T-PR address

SA46T-PR resolve SA46T-PR address using SA46T Prefix Resolution Table (SA46T-PR Table). SA46T-PR generate SA46T-PR address resolving SA46T-PR prefix from IPv4 network plane ID and IPv4 address. FigureFigure 4 show this processing.

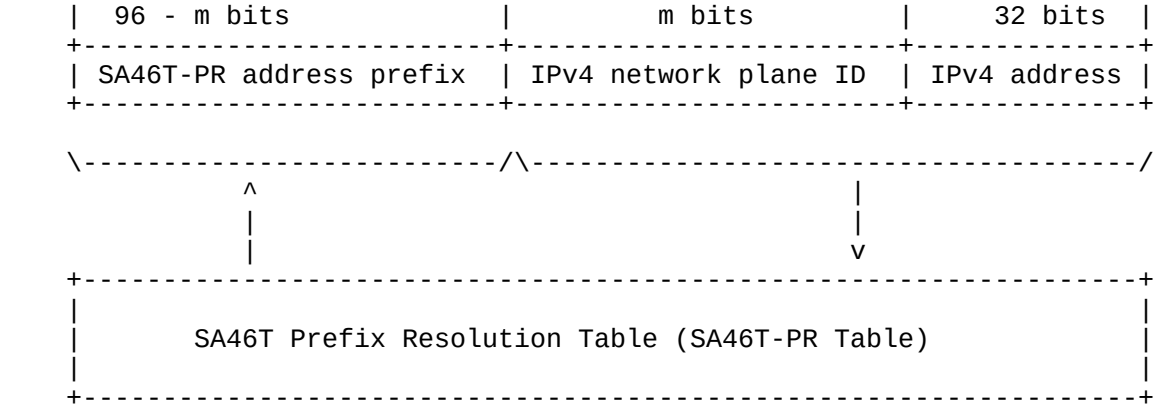


Figure 4

Figure Figure 5show SA46T-PR Table. This table consists four parts, IPv4 network plane ID, IPv4 address, netmask, and SA46T-PR address prefix.

Figure 5

SA46T-PR configured IPv4 network plane ID, so SA46T-PR know IPv4 network plane ID value the interface belongs.

Resolving destination address, SA46T-PR use pre-configured IPv4

network plane ID valude, and destination address of IPv4 packets, and serch the SA46T-PR table. SA46T-PR table return the SA46T-PR address prefix value correspoiding IPv4 network plane ID and IPv4 destination address. Then SA46T-PR generate whole SA46T-PR address.

Resolving source address, SA46T-PR already know IPv4 network plane ID value and IPv6 address prefix as SA46T-PR prefix. So, searching the SA46T-PR table does not require for resolving source address.

4. Mode of SA46T-PR

SA46T-PR has two working mode, one is router mode, another is host mode.

4.1. Router mode

In router mode, SA46T-PR act as a IPv6 router. SA46T-PR occupy IPv6 subnet, and SA46T-PR advertise route for SA46T-PR.

4.2. Host mode

In host mode, SA46T-PR act as a IPv6 host. SA46T-PR share IPv4 subnet, that mean, SA46T-PR and IPv6 hosts exists on same IPv6 subnet. SA46T-PR do proxy NDP function for IPv4 host.

5. Sample configuration

Figure Figure 6shows sample confiuration of SA46T-PR. In this example, there are three IPv4 stub network with the same IPv4 network plane.

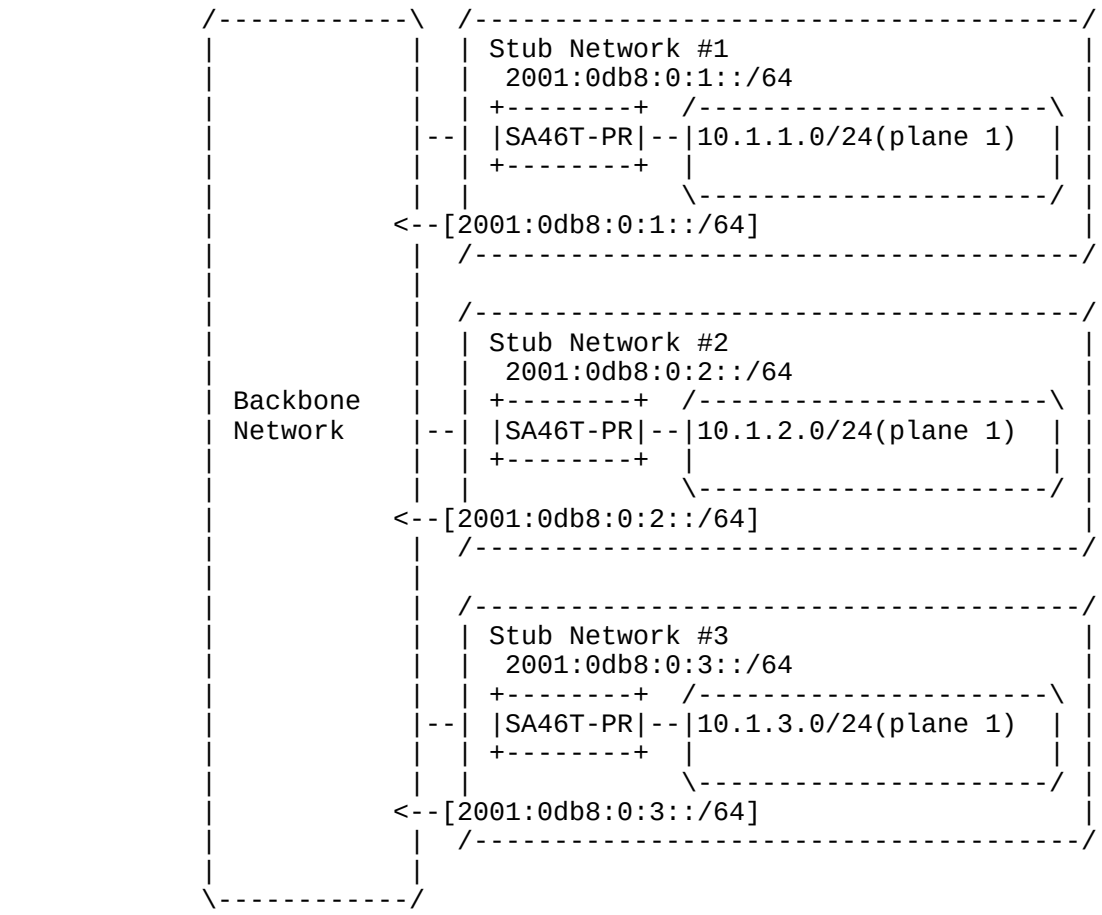


Figure 6

Figure Figure 7 shows SA46T-PR table for sample network.

IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
1	10.1.1.0	/120	2001:0db8:0:1
1	10.1.2.0	/120	2001:0db8:0:2
1	10.1.3.0	/120	2001:0db8:0:3

Figure 7

6. Comparison with SA46T

SA46T is backbone network based approach, and SA46T-PR is stub network based approach.

6.1. difference with SA46T

SA46T require route advertisement of SA46T prefix, so additional route are require, however configuration is few. On the other hand, SA46T-PR does not require additional route, however SA46T-PR table is require.

There are such trade-off between SA46T and SA46T-PR.

6.2. Compatibility with SA46T

If configure SA46t-PR table with default prefix as SA46T prefix, SA46T-PR acts as SA46T. In this case, netmask value of SA46T-PR table is /0, that mean any IPv4 network plane ID and IPv4 address pair match this entry.

7. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

8. Security Considerations

Security Considerations does not discussed in this memo.

9. Acknowledgements

10. References

10.1. Normative References

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

10.2. Informative References

[I-D.draft-matsuhira-sa46t-spec]
Matsuhira, N., "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology: Specification", January 2014.

Author's Address

Naoki Matsuhira
Fujitsu Limited
1-1, Kamikodanaka 4-chome, Nakahara-ku
Kawasaki, 211-8588
Japan

Phone: +81-44-754-3466
Fax:
Email: matsuhira@jp.fujitsu.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 24, 2016

N. Matsuhira
Fujitsu Limited

July 23, 2015

SA46T Prefix Translator (SA46T-PT)
draft-matsuhira-sa46t-pt-spec-05

Abstract

This document specifies SA46T Prefix Translator (SA46T-PT) specification. SA46T-PT expand IPv4 network plane by connecting SA46T domain and SA46T-PR domain. SA46T-PT translate prefix part of SA46T address and SA46T-PR address both are IPv6 address. SA46T-PT does not translate IPv4 packet which is encapsulated, so transparency of IPv4 packet is not broken.

Requirements Language

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 RFC 2119 [RFC2119].

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 January 24, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents

(http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Basic Network Configuration 3
3. Basic function of SA46T-PT 4
3.1. Translation processing 5
3.2. Address format of SA46T and SA46T-PR 5
3.3. Resolving translate prefix 6
3.4. Destination address resolution 6
3.5. Source address resolution 7
4. Sample Configuration 7
5. IANA Considerations 9
6. Security Considerations 9
7. Acknowledgements 9
8. References 10
8.1. Normative References 10
8.2. Informative References 10
Author's Address 10

1. Introduction

This document provide SA46T Prefix Translator (SA46T-PT) specification.

The basic strategy for IPv6 deployment is dual stack. However, because of exhaustion of IPv4 address, there will be no IPv4 addresses for configuring dual stack in near future. That means there will be IPv6 only networks automatically.

However, there are many IPv4 only networks still exist and those seems continuous use in near future. That means methods continuous use of IPv4 network over IPv6 only network will be required.

SA46T [I-D.draft-matsuhira-sa46t-spec] provide such methods. In addition, SA46T-PR [I-D.draft-matsuhira-sa46t-pr-spec] also provide such methots. SA46T is backbone network based approach, on the other hand, SA46T-PR is stub network based approach.

SA46T-PT expand IPv4 network plane by connecting SA46T domain and SA46T-PR domain. SA46T-PT translate prefix part of SA46T address and SA46T-PR address both are IPv6 address. SA46T-PT does not translate IPv4 packet which is encapsulated, so transparency of IPv4 packet is not broken.

2. Basic Network Configuration

Figure 1 shows network configuration with SA46T-PT. At large view, the network consists three parts, SA46T domain, SA46T-PR domain, and SA46T-PT. SA46T-PT connect SA46T domain and SA46T-PR domain.

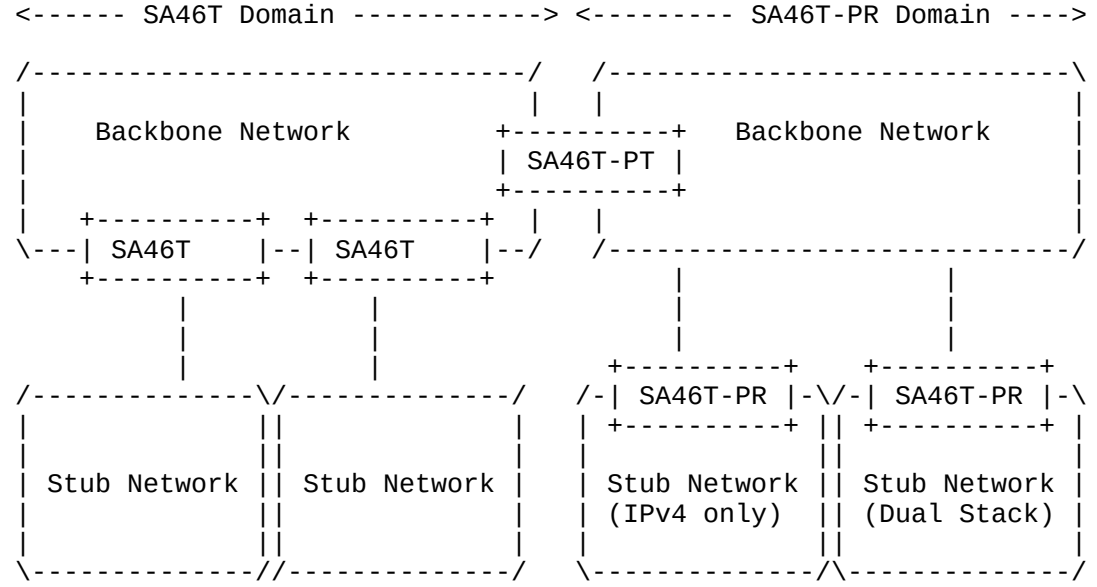


Figure 1

SA46T domain consists three parts, backbone network, stub network and SA46T. Backbone network can be operated with IPv6 only. Stub network has three cases, IPv4 only, Dual Stack (both IPv4 and IPv6), and IPv6 only.SA46T connects backbone network and stub network in case IPv4 still works in that stub network. If stub network is IPv6 only, SA46T is not needed. SA46T is a backbone network based approach, that mean SA46T advertise special route for SA46T.

And also, SA46T-PR domain consists three parts, backbone network, stub network and SA46T. Backbone network can be operated with IPv6 only. Stub network has three cases, IPv4 only, Dual Stack (both IPv4 and IPv6), and IPv6 only.SA46T connects backbone network and stub network in case IPv4 still works in that stub network. If stub network is IPv6 only, SA46T-PR is not needed. SA46T-PR is a stub network based approach.

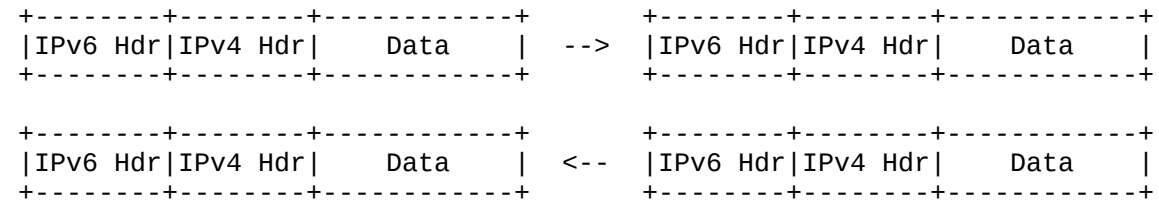
3. Basic function of SA46T-PT

This section describe basic function of SA46T-PT.

3.1. Translation processing

SA46T-PT translate between SA46T packet and SA46T-PT packet. SA46T packet and SA46T-PT packet are almost the same, however IPv6 address are different.

Fig shows packet format of SA46T domain and SA46T-PT domain.



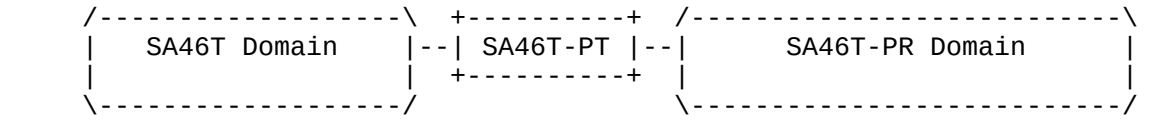


Figure 2

3.2. Address format of SA46T and SA46T-PR

figure Figure 3 shows SA46T address format and Figure 4 shows SA46T-PR address format. These format almost the same except SA46T address prefix in SA46T address and SA46T-PR address prefix in SA46T-PR address.

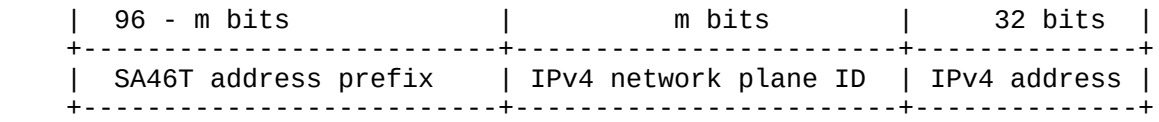


Figure 3

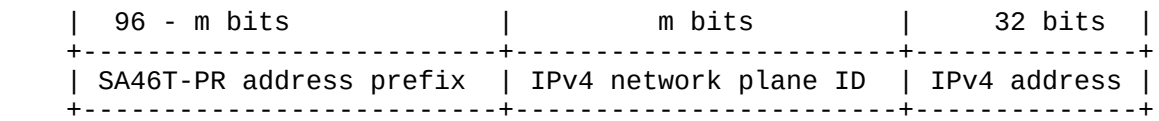


Figure 4

3.3. Resolving translate prefix

SA46T-PT translate from SA46T prefix to SA46T-PR prefix, or from SA46T-PR prefix to SA46T prefix using SA46T Prefix Translation (SA46T-PT) table. fig Figure 5shows address resolution manner and fig Figure 6shows SA46T-PT table.

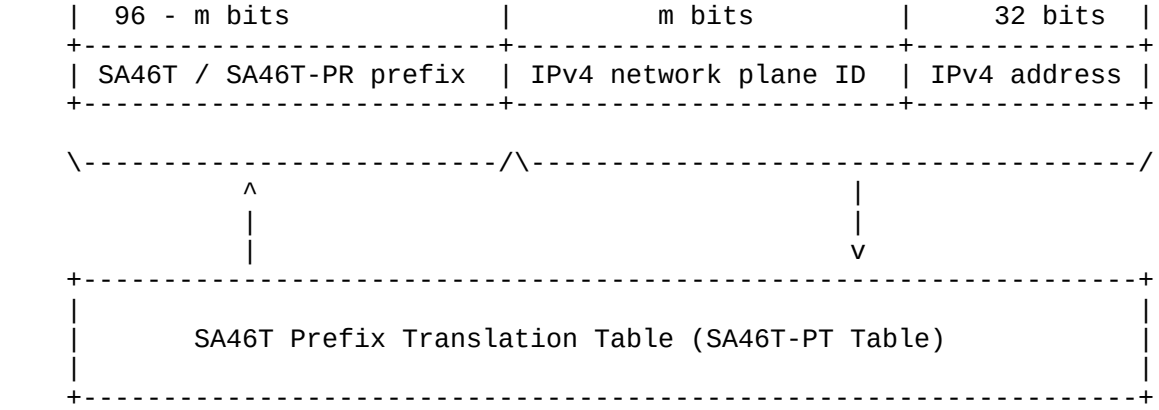


Figure 5

SA46T-AT table is similar with SA46T-PR table, however SA46T-AT table may contain SA46T prefix.

IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
:	:	:	
IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix

Figure 6

3.4. Destination address resolution

For address resolution for destination address, SA46T-PT use SA46T-PT table.

3.5. Source address resolution

For address resolution for source address, SA46T-PT use interface information, not SA46T-PT table. From SA46T domain to SA46T-PR domain, SA46T-PT use IPv6 address prefix of the interface which belong SA46T-PR domain. From

4. Sample Configuration

Figure Figure 7shows sample confiuration of SA46T-PT. In this example, there are four IPv4 stub network with the same IPv4 network plane, and two of four are in SA46T domain and other two of four are in SA46T-PR domain.

In this example, SA46T prefix is 2001:0db8:0:46::/64.

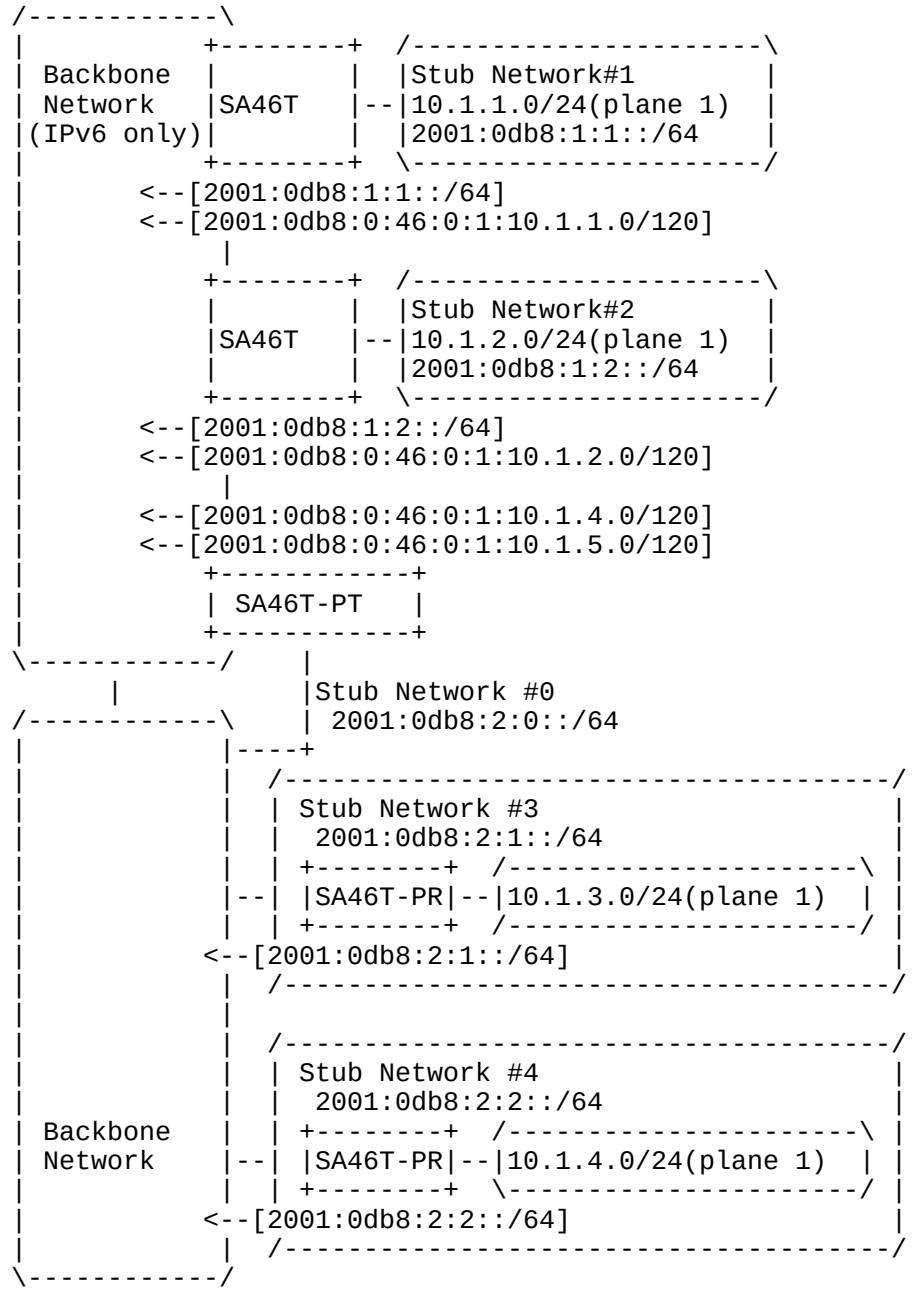

Backbone Network (IPv6 only)
SA46T
Stub Network#1
10.1.1.0/24(plane 1)
2001:0db8:1:1::/64
<--[2001:0db8:1:1::/64]
<--[2001:0db8:0:46:0:1:10.1.1.0/120]
Stub Network#2
10.1.2.0/24(plane 1)
2001:0db8:1:2::/64
<--[2001:0db8:1:2::/64]
<--[2001:0db8:0:46:0:1:10.1.2.0/120]
<--[2001:0db8:0:46:0:1:10.1.4.0/120]
<--[2001:0db8:0:46:0:1:10.1.5.0/120]
SA46T-PT
Stub Network #0
2001:0db8:2:0::/64
Stub Network #3
2001:0db8:2:1::/64
SA46T-PR
10.1.3.0/24(plane 1)
<--[2001:0db8:2:1::/64]
Stub Network #4
2001:0db8:2:2::/64
SA46T-PR
10.1.4.0/24(plane 1)
<--[2001:0db8:2:2::/64]

Figure 7

Figure Figure 8 shows SA46T-PT table for this example. This example is default free case.

IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
1	10.1.1.0	/120	2001:0db8:0:46
1	10.1.2.0	/120	2001:0db8:0:46
1	10.1.3.0	/120	2001:0db8:2:1
1	10.1.4.0	/120	2001:0db8:2:2

Figure 8

Fig Figure 9shows another SA46T-PT table for this example. This example use default for SA46T. If there are many stub network in SA46T domain, by using default as SA46T prefix, reduction of SA46T-PT table size can be possible.

IPv4 network plane ID	IPv4 address	netmask	SA46T-PR address prefix
1	10.1.3.0	/120	2001:0db8:2:1
1	10.1.4.0	/120	2001:0db8:2:2
1	0.0.0.0	/0	2001:0db8:0:46

Figure 9

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Security Considerations

Security Considerations does not discussed in this memo.

7. Acknowledgements

8. References

8.1. Normative References

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

8.2. Informative References

[I-D.draft-matsuhira-sa46t-pr-spec]
Matsuhira, N., "SA46T Prefix Resolution (SA46T-PR)", January 2014.

[I-D.draft-matsuhira-sa46t-spec]
Matsuhira, N., "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology: Specification", January 2014.

Author’s Address

Naoki Matsuhira
Fujitsu Limited
1-1, Kamikodanaka 4-chome, Nakahara-ku
Kawasaki, 211-8588
Japan

Phone: +81-44-754-3466
Fax:
Email: matsuhira@jp.fujitsu.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 24, 2016

N. Matsuhira
Fujitsu Limited

July 23, 2015

Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology: Specification
draft-matsuhira-sa46t-spec-11

Abstract

This document specifies Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology (SA46T) base specification. SA46T makes backbone network to IPv6 only. And also, SA46T can stack many IPv4 networks, i.e. the networks using same IPv4 (private) addresses, without interdependence.

Requirements Language

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 RFC 2119 [RFC2119].

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 January 24, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents

(http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Architecture of SA46T 3
3. Basic Network Configuration 5
4. Basic Function of SA46T 6
4.1. IPv4 over IPv6 Encapsulation / Decapsulation 6
4.2. SA46T address architecture 7
4.3. Route Advertisement 8
5. SA46T address format 9
5.1. IPv6 Global Unicast Address as SA46T address 9
5.2. Global SA46T address format 10
6. Stacking IPv4 Networks 10
7. Redundancy of SA46T 12
8. Configuration of SA46T and address allocation 12
9. Example of SA46T Operation 16
9.1. Basic SA46T Operation 16
9.2. SA46T Operation with plane ID 18
10. Characteristic 21
11. IANA Considerations 22
12. Security Considerations 22
13. Acknowledgements 22
14. References 23
14.1. Normative References 23
14.2. References 23
Appendix A. Test implementation of SA46T 24
Appendix B. SA46T experiments 24
B.1. WIDE camp at Sept 2010 24
B.2. NICT JGN2Plus Testbed at Feb 2011 24
B.3. Some corporate network 25
B.4. Interop 2011 Tokyo at Jun 2011 25
Author's Address 25

1. Introduction

This document provides Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology (SA46T) base specification.

The basic strategy for IPv6 deployment is dual stack. Viewing this strategy from operational side, operation cost of dual stack is higher than single stack operation. Viewing from future, IPv6 only operation is more reasonable rather than IPv4 only operation. Therefore IPv6 only operation is desired.

SA46T makes backbone network to IPv6 only. And also, SA46T can stack many IPv4 networks, i.e. the networks using same IPv4 (private) address, without interdependence.

2. Architecture of SA46T

IP address contain two information, one is locator information, and another is identifier information. This is basic architecture of internet protocol, and also the Internet, and no difference between IPv4 and IPv6.

Locater is a information related "Where", and indentifier is a information related "Who". That mean, IP address's semantics is "Where's Who" meaning. Host is identified whole IP address information, that is "Where's Who", however route to the host is identified just locator information in IP address, that is "Where". See Figure 1.

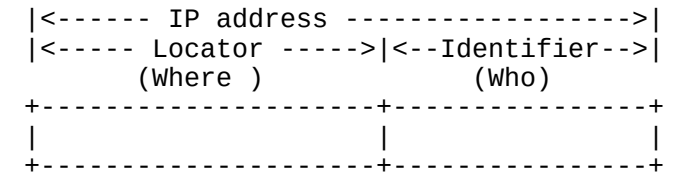


Figure 1

In IPv4 address space, some host has IPv4 address, which consist n bits length identifier and 32 - n bits locator. In Where's Who representation, 32 - n bits "Where" and n bits "Who".

Keeping such "Where's Who" relation, IPv4 address can be represent as IPv6 address by expanding "Where" information from 32 - n bits to 128 - n bits. Expanding " Where" information, IPv4 address can be mapped

to IPv6 address. Figure 2 shows such expanding.

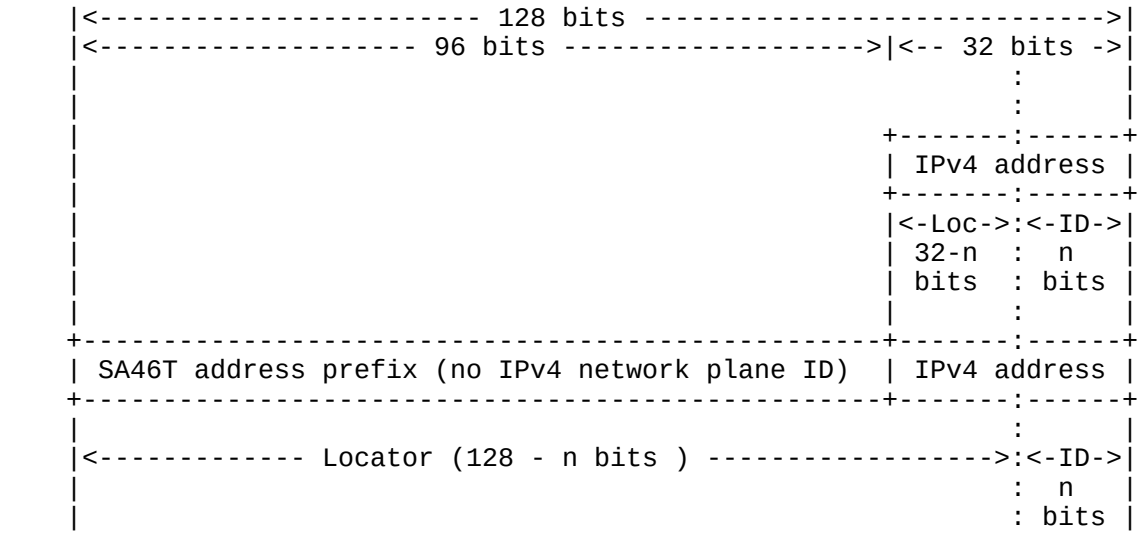


Figure 2

IPv4 address space contain private address, that is non globally unique IP address. If some identifier which distinguish private address can introduce in IPv6 address space, we can treate IPv4 private address as different address in IPv6 address space. This document define such identifier as "IPv4 network plane ID". "IPv6 network plane ID" can provide VPN (Virtual Private Network) like service.

That is SA46T address. In SA46T address, "Where" information's bit length is 128 -n bits, and "Who" information's bit length is n bits. Figure 3 shows summary of IPv4 address and SA46T address relation.

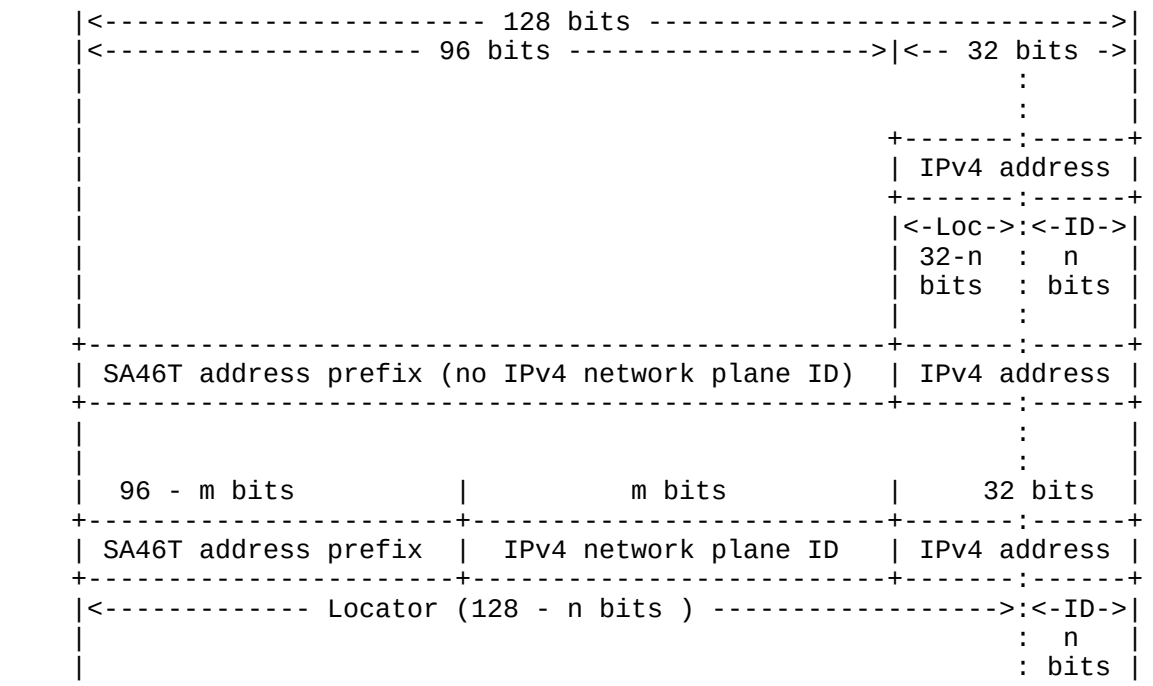


Figure 3

3. Basic Network Configuration

Figure 4 shows network configuration with SA46T. The network consists of three parts. Backbone network, stub network, and SA46T.

Backbone network is operated with IPv6 only. Stub network has three cases. IPv4 only, Dual Stack (both IPv4 and IPv6), and IPv6 only.

SA46T connects backbone network and stub network in case IPv4 still works in that stub network. If stub network is IPv6 only, SA46T is not needed.

Campus network, corporate network, and ISP network are the example for such network.

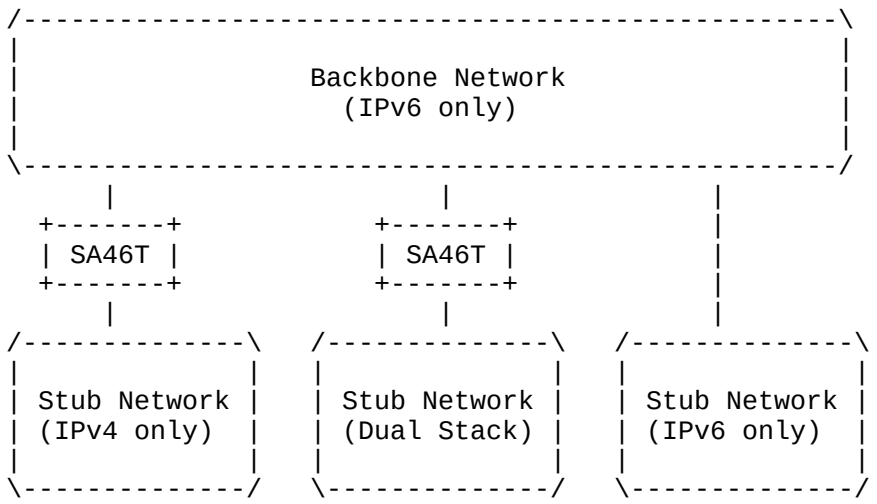


Figure 4

4. Basic Function of SA46T

SA46T has mainly two function. One is IPv4 over IPv6 Encapsulation / Decapsulation, and another is advertise route for stub network.

4.1. IPv4 over IPv6 Encapsulation / Decapsulation

SA46T encapsulates IPv4 packet to IPv6 from stub network to backbone network, and decapsulates IPv6 packet to IPv4 from backbone network to stub network. Figure 5 shows such movement.

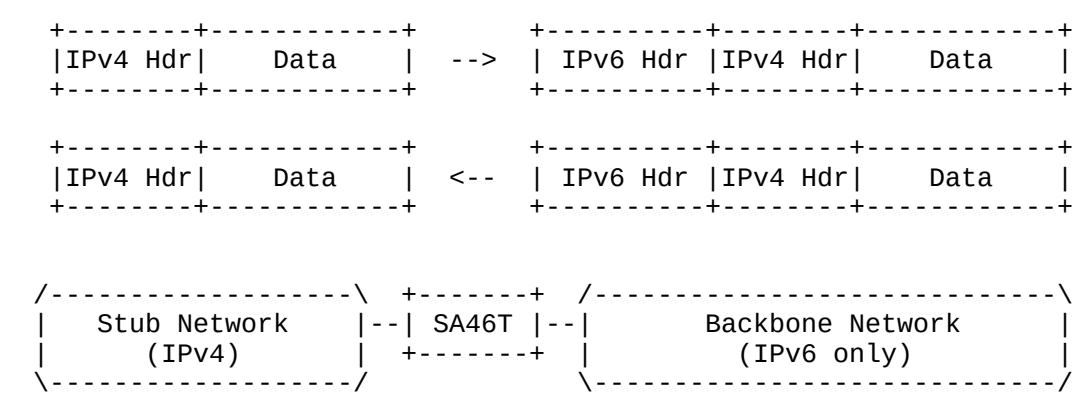


Figure 5

SA46T MUST support tunnel MTU discovery [RFC1853]. When encapsulated IPv6 Packet size exceed path MTU and inner IPv4 packet have the Don't Fragment bit is set, SA46T MUST return ICMP Destination unreachable message with Type3 Code4, fragmentation needed and DS set [RFC0792].

In case IPv6, SA46T just relays IPv6 packet.

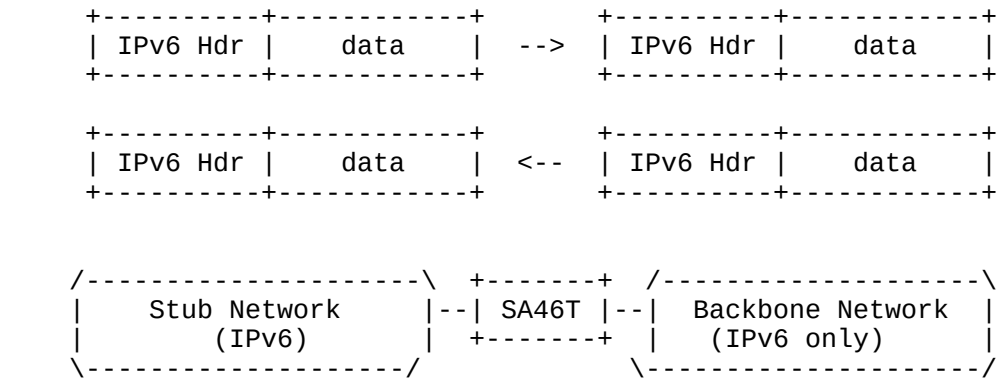


Figure 6

By IPv4 over IPv6 function, SA46T make backbone network to IPv6 only.

4.2. SA46T address architecture

SA46T address is a IPv6 address used in outer IPv6 header which encapsulate IPv4 packet by SA46T.

Figure 7 shows SA46T address architecture

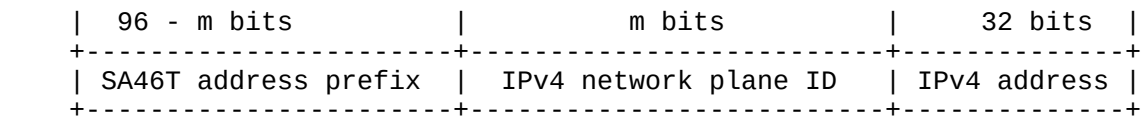


Figure 7

SA46T address consists of three parts as follows.

SA46T address prefix

SA46T address prefix indicates this packet is encapsulated by SA46T and MUST be encapsulated by SA46T. This value is preconfigured to all SA46T in the networks.

IPv4 network plane ID

IPv4 network plane ID is an identifier of IPv4 network stack over IPv6 backbone network. This value is preconfigured depend on the SA46T belong which IPv4 network plane. For more detail see Section 6.

IPv4 address

IPv4 address in inner IPv4 packet.

SA46T address is resolved copying IPv4 address in inner IPv4 packet, and preconfigured values, SA46T prefix and IPv4 network plane ID.

Table 1 shows SA46T IPv4 network plane ID length (m) and number of plane.

m	# of plane
16	65536
32	4294967296
64	18446744073709551616

Table 1

4.3. Route Advertisement

SA46T converts stub network's IPv4 route to SA46T IPv6 route and advertises to backbone network. And reverse direction, SA46T converts SA46T IPv6 route to IPv4 route, that advertises other IPv4 stub networks.

If IPv4 stub network's prefix length is n, the prefix length of SA46T IPv6 route which converts from that IPv4 prefix is 128 - 32 + n. Table 2 shows detail value.

IPv4 prefix length	SA46T IPv6 prefix length
/8	/104
/16	/112
/24	/120

Table 2

The IPv4 route for stub network is map to SA46T IPv6 route one to one, so number of route of IPv4 is same as number of route of SA46T IPv6 route. Total number of route is same as when backbone network operate dual stack, without SA46T.

In stub network, usual dynamic routing protocol for IPv4 and IPv6 can be used such as RIPv2 [RFC2453], RIPng [RFC2080], OSPFv2 [RFC2328], OSPFv3 [RFC2740] and IS-IS [RFC1195][RFC5308]. Similarly, in backbone network, usual dynamic routing protocol for IPv6 can be used such as RIPng [RFC2080], OSPFv3 [RFC2740] and IS-IS [RFC5308] .

If want using default route, default SA46T advertise the route [SA46T address prefix/(96 - m)] as default route. If want using different default route by IPv4 network plane ID, default SA46T in IPv4 network plane #1 advertise the route [SA46T address prefix + IPv4 network plane ID #1 / 96] as default route. Figure 15 in Section 9 show the example using default route.

5. SA46T address format

SA46T can be used closely in the backbone network, so SA46T address does not be advertised outside of the backbone network, and IPv6 packet which contains SA46T address does not be forwarded outside of the backbone network.

So, SA46T address format and SA46T address prefix can be decided each backbone network. But for your information, one example is shown as follows. That is based on IPv6 Global Unicast Address.

Of course, SA46T can be used in the Internet, or between the ASs. This case is discussed shortly in Section 5.2.

5.1. IPv6 Global Unicast Address as SA46T address

This example is based on IPv6 Global Unicast Address Format [RFC3587].

Figure 8 shows IPv6 Global Unicast Address Format.

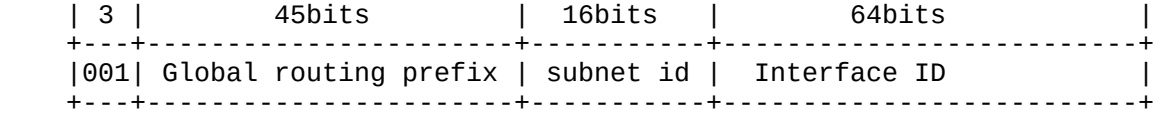
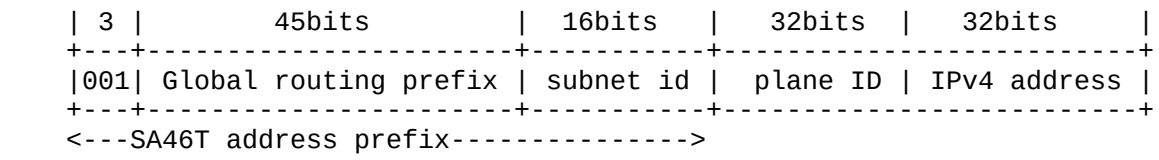


Figure 8

Figure 9 shows SA46T address format using part of IPv6 Global Unicast Address.



Where:

Global routing prefix

global routing prefix

subnet id

indication for SA46T prefix. Example is 0x5A46.

plane id

IPv4 network plane ID. The value 0 should be for the global IPv4 Internet.

IPv4 address

IPv4 address of inner IPv4 packet

5.2. Global SA46T address format

SA46T can be used in The Internet, or between AS. This is achieved by recognizing SA46T address format as common address. Such address should be Global SA46T address.

Global SA46T address format and prefix requires IANA assignment of IPv6 address prefix. Global SA46T address is proposed in [I-D.draft-matsuhira-sa46t-gaddr].

6. Stacking IPv4 Networks

SA46T can provide VPN like service to stub networks by using different IPv4 network plane ID value. Table 3 shows example of IPv4 network plane ID and its usage.

If backbone network operator provide IPv4 privates network service to Organization A, backbone network operator sets IPv4 network plane ID value =1 to the SA46T which connects stub network of organization A. If there are five stub network of organization A, backbone network operator sets same IPv4 network plane ID = 1, to five SA46Ts which connect stub network of organization A. If there are one hundred stub network of organization B, backbone network operator sets same IPv4 network plane ID = 2, to one hundred SA46Ts which connect stub network of organization B. If a new stub network in organization B join, backbone network operator configures same IPv4 network plane ID = 2, to the new stub network only, which connect stub network of organization B, and no configuration is needed to one hundred SA46Ts which are already connected.

Such configuration, that means same stub network group to same IPv4 network plane ID value, is simple and easy to understand, so, it is expected that possibility of misconfiguration is very low. And also, number of configuration is minimum, that mean, number of configuration is same as number of stub networks, and add new stub network, configure to new one only.

Describe above, SA46T can provide VPN like service, for example, Intranet or extranet. And, after IPv4 global address running out, some service provider may want to reuse IPv4 private address. SA46T can provide such IPv4 private address networks over single IPv6 backbone network. By SA46T, some service providers may reuse IPv4 private address.

IPv4 network plane ID value	usage
0	IPv4 Internet (Global)
1	IPv4 Private network for Organization A (Intranet)
2	IPv4 Private network for Organization B (Intranet)
3	IPv4 Private network for Group A (Extranet)
4	IPv4 Private network for Group B (Extranet)
5	Net10 reuse network for consumer group A (Private address access)
6	Net10 reuse network for consumer group B (Private address access)
7	Net10 reuse network for consumer group C (Private address access)
....	....

Table 3

7. Redundancy of SA46T

SA46T brings no limit for redundancy. Figure 10 shows such example in case two connection between backbone network and stub network. Number of link between backbone network and stub network is not limited, and different type of link can be used, for example, for wire and wireless.

Configuration of SA46Ts, which connect same stub network, is same. That mean same SA46T prefix and same IPv4 network plane ID value.

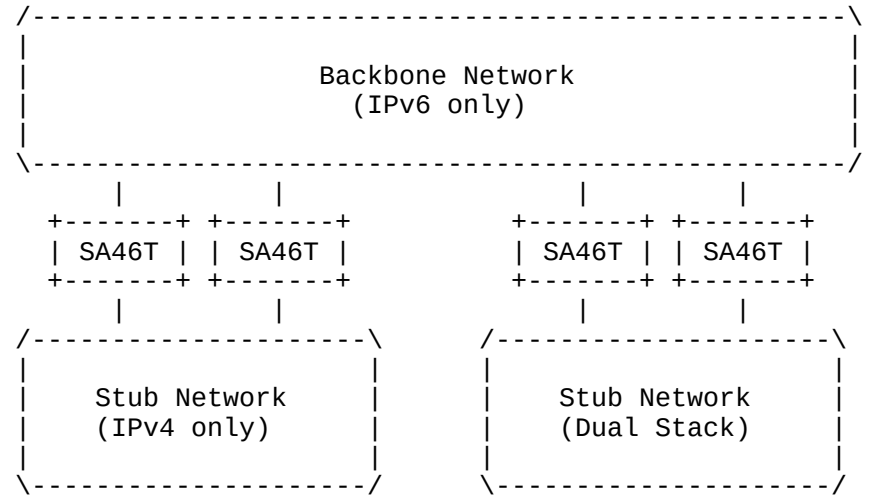


Figure 10

8. Configuration of SA46T and address allocation

Configuration of SA46T require just three information, SA46T address prefix, IPv4 Network plane ID, and prefix length of SA46T route. These information could explain just only one line, "<SA46T address prefix><IPv4 network plane ID>/ prefix length of SA46T route".

When there are N numbers SA46Ts in a certain backbone network, configure one line per SA46T to the N numbers SA46Ts are needed. Total line is just N. If adding new SA46T to the backbone network, configure one line to the new SA46T only is needed, and addition or change does not needed to existing N numbers SA46Ts. Now new 1 line

and total numbers of line is N+1.

Static configured tunnel require N(N-1) configurations. So, SA46T needs less configuration than static configured tunnel, especially when value of N is large number.

SA46T require few configuration, so when numbers of SA46T is small, manual configuration may be enough. However, when large number of SA46T needed in big network, configuration via server may useful. For automatic configuration of SA46T, IPv4 address allocation in stub network should consider, both static address allocation and automatic address allocation. In the latter case, using DHCP should be reasonable.

Figure 11 shows example of configuration database for SA46T. As identifier of SA46T, MAC address is used, however, other information may be used.

When stub network connected SA46T is configured with dynamic address, allocate IPv4 address in allocatable IPv4 address block to the stub network side interface of SA46T at startup phase. That is default router address in the stub network. When SA46T receive DHCP request from a host in stub network, DHCP server allocate IP address from allocatable IPv4 address block, and notify IP address of DNS server and IP address of default router.

When stub network connected SA46T is configuraed with static address, a value of allocatable IPv4 address block should be 0.0.0.0/0 and a value of DNS Server should be 0.0.0.0..

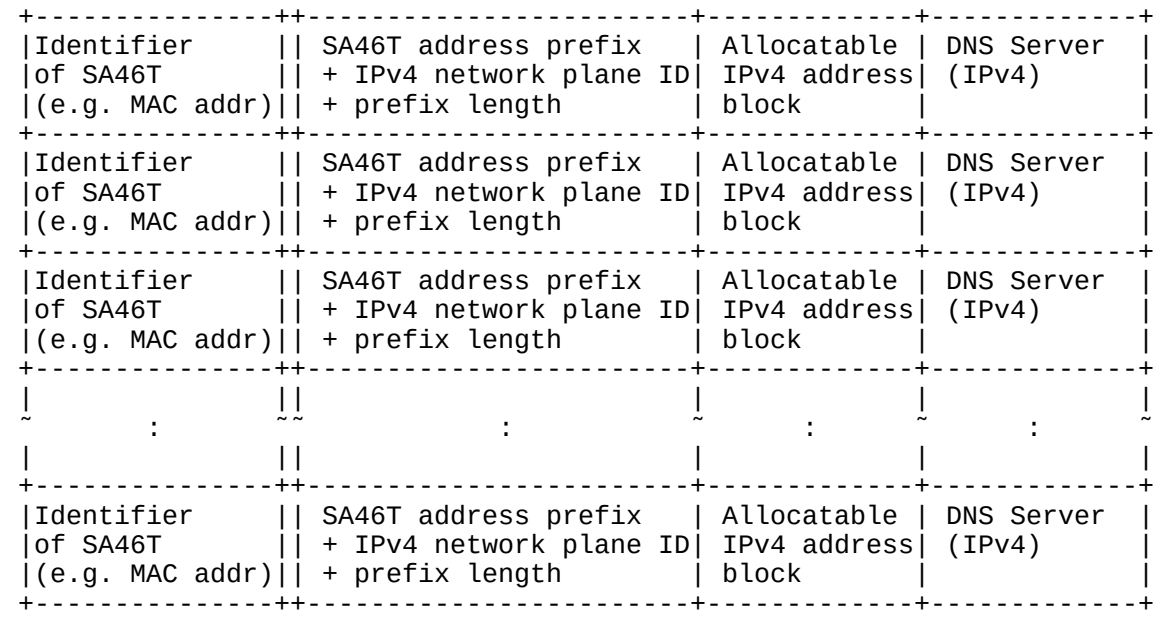


Figure 11

Figure 12 shows timeline diagram of message exchange between SA46T and host in stub network and SA46T configuration server when stub network is configured with dynamic address. Protocol between SA46T and SA46T configuration server including SA46T server discovery may be defined in future.

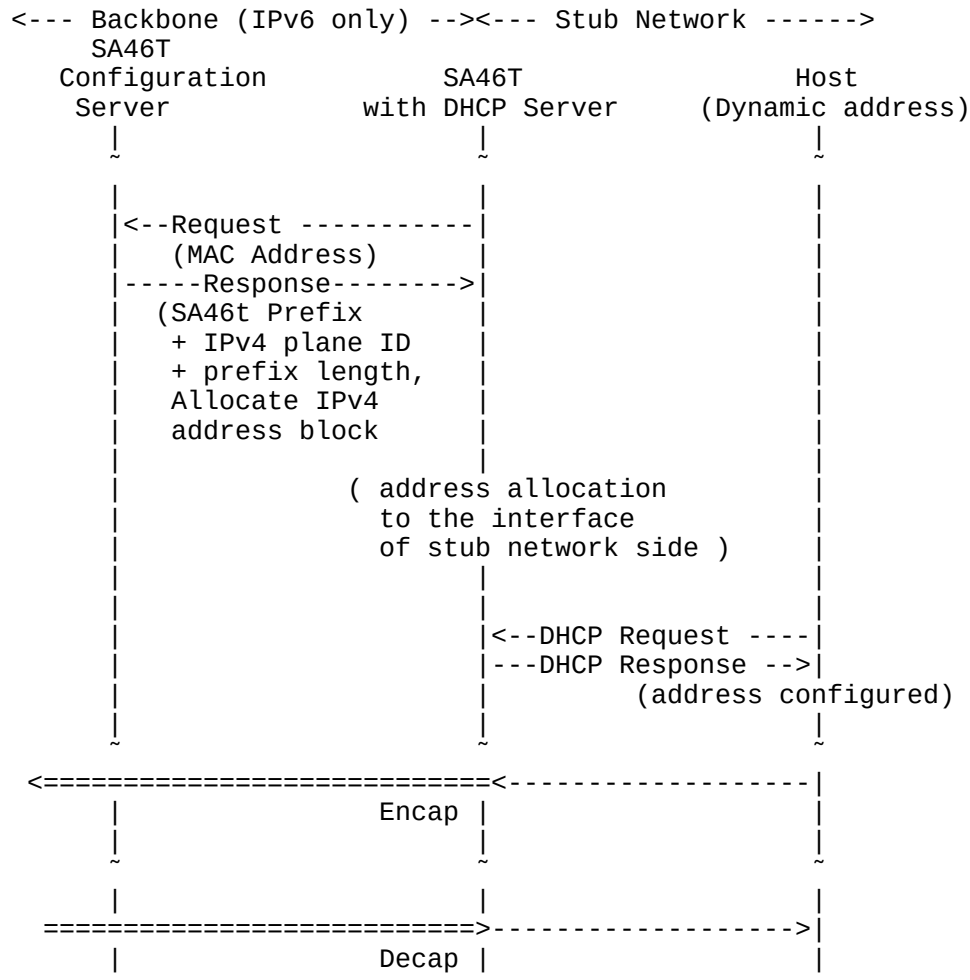


Figure 12

Figure 13 shows timeline diagram of message exchange between SA46T and host in stub network and SA46T configuration server when stub network is configured with static address. Such static address configuration may be used mainly at server zone, so such stub network may be well managed, so SA46T may also configured manually.

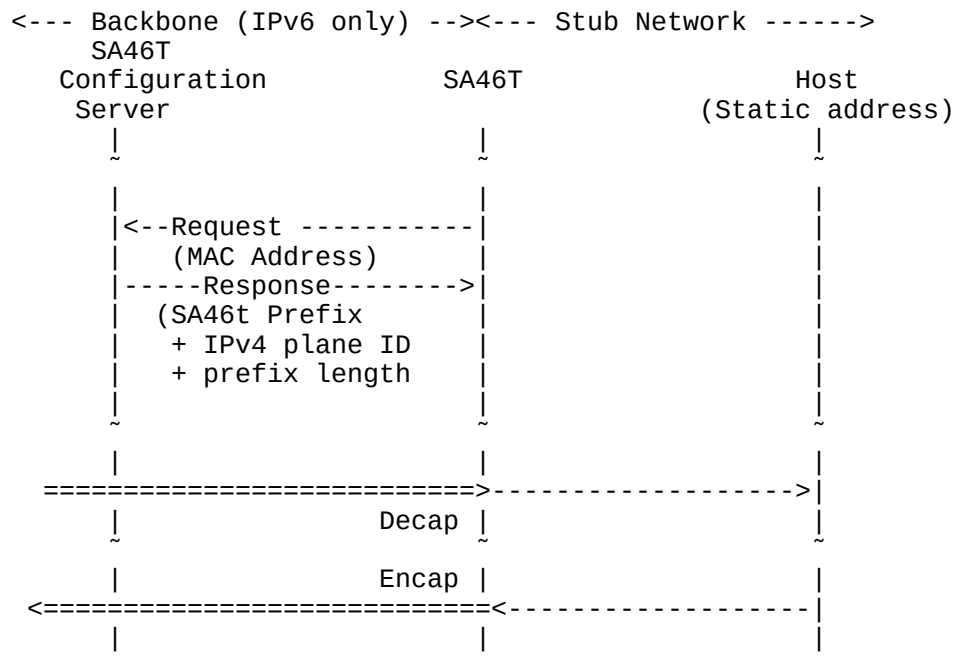


Figure 13

9. Example of SA46T Operation

9.1. Basic SA46T Operation

Figure 14 shows SA46T operation which does not use IPv4 network plane ID. In this example, two stub network is connected to backbone network via SA46T. One stub network is 10.1.1.0/24 sub network, and the other is 10.1.2.0/24 sub network.

When SA46T receives IPv4 route advertisement, then SA46T convert this IPv4 route to IPv6 route by address resolution to SA46T address, and advertise this IPv6 route to backbone network. When SA46T receives IPv6 route advertisements, then SA46T converts this IPv6 route to IPv4 route if this IPv6 route is match SA46T address (same prefix with SA46T), and advertise this IPv4 route to stub network.

In this example. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix>:10.1.1.0/120,and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix>:10.1.2.0/120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix>:10.1.1.0/120 is converted to IPv4 route,

10.1.1.0/24, and IPv6 route, <SA46Tprefix>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

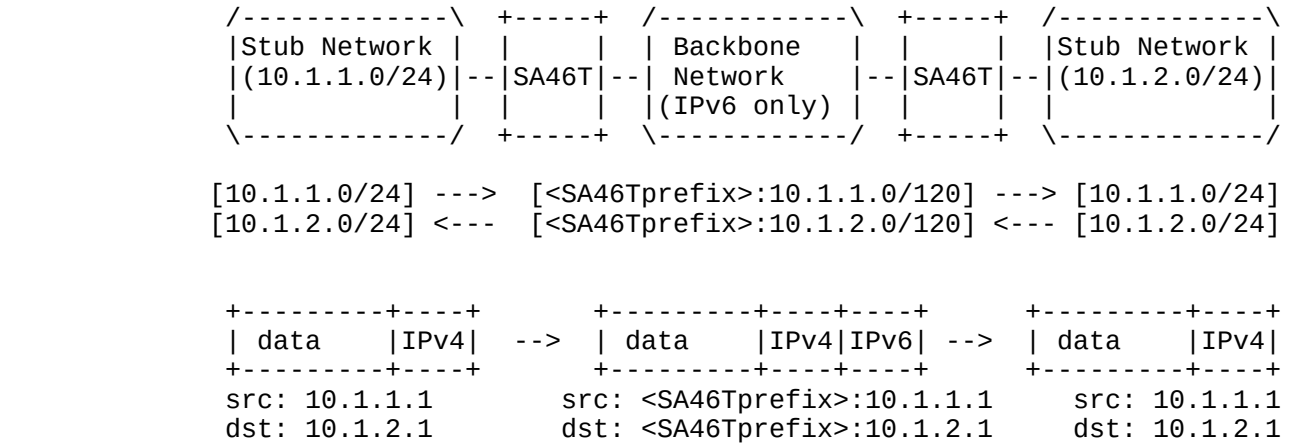


Figure 14

Figure 15 shows the example using default route. Default route is useful in case most packets are routed same path. Typically, access network is one of the example. Although using default route, communication between stub networks can be done. Communication between host 10.1.1.1 and host 10.1.2.1 can be done inside in access network, and does not pass over default SA46T.

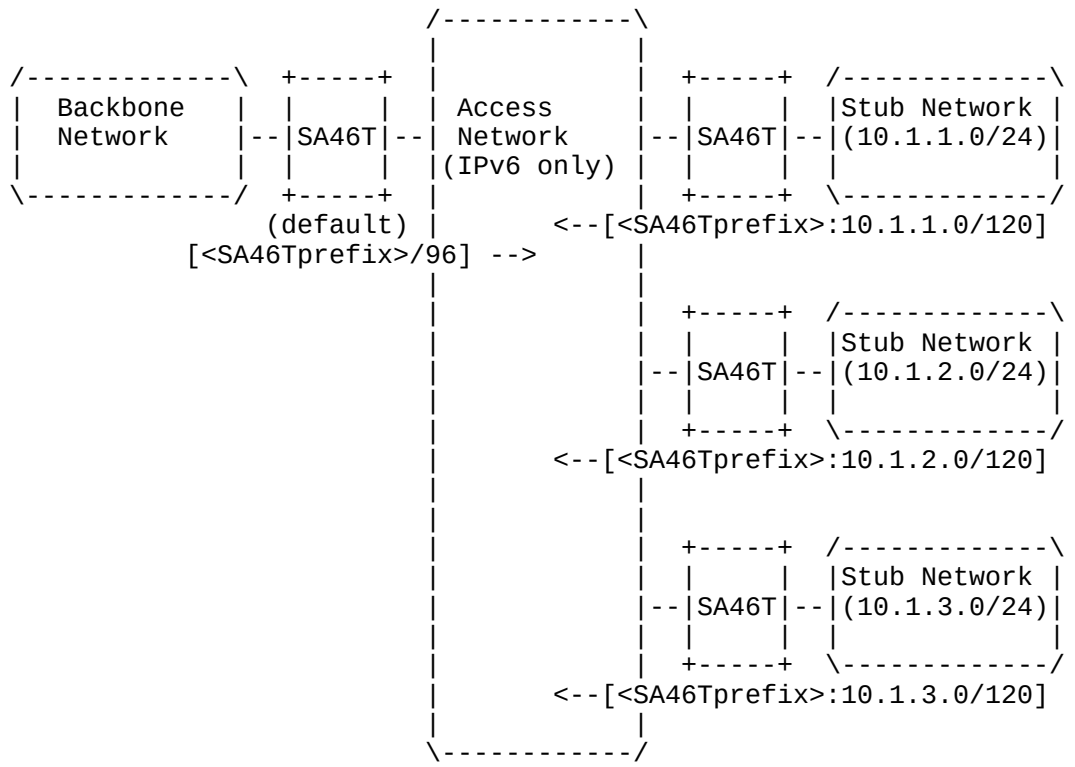


Figure 15

9.2. SA46T Operation with plane ID

Figure 16 shows SA46T operation which uses IPv4 network plane ID. In this example, there are two planes, and two stub network in each plane is connected to backbone network via SA46T. In each plane, one stub network is 10.1.1.0/24 sub network, and the other is 10.1.2.0/24 sub network, that means same IPv4 address is used in different plane.

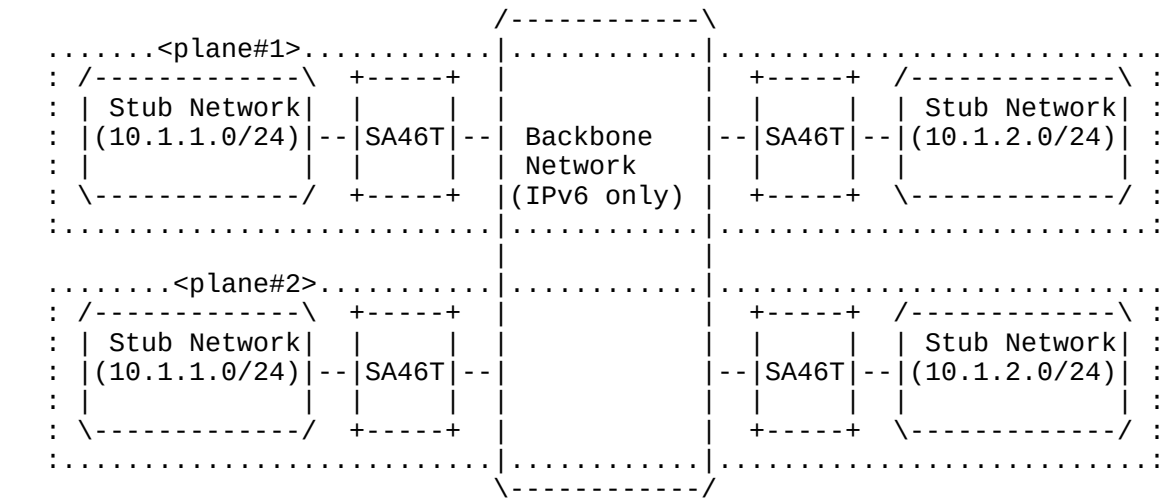
When SA46T receives IPv4 route advertisements, then SA46T converts this IPv4 route to IPv6 route by address resolution to SA46T address, and advertise this IPv6 route to backbone network. When SA46T receives IPv6 route advertisements, then SA46T converts this IPv6 route to IPv4 route if this IPv6 route is match SA46T address (same prefix with SA46T), and advertises this IPv4 route to stub network.

In this example in plane #1. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix><#1>:10.1.1.0/120,and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix><#1>:10.1.2.0/

120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix><#1>:10.1.1.0/120 is converted to IPv4 route, 10.1.1.0/24, and IPv6 route, <SA46Tprefix><#1>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

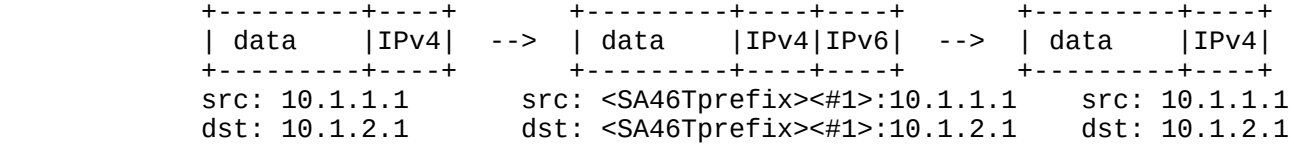
And also, In this example in plane #2. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix><#2>:10.1.1.0/120,and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix><#2>:10.1.2.0/120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix><#2>:10.1.1.0/120 is converted to IPv4 route, 10.1.1.0/24, and IPv6 route, <SA46Tprefix><#2>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

In IPv6 space, address <SA46Tprefix><#1>:10.1.1.1 and address <SA46Tprefix><#2>:10.1.1.1 are different address, route <SA46Tprefix><#1>:10.1.1.0/120 and route <SA46Tprefix><#2>:10.1.1.0/120 are different route, although in IPv4 space, address 10.1.1.1 in plane #1 and 10.1.1.1 in plane#2 are same address, route 10.1.1.0/24 in plane#1 and route 10.1.1.0/24 in plane#2 are same route.



<<plane #1>>

[10.1.1.0/24] --->[<SA46Tprefix><#1>:10.1.1.0/120] ---> [10.1.1.0/24]
[10.1.2.0/24] <---[<SA46Tprefix><#1>:10.1.2.0/120] <--- [10.1.2.0/24]



<<plane#2>>

[10.1.1.0/24] --->[<SA46Tprefix><#2>:10.1.1.0/120] ---> [10.1.1.0/24]
[10.1.2.0/24] <---[<SA46Tprefix><#2>:10.1.2.0/120] <--- [10.1.2.0/24]

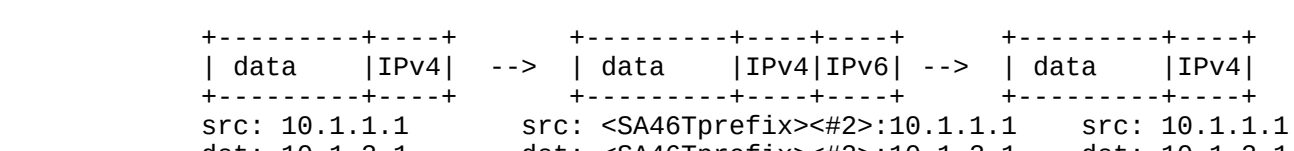


Figure 16

Figure 17shows the example using default route with IPv4 network plane. In this case, default SA46T may configure different by each IPv4 network plane.

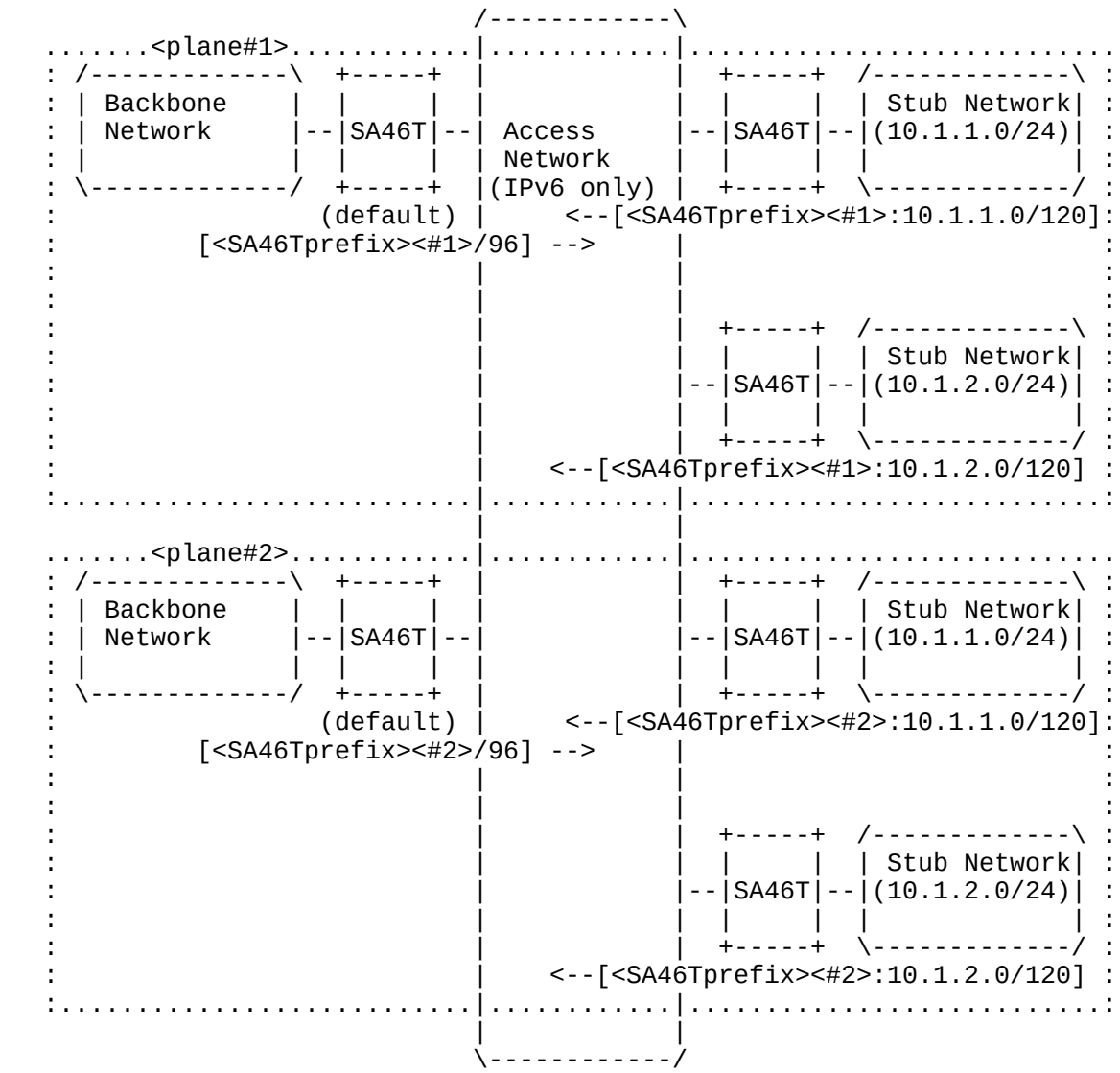


Figure 17

10. Characteristic

SA46T has following useful characteristics.

- Reduce backbone network operation cost with IPv6 single stack (at least less than Dual Stack)
- Can allocate IPv4 address to stub networks, which used in backbone network before installing SA46T
- Less configuration
- No need for special protocol
- No dependent Layer 2 network
- Can Stack IPv4 Private networks
- Easy stop IPv4 operation in stub network for future (just remove SA46T)
- Provide redundancy

11. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

12. Security Considerations

SA46T use automatic Encapsulation / Decapsulation technologies. Security consideration related tunneling technologies are discussed in RFC2893[RFC2893], RFC2267[RFC2267], etc.

13. Acknowledgements

This document is based on Naoki Matsuhira’s original ideas and an individual effort of the author.

Review and encouragement have been provided by many peoples. Particulary Akira Kato at WIDE Project / Keio University and Masanobu Katoh at Fujitsu in initial stage. And many discussions and assists are provided from Toshiya Asaba, Osamu Nakamura, Yoshiki Ishida, Ichiro Mizukoshi, Noriyuki Shigechika, Miya Kohno, Yoshinobu Matsuzaki, Akira Nakagawa. And comments and discussions are provided in IETF meeting from Fred Baker, Brian Carpenter, Randy Bush, Dave Thaler and Alain Duland. If there is a comment not refrected, it is

surely because of my English language capability, and the author still want reflect it include missing.

The author would like to thank all above people, and others discussed with in WIDE project meeting and inside Fujitsu.

Originally, SA46T is an abbreviation for "Stateless Automatic IPv4 over IPv6 Tunneling". Now, SA46T is an abbreviation for "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology". This change was made in response to the indication from the softwire WG chair at 4th softwire interim meeting in September 2011.

14. References

14.1. Normative References

[I-D.draft-matsuhira-sa46t-gaddr] Matsuhira, N., "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology: Global SA46T Address Format", January 2014.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[RFC1853] Simpson, W., "IP in IP Tunneling", RFC 1853, October 1995.

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

[RFC3587] Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global Unicast Address Format", RFC 3587, August 2003.

14.2. References

[RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.

[RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, January 1997.

[RFC2267] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", RFC 2267, January 1998.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[RFC2453] Malkin, G., "RIP Version 2", STD 56, RFC 2453,

November 1998.

[RFC2740] Coltun, R., Ferguson, D., and J. Moy, "OSPF for IPv6", RFC 2740, December 1999.

[RFC2893] Gilligan, R. and E. Nordmark, "Transition Mechanisms for IPv6 Hosts and Routers", RFC 2893, August 2000.

[RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, October 2008.

Appendix A. Test implementation of SA46T

Test implementation of SA46T is developed for evaluation the SA46T technology. This implementation is developed as module in kernel space of CentOS. The amount of development is about 300 step with C language.

Appendix B. SA46T experiments

B.1. WIDE camp at Sept 2010

SA46T implementation is tested at WIDE camp in 4.5 days at Semtemper 2010. Attendees of WIDE camp served SA46T service via WIreless LAN. SA46T provide both IPv4 and IPv6. IPv4 packets are encapsulated and decapsulated in camp net, that mean this test is in LAN environments. This time single IPv4 plane was used.

About 200 peoples joins this experiments and 275 clients are used, inculude Windows, MacOS, Linux, FreeBSD, iPhone and iPod Touch, etc. IPv4 address is allocated via DHCP. There are no change in clients, servers, and network equipment, just add SA46T. Total, four SA46T boxes were used in this experiments.

SA46T work fine and very stable.

B.2. NICT JGN2Plus Testbed at Feb 2011

SA46T implementation is tested at NICT JGN2Plus testbed at February 2011. This test is held at WAN environments. SA46T is setted up at Sapporo, Osaka, Okayama and Okinawa in Japan and Thai, and carry HDTV Live Stream and 3D HDTV Live stream. Experimental period is about an one month. Total, five SA46T boxes were used in this experiments.

In JGN2Plus, OSPFv3 was used, and BGP4+ is used for peering with Thai.

This time, single IPv4 plane was used too.

SA46T work fine and very stable, too.

B.3. Some corporate network

SA46Ts are installed some corporate network. This installation is done with secrets basically, that mean, nobody know SA46T was installed, and if there are some trouble, someone craim or report the problem.

After few month trial, there was no problem.

B.4. Interop 2011 Tokyo at Jun 2011

SA46T is demonstrated at Interop 2011 Tokyo at June 2011.

At this time, three planes were used. Plane #0 is used for Internet access, using IPv4 Global address. Visitor can have a experiments with SA46T from the cables wich connected to SA46T in access corner. Plane #1 is used for closed network, such like between Data Center network and enterprise network. In this plane, private addresses are used. Plane #2 is used for video streaming. In this plane, same private addresses which used in Plane#1 are used by intention. And this plane in Interop ShowNet and NICT and Thai were connected.

Total, nine SA46T boxes are used in this demonstration.

About 128,000 peoples visit in this event, and see many demonstration include SA46T.

SA46T work fine and very stable, too.

Author's Address

Naoki Matsuhira
Fujitsu Limited
1-1, Kamikodanaka 4-chome, Nakahara-ku
Kawasaki, 211-8588
Japan

Phone: +81-44-754-3466
Fax:
Email: matsuhira@jp.fujitsu.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 05, 2013

C. Xie
China Telecom
S. Perreault
Viagenie
C. Zhou
Huawei Technologies

June 03, 2013

Provisioning Lightweight 4over6 (lw4o6) with the Port Control Protocol (PCP)

draft-perreault-softwire-lw4over6-pcp-00

Abstract

This memo defines the procedures that a Lightweight B4 uses for provisioning its parameters with the Port Control Protocol.

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 December 05, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 2
3. Lightweight B4 Provisioning with PCP 3
3.1. Setting Up the Tunnel 3
3.2. Configuration of the NAT44 4
3.3. PCP Proxy 4
3.4. Failover Mechanism 4
4. Security Considerations 5
5. IANA Considerations 5
6. Acknowledgements 5
7. References 5
7.1. Normative References 5
7.2. Informative References 6
Authors' Addresses 6

1. Introduction

Lightweight 4over6 (lw4o6) [I-D.ietf-softwire-lw4over6] defines a model for providing IPv4 access over an IPv6 network in which the Network Address Translation (NAT) function is performed by the Customer-Premises Equipment (CPE) instead of being centralized on a Carrier-Grade NAT (CGN).

Separately, the Port Control Protocol [RFC6887] is used to manipulate port mappings in a NAT, firewall, port range router, or similar equipment. It is extended in [I-D.ietf-pcp-port-set] with the ability to manipulate sets of ports instead of individual ports.

This document describes how PCP is used to provision a Lightweight B4 (lwB4) with its port set and how to establish a tunnel to the Lightweight AFTR (lwAFTR).

2. Terminology

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].

Terminology defined in [I-D.ietf-softwire-lw4over6] is used extensively in this document.

3. Lightweight B4 Provisioning with PCP

The elements that are needed for lwB4 provisioning are listed in Section 5.1 of [I-D.ietf-softwire-lw4over6].

> Note (to be removed before publication): These elements can be provisioned with plain mode or encapsulation mode.
>
> In the plain mode, PCP port-set request is sent using native IPv6 packet, while in the encapsulation mode, PCP Port-set request is sent using ip-in-ip tunnel. In this draft, encapsulation mode is recommended to guarantee that the same lwAFTR/PCP server would be selected for PCP requests and subsequent ip-in-ip traffic.

3.1. Setting Up the Tunnel

The lwB4 initiates the provisioning procedure by requesting the OPTION_AFTR_NAME DHCPv6 option as indicated in [RFC6334]. This option provides the IPv6 address for the lwAFTR.

Once this address is known, the lwB4 sets up an IPv4-in-IPv6 tunnel with the following characteristics:

IPv6 destination: value of OPTION_AFTR_NAME, after resolution of the name

IPv6 source: derived from the IPv6 destination by applying Default Address Selection [RFC3484]

IPv4 source: 192.0.0.2

IPv4 destination: 192.0.0.1

The IPv4 addresses correspond to the well-known B4 and AFTR addresses defined in Section 5.7 of [RFC6333].

3.2. Configuration of the NAT44

Once the tunnel is up, the lwB4 sends a PCP MAP request with a PORT_SET option. The request is sent inside the tunnel to 192.0.0.1. The source is accordingly set to 192.0.0.2.

The MAP request's Internal Port is set to 1 and the PORT_SET option's Port Set Size field is set to 65535, indicating that the lwB4 is prepared to accept a maximal size port set. Practically, the server will reply with a port set size corresponding to its configuration.

Note: Since there is no NAT in the lwAFTR, the internal port is always equal to the external port. The PCP server cannot change the internal port that the client sends. How can we overcome this? Add an offset parameter in the PORT_SET option?

The PORT_SET option's P bit is set to 0.

When a success response is received from the PCP server, the lwB4 extracts the external IPv4 address and port set from the response and uses them to configure its NAT44 function as described in [I-D.ietf-softwire-lw4over6]. The lwB4 is now provisioned.

The lwB4 needs to periodically refresh the port set it obtained with PCP as described in [RFC6887] section 15 for as long as the lw4over6 tunnel is to be operational.

3.3. PCP Proxy

The lwB4 SHOULD implement a back-to-back PCP server-client. The PCP port-set client in lwB4 would get a public address and port-set from the PCP port-set server, and then the PCP server in the lwB4 will setup the mapping for the host behind the lwB4 and response with PCP client.

The lwB4 MAY also implement a PCP proxy in case the host initiates a port-set request directly. It would forward the port-set request to PCP server to get a new port-set mapping or refresh an existing mapping.

3.4. Failover Mechanism

This document considers two failover mechanisms: ICMP and PCP ANNOUNCE. In the ICMP case, when the lwB4 receives an ICMP error message from the lwAFTR, the lwB4 MAY re-initiate the dynamic port-restricted provisioning process. The detailed ICMP processing is introduced in [I-D.ietf-softwire-lw4over6].

In the PCP case, when the lwAFTR receives traffic it doesn't have before, lwAFTR MAY send back a PCP unicast ANNOUNCE message. The lwB4 then will re-initiate the PCP Port-set request after receiving the ANNOUNCE message. In the case when there are large amount of lwB4s, an optimization of this mechanism MAY be needed to achieve fast failure recovery. Since it is layer 2 network between lwB4 and BNG, A BNG device MAY act a PCP proxy to receive unicast ANNOUNCE message from lwAFTR. It will then replace the unicast address of itself with the lwb4's multicast address and sends multicast ANNOUNCE message to the lwB4s.

4. Security Considerations

TO BE COMPLETED

5. IANA Considerations

This document has no IANA actions.

6. Acknowledgements

Special thanks to Qiong Sun for her many contributions to this document.

The authors would like to thank the following individuals who have participated in the drafting, review, and discussion of this memo: Jean-Philippe Dionne, Marc Blanchet, and Tina Tsou.

7. References

7.1. Normative References

[I-D.ietf-pcp-port-set]
Sun, 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-00 (work in progress), March 2013.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 2013.

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

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

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

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

7.2. Informative References

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

Authors' Addresses

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

Email: xiechf@ctbri.com.cn

Simon Perreault
Viagenie
246 Aberdeen
Quebec, QC G1R 2E1
Canada

Phone: +1 418 656 9254
Email: simon.perreault@viagenie.ca
URI: http://viagenie.ca

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: cathy.zhou@huawei.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: March 15, 2014

B. Sarikaya
T. Tsou
Huawei Technologies (USA)
H. Ji
China Telecom
C. Zhou
Huawei Technologies
September 11, 2013

IPv6 Multicast in a 6rd Deployment
draft-sarikaya-softwire-6rdmulticast-06

Abstract

This memo specifies 6rd's multicast component so that IPv6 hosts can receive multicast data from IPv6 servers. In the 6rd encapsulation solution, multicast communication is completely integrated into the 6rd tunnel. In the 6rd translation solution, the protocol is based on proxying MLD at the 6rd Customer Edge router interworking the MLD messages to IGMP messages and sending them upstream through a network which supports IPv4 multicast. The 6rd Border Relay is a multicast router and interworks the IGMP to MLD for onward propasgation toward the IPv6 multicast source. IPv6 Multicast data received at 6rd Border Relay is translated into IPv4 multicast data and and delivered through the IPv4 multicast tree downstream to the 6rd Customer Edge. The latter translates it back to IPv6 multicast data then delivers it to the hosts.

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 March 15, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Requirements 3
4. Architecture 4
4.1. 6rd Tunneling Architecture 4
4.2. Translation Architecture 5
5. 6rd Tunneling Multicast Operation 6
5.1. Tunnel Interface Considerations 7
5.2. Avalanche Problem 8
6. 6rd Translation Multicast Operation 8
6.1. Solution Based on Layer 2 Multicast Support 10
6.2. Analysis 11
7. Security Considerations 11
8. IANA Considerations 11
9. Acknowledgements 12
10. References 12
10.1. Normative References 12
10.2. Informative References 13
Authors' Addresses 13

1. Introduction

With IPv4 address depletion on the horizon, many techniques are being standardized for IPv6 migration including 6rd [RFC5969]. 6rd enables IPv6 hosts to communicate with external hosts using an IPv4-only legacy ISP network. The 6rd Customer Edge (CE) device's LAN side is dual stack and the WAN side is IPv4 only. The CE tunnels IPv6 packets received from the LAN side to 6rd Border Relays (BR) after encapsulating them as IPv4 packets. The BRs have anycast IPv4 addresses and receive encapsulated packets from CEs over a virtual interface. 6rd operation is stateless. Packets are received/ sent independently of each other and no state needs to be maintained.

It should be noted that there is no depletion problem for IPv4 address space allocated for any source multicast and source specific multicast [RFC3171]. This document is not motivated by the depletion of IPv4 multicast addresses.

6rd as defined in [RFC5969] and [RFC5569] is unicast only. It does not support multicast. In this document we specify how multicast from home IPv6 users can be supported in 6rd. This is what is meant by 6rd multicast protocol.

In the 6rd encapsulation approach, 6rd multicast is integrated into the 6rd unicast solution. 6rd customer premise equipment (CPE) is extended to support an MLD proxy [RFC4605]. This proxy receives MLD Membership Report messages [RFC4601] requesting to join a multicast group from its subtended hosts. It tunnels aggregated join requests upstream to the 6rd Border Router (BR) using IPv6 in IPv4 encapsulation. The 6rd Border Router is extended to support an MLD querier, which sends join requests upstream towards the multicast source(s, becomes part of the multicast tree, and thus receives IPv6 multicast data. The 6rd Border Router encapsulates the IPv6 multicast data using 6rd's IPv6 in IPv4 encapsulation and sends it to each member CPE that has joined the stream concerned. The CPE decapsulates the packet and the MLD proxy sends the IPv6 multicast data downstream to the member hosts.

In the translation approach, native IPv4 multicast support in the network between Customer Edge routers and Border Router can be exploited. The translation approach requires MLD to IGMP interworking at the Customer Edge and IGMP to MLD interworking at the border router. The border router needs to translate IPv6 multicast data into IPv4 multicast data and the Customer Edge router needs to translate IPv4 multicast data back into IPv6 multicast data.

6rd's CE to CE forwarding feature is not used in either approach.

2. Terminology

This document uses the terminology defined in [RFC5969], [RFC5569], [RFC3810], [RFC3376], and [I-D.ietf-softwire-dslite-multicast].

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 RFC 2119 [RFC2119].

3. Requirements

This section states requirements on 6rd multicast support protocol.

IPv6 hosts connected to 6rd CE router MUST be able to join multicast groups in IPv6 and receive multicast data.

Both any source multicast (ASM) and source specific multicast (SSM) MUST be supported.

6rd multicast MUST NOT introduce the need to use more IPv4 addresses, thereby contributing to public IPv4 address depletion.

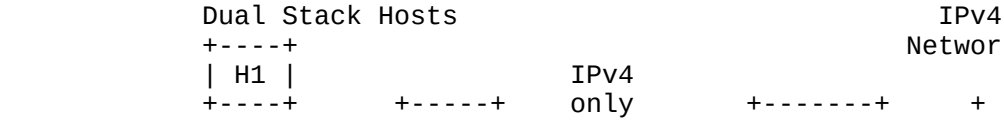
4. Architecture

In 6rd, IPv6 or IPv4/IPv6 dual stack hosts are served by the 6rd Customer Edge device (CE). The CE is dual stack facing the hosts and IPv4 only facing the network or WAN side. The CE tunnels IPv6 packets in IPv4 to the 6rd Border Relay (BR). The BR decapsulates the tunneled packets and forwards them to the IPv6 network. In the reverse direction, the BR receives IPv6 packets from the IPv6 network tunnels them in IPv4 to the CE. The CE decapsulates the IPv6 packets and forwards them to the hosts.

Unicast 6rd is stateless. Each IPv6 packet sent by the CE is treated separately and different packets from the same CE may go to different BRs. The CE encapsulates IPv6 packets in IPv4 with the IPv4 destination address set to the BR address (usually an anycast IPv4 address). BRs are placed where IPv6 native connectivity exists to other networks. A CE is configured with its own IPv4 address (public or private), with a 6rd IPv6 prefix from which the CE's IPv4 address can be derived, and with one or more BR IPv4 addresses. When the BR receives IPv6 packets addressed to the CE, it extracts the CE's IPv4 address from the destination IPv6 address and uses this address as the destination address for the IPv4 encapsulation of the IPv6 packet. 6rd views the IPv4-only network as an NBMA link from the IPv6 point of view and all 6rd CEs and BRs are defined as off-link neighbors from one other.

4.1. 6rd Tunneling Architecture

In order to support multicast, the CE implements an MLD Proxy function [RFC4605]. IPv6 hosts send their join requests (MLD Membership Report messages) to CE. The CE as a proxy sends aggregated Report messages upstream towards BR in unicast using IPv6 in IPv4 encapsulation.



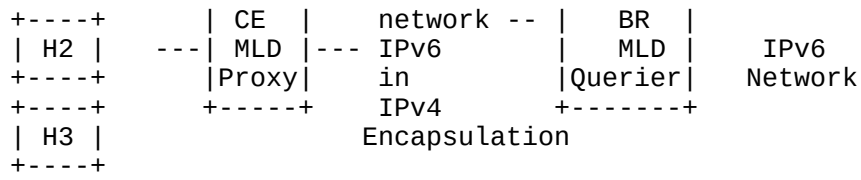


Figure 1: Architecture of 6rd Tunneling Multicast Protocol

The BR is the default multicast querier for the CE. The BR implements a multicast router function or it could be another MLD proxy.

All the elements of 6rd multicast support system are shown in Figure 1.

4.2. Translation Architecture

In order to support multicast, CE implements MLD Proxy [RFC4605] and MLD to IGMP interworking function [ID.perreault-igmp-mld-translation]. IPv6 hosts send their join requests (MLD Membership Report messages) to CE. CE as a proxy sends aggregated IGMP Report messages upstream towards BR.

In order to support SSM, MLDv2 [RFC3810] and IGMPv3 [RFC3376] must be supported by the CE and BR, and MLDv2 must be supported by the host.

The BR is the default multicast querier for the CE. The BR implements an IGMP to MLD interworking function and multicast router function or it could be another MLD proxy.

It is assumed that the IPv4 only network to which the CE and the BR are connected supports native IPv4 multicast.

All the elements of 6rd translation-based multicast support system are shown in Figure 2.

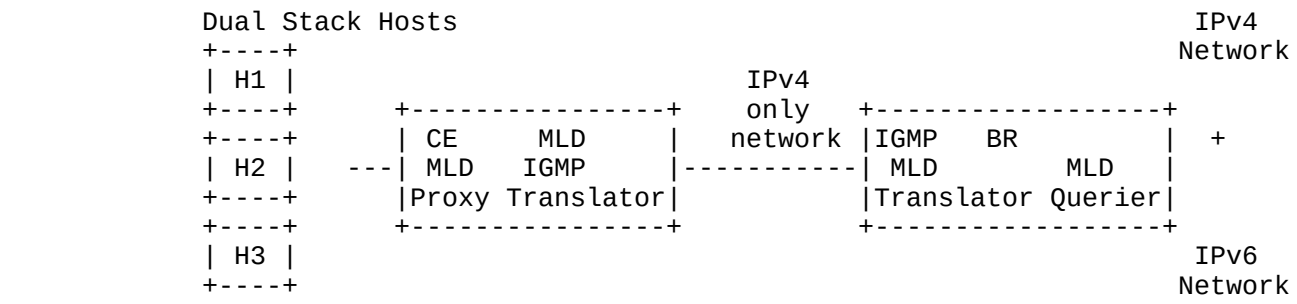


Figure 2: Architecture of 6rd Translation-Based Multicast

5. 6rd Tunneling Multicast Operation

In this section we specify how the host can subscribe and receive IPv6 multicast data from IPv6 content providers based on the architecture defined in Figure 1.

The hosts will send their subscription requests for IPv6 multicast groups upstream to the default router, i.e., Costumer Edge device. After subscribing the group, the host can receive multicast data from the CE. The host implements MLD protocol's host part.

The Customer Edge device is an MLD Proxy. After receiving the first MLD Report message requesting subscription to an IPv6 multicast group, the CE establishes a tunnel interface with a Border Relay. The tunnel is IPv4 based but it will carry MLD messages back and forth and IPv6 multicast data messages downstream.

The CE is a regular MLD proxy and it keeps an MLD proxy membership database. The CE inserts multicast forwarding state on the incoming interface, and merges state updates into the MLD proxy membership database. The CE updates or remove elements from the database as required. The CE will then send an aggregated Report via the upstream tunnel to the BR when the membership database changes.

The CE answers MLD queries from the BR based on the membership database. The CE's downstream link follows the traditional multipoint channel forwarding and does not pose any specific problems.

The CE receives IPv6 multicast data from the BR tunneled over the tunnel interface. The CE decapsulates the packet and then forwards it downstream. Each member host receives the data packet based on the Layer 2 multicast interface. No packet duplication is necessary.

The Border Relay acts as the default multicast querier for all CEs that have established an IPv4 tunnel with it. In order to keep a consistent multicast state between a CE and BR, once a CE is connected it will stay connected until the state becomes empty. After that point, the CE may establish another tunnel to a different BR.

According to aggregated MLD reports received from subtending CEs, the BR establishes group/source-specific multicast forwarding states at its corresponding downstream tunnel interfaces. After that, the BR maintains or removes the state as required by the aggregated reports received from CEs.

At the upstream interface, the BR procures for aggregated multicast membership maintenance. Based on the multicast-transparent operations of the CEs, the BR treats its tunnel interfaces as multicast enabled downstream links, serving zero to many listening nodes.

When the BR receives MLD join requests from downstream CEs, the BR sends PIM join messages upstream towards multicast source(s). This results in a multicast tree formation where the BR is at the leaf of the multicast tree, enabling the BR to receive IPv6 multicast data sent by the source.

Multicast traffic arriving at the BR is transparently forwarded according to its multicast forwarding information base. Multicast data is first replicated according to MLD multicast group state and then forwarded in IPv6-in-IPv4 tunnels from the BR to the corresponding CEs.

5.1. Tunnel Interface Considerations

IPv6 in IPv4 tunneling is performed as specified in [RFC4213]. Considerations specified in [RFC5969] apply. Packets passing upstream from the CE carry only MLD signaling messages and they are not expected to be fragmented. However packets downstream, i.e., multicast data to the CEs, may be subject to fragmentation.

Source and destination addresses of MLD messages in IPv6-in-IPv4 tunnel from CE are as follows:

- The source address of IPv4 header is the CE WAN interface IPv4 address. The destination address is the BR anycast address when an invite message is sent to group G. Subsequent messages to group G contain the BR unicast address as destination address.

- o The source address of the inner MLD message is the link local address. The destination address is all MLDv2-capable multicast routers or FF02::16 for MLD Version 2 Multicast Listener Reports.

The source and destination addresses of MLD messages in the IPv6- in-IPv4 softwire from BR are as follows:

- o The source address of the IPv4 header is the BR IPv4 unicast address. The destination address is the CE IPv4 address. This also holds for multicast data.

- o The source address of the inner MLD message is the link local address. The destination address is the link-scope all-nodes multicast address (FF02::1) for General Queries, or the IPv6 multicast group address for specific queries.

The source address of IPv6 multicast data is the unicast IPv6 address of the multicast source, e.g., the content provider. The destination address is the3 IPv6 multicast group address.

5.2. Avalanche Problem

In Section 5.1, multicast data is replicated to all interfaces, i.e., to all member CEs at the BR. This replication (often called avalanche problem) can be very costly if is a very large number of downstream member CEs such as in the IPTV application. See Appendix A in [I-D.ietf-softwire-dslite-multicast].

In 6rd tunneling multicast, the avalanche problem can be reduced by careful network partitioning. More BRs can be deployed in areas where IPv6 users are increasing in numbers. Deploying BRs by collocating them with the access network gateway as with the Border Network Gateway (BNG) is another possibility.

In the 6rd tunneling multicast operation, CEs are enabled to exploit multiple BRs that can be deployed in the network by using the BR anycast address any time they send an upstream MLD join request and then using the same BR that received the join message in subsequent MLD messages by using the same BR’s unicast address.

6. 6rd Translation Multicast Operation

In this section we specify how the host can subscribe and receive IPv6 multicast data from IPv6 content providers based on the architecture defined in Figure 2.

The hosts will send their subscription requests for IPv6 multicast groups upstream to the default router, i.e., the Costumer Edge

device. After subscribing the group, the host can receive multicast data from the CE. The host implements the MLD protocol's host part.

The Customer Edge device is an MLD Proxy. After receiving the first MLD Report message requesting subscription to an IPv6 multicast group, the CE interworks the MLD Membership Report message to an IGMP Membership report message. It sends it upstream only if joining a new group is needed.

Address translation in generating an IGMP Membership report message is done as follows: the destination address is copied from the last 32 bits of IPv6 multicast group address. The CE inserts the IPv4 address of its WAN interface into the source address. It is assumed that the IPv6 multicast group address in MLD Report message conforms to the addressing scheme described in [I-D.ietf-mboned-64-multicast-address-format], for any-source and source-specific multicast address formats.

Source addresses in the MLDv2 payload are translated as follows. Multicast source addresses in MLD Membership Report message MUST use uPrefix64, i.e. 64:ff9b::/96 defined in [RFC6052]. uPrefix64 facilitates translation into an IPv4 source address to be used in IGMPv3 Membership Report messages for source-specific multicast, i.e., by extracting the last 32 bits of IPv6 source address.

The IGMP Report message is received by the IGMP Querier/Proxy upstream on the link. (Normally this node is the Broadband Network Gateway, BNG in broadband networks.) The IGMP Querier/Proxy sends IGMPv3 Report message to the neighboring routers to join the group. In networks where PIM is supported, the IGMP Report message may be received by the PIM Designated Router. The PIM router sends a PIMv4 join message to join an IPv4 group.

The border router that receives the join message translates the message into MLD. To join an IPv6 group for any-source multicast, the IPv6 Multicast group address is obtained from the destination address. For source-specific multicast, the IPv6 source address is generated after obtaining the IPv4 source address of Membership Report message's Group Record Source Address field. The BR sends the PIMv6 join message upstream towards the source.

The BR MUST act as the designated router to which the source of the source-specific IGMP join message is connected. The BR MUST act as the rendez-vous point (RP) of the multicast group for the any-source multicast IGMP join message. Normally there is one such BR in an operator's network. An IPv4 multicast tree eventually forms in the network between the CE and BR and an IPv6 multicast tree upstream from the BR for the same ASM or SSM group.

IPv6 multicast data received at the border router from the source is translated into IPv4. The last 32 bits of the source and destination address fields determine the source and destination addresses of the IPv4 multicast data packet. This packet is sent downstream on the multicast tree already formed for this IPv4 multicast group.

Multicast data packet address translation follows the rules in [I-D.ietf-mboned-64-multicast-address-format] for the multicast group address and [RFC6052] for source- specific multicast source address, i.e. using uPrefix64. For any- source multicast, the Border Router inserts an IPv4 source address, different for each source.

Packet header translation follows the rules in [RFC6145]. Fragmentation and reassembly are handled as described in [RFC6145]. After the IPv4 multicast data packet is sent downstream from the BR it may be fragmented by the routers.

The CE receives the IPv4 multicast data packet, possibly in fragments, and reassembles the fragments. The CE translates the IPv4 multicast data packet back to an IPv6 multicast data packet. Address translation is done following [I-D.ietf-mboned-64-multicast-address-format] for multicast group addresses and [RFC6052] for unicast SSM source addresses. Header translation is done as in [RFC6145].

IPv6 multicast data is sent on the home link to the host(s). IEEE 802.3 or IEEE 802.11 multicast link support usually handles this delivery in Layer 2 without any packet duplication if there are more than one members to the any-source multicast group or SSM source and multicast group.

6.1. Solution Based on Layer 2 Multicast Support

In this section we assume that Layer 2 multicast is supported in the network. Layer 2 multicast support is done in order to forward multicast data downstream to the ports of Layer 2 devices, i.e. switches that requested a multicast group instead of flooding the data to all the ports.

In the switches, called snooping switches, multicast MAC address based filters are set up which link layer 2 multicast groups to the egress ports. IGMP snooping switches are commonly used in operators' networks, most commonly at the access nodes (AN) [RFC6788].

When an IGMP Report message is received, the bridge will set up a multicast filter entry that allows (in case of a join message) or prevents (in case of a leave message) packets to flow on the port on which the IGMP Report message was received. In terms of IPv4

multicast addresses, the mapping is not unique as 32 IPv4 multicast addresses map to a single Ethernet multicast MAC address [RFC4541].

The main functionality of a snooping switch is to forward multicast data packets based on the filters that are setup, i.e. to those egress ports with multicast groups downstream and also to the router ports.

In a 6rd network the snooping switches must detect IGMP packets sent upstream by the CE and set the filtering rules accordingly. When IPv4 data packets are received the IGMP snooping switches forward these packets towards all CEs that have members, effectively achieving packet duplication at the access node level.

6.2. Analysis

An analysis of the translation solution reveals the following:\

- The translation solution imposes a requirement on the IPv6 source-specific multicast sources to use uPrefix64 compatible source addresses. This requirement cannot be satified with simple configuration of the CPE router and Border Router.

- In the case of any-source multicast, the border router must use a public IPv4 address distinctively to represent each IPv6 any-source multicast source.

- In deployments which use IGMP routers, not PIM routers, source-specific multicast can be supported only if all routers have been upgraded to IGMPv3 and no IGMPv1 or IGMPv2 systems are present. Otherwise the operation reverts to the older version of IGMP to preserve compatibility and thus SSM can not be supported. With the use of PIM routers, this is avoided.

- The border router must act as the designated router or the rendez-vous point for the IPv4/IPv6 multicast group and this may lead to the use of a single border router in the network instead of load sharing with various border routers.

7. Security Considerations

6rd Translation Multicast control and data message security are as described in [RFC5969]. The threats and their mitigation described in [RFC5969] apply to multicast communication as well.

8. IANA Considerations

TBD.

9. Acknowledgements

We would like to specially thank Mark Townsley for his constructive comments. Steve Wright's online and very many offline comments helped us improve the document.

10. References

10.1. Normative References

[I-D.ietf-mboned-64-multicast-address-format]
Boucadair, M., Qin, J., Lee, Y., Venaas, S., Li, X., and M. Xu, "IPv4-Embedded IPv6 Multicast Address Format (Work in progress)", August 2012.

[I-D.ietf-mboned-auto-multicast]
Bumgardner, G., "Automatic Multicast Tunneling (work in progress)", June 2012.

[I-D.ietf-softwire-dslite-multicast]
Qin, J., Boucadair, M., Jacquenet, C., Lee, Y., and Q. Wang, "Delivery of IPv4 Multicast Services to IPv4 Clients over an IPv6 Multicast Network", draft-ietf-softwire-dslite-multicast-05 (work in progress), April 2013.

[ID.perreault-igmp-mld-translation]
Perrault, S. and T. Tsou, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Translation ("IGMP/MLD Translation") (Work in progress)", February 2013.

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

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, January 1999.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

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

[RFC4286] Haberman, B. and J. Martin, "Multicast Router Discovery", RFC 4286, December 2005.

[RFC4541] Christensen, M., Kimball, K., and F. Solensky, "Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches", RFC 4541, May 2006.

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC4605] Fenner, B., He, H., Haberman, B., and H. Sandick, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding ("IGMP /MLD Proxying")", RFC 4605, August 2006.

[RFC5569] Despres, R., "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd)", RFC 5569, January 2010.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

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

10.2. Informative References

[RFC3171] Albanna, Z., Almeroth, K., Meyer, D., and M. Schipper, "IANA Guidelines for IPv4 Multicast Address Assignments", RFC 3171, August 2001.

[RFC6788] Krishnan, S., Kavanagh, A., Varga, B., Ooghe, S., and E. Nordmark, "The Line-Identification Option", RFC 6788, November 2012.

Authors' Addresses

B. Sarikaya
Huawei Technologies (USA)
5340 Legacy Dr. Building 175
Plano, TX 75024
USA

Email: sarikaya@ieee.org

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

Phone: +1 408 330 4424
Email: Tina.Tsou.Zouting@huawei.com
URI: http://tinatsou.weebly.com/contact.html

Hui Ji
China Telecom
NO19.North Street
Beijing, Chaoyangmen,Dongcheng District
P.R. China

Email: jihui@chinatelecom.com.cn

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: cathy.zhou@huawei.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 10, 2015

B. Sarikaya
Huawei USA
H. Ji
China Telecom

June 8, 2015

Multicast Support for Mapping of Address and Port Protocol and Light Weight 4over6
draft-sarikaya-softwire-map-multicast-04

Abstract

This memo specifies multicast component for MAP and Light Weight 4over6 so that IPv4 hosts can receive multicast data from IPv4 servers over an IPv6 network. The solution developed is based on translation. In the Translation Multicast solution for MAP (MAP-E and MAP-T) and lw4o6, IGMP messages are translated into MLD messages and sent to the network in IPv6. MAP Border Relay/lwAFTR does the reverse translation and joins IPv4 multicast group for the hosts. Border Relay/lwAFTR as multicast router receives IPv4 multicast data and translates the packet into IPv6 multicast data and sends downstream on the multicast tree. Member CEs/lwB4s receive multicast data, translate it back to IPv4 and transmit to the hosts.

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 December 10, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Architecture 4
3.1. MAP-T and 4rd Translation Architecture 4
4. MAP-T and 4rd Translation Multicast Operation 7
4.1. Address Translation 7
4.2. Protocol Translation 9
4.3. Learning Multicast Prefixes for IPv4-embedded IPv6 Multicast Addresses 10
4.4. Supporting IPv4 Multicast at CE Router and lwB4 11
5. Security Considerations 11
6. IANA Considerations 11
7. Acknowledgements 12
8. References 12
8.1. Normative References 12
8.2. Informative references 14
Appendix A. Group Membership Message Translation Details 15
Authors' Addresses 16

1. Introduction

With IPv4 address depletion on the horizon, many techniques are being standardized for IPv6 migration including Mapping of Address and Port (MAP) - Encapsulation, - Translation and 4rd [I-D.ietf-softwire-map], [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd]. MAP/4rd enables IPv4 hosts to communicate with external hosts using IPv6 only ISP network. MAP/4rd Customer Edge (CE) device's LAN side is dual stack and WAN side is IPv6 only. CE tunnels/translates IPv4 packets received from the LAN side to 4rd Border Relays (BR). BRs have anycast IPv6 addresses and receive encapsulated/translated packets from CEs over a virtual interface. MAP/4rd operation is stateless. Packets are received/ sent independent of each other and no state needs to be maintained except for NAT44 operation on IPv4 packets received from the user.

Light Weight 4 Over 6 (lw4o6) is a variant of Dual Stack Lite where carrier grade NAT is moved from AFTR to B4 element, i.e. NAPT is done locally at each B4 called light weight B4 or lwB4. Unicast lw4o6 takes user IPv4 packets from the local LAN and lwB4 does a NAPT and then tunnels the packets in an IPv4-in-IPv6 tunnel to lwAFTR which decapsulates the packet and then sends it to IPv4 network. Incoming packets follow reverse route and are encapsulated at lwAFTR and sent to lwB4 which decapsulates and after NAPT operation transmits to the destination.

It should be noted that there is no depletion problem for IPv4 address space allocated for any source multicast and source specific multicast [RFC3171]. This document is not motivated by the depletion of IPv4 multicast addresses.

MAP-E, MAP-T, 4rd and lw4o6 are unicast only. They do not support multicast. In this document we specify how multicast from home IPv4 users can be supported in MAP-E (as well as MAP-T and 4rd) and lw4o6.

In case IPv6 network is multicast enabled, MAP-T/4rd can provide multicast service to the hosts using MAP-T/4rd Multicast Translation based solution. A Multicast Translator can be used that receives IPv4 multicast group management messages in IGMP and generates corresponding IPv6 group management messages in MLD and sends them to IPv6 network towards MAP-T/4rd Border Relay. We use [I-D.ietf-softwire-map-t] or [I-D.ietf-softwire-4rd] for sending IPv4 multicast data in IPv6 to the CE routers. At MAP-T/4rd CE router another translator is needed to translate IPv6 multicast data into IPv4 multicast data.

It should be noted that if IPv6 network is multicast enabled the translation multicast solution presented in Section 4 can also be used for MAP-E.

In this document we address MAP-E (and MAP-T/4rd) and lw4o6 multicast problem and propose the architecture of Multicast Translation based solution. Section 2 is on terminology, Section 3 is on architecture, Section 4 is on multicast translation protocol, and Section 5 states security considerations.

2. Terminology

This document uses the terminology defined in [I-D.ietf-softwire-map], [I-D.ietf-softwire-lw4over6], [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd], [RFC3810] and [RFC3376].

3. Architecture

In MAP-E, MAP-T and 4rd, there are hosts (possibly IPv4/ IPv6 dual stack) served by MAP-E, MAP-T and 4rd Customer Edge device. CE is dual stack facing the hosts and IPv6 only facing the network or WAN side. MAP-E, MAP-T and 4rd CE may be local IPv4 Network Address and Port Translation (NAPT) box [RFC3022] by assigning private IPv4 addresses to the hosts. MAP-E, MAP-T and 4rd CEs in the same domain may use shared public IPv4 addresses on their WAN side and if they do they should avoid ports outside of the allocated port set for NAPT operation. At the boundary of the network there is MAP-E, MAP-T and 4rd Border Relay. BR receives IPv4 packets tunneled in IPv6 from CE and decapsulates them and sends them out to IPv4 network.

Unicast MAP-E, MAP-T and 4rd are stateless except for the local NAPT at the CE. Each IPv4 packet sent by CE treated separately and different packets from the same CE may go to different BRs or CEs. CE encapsulates IPv4 packet in IPv6 with destination address set to BR address (usually anycast IPv6 address). BR receives the encapsulated packet and decapsulates and send it to IPv4 network. CEs are configured with Rule IPv4 Prefixes, Rule IPv6 Prefixes and with an BR IPv6 anycast address. BR receives IPv4 packets addressed to this ISP and from the destination address it extracts the destination host’s IPv4 address and uses this address as destination address and encapsulates the IPv4 packet in IPv6 and sends it to IPv6-only network.

Unicast Lightweight 4over6 (lw4o6) is a variation of Dual-Stack Lite (DS-Lite) [RFC6333] which moves carrier-grade IPv4-IPv4 NAT from the Address Family Transition Router (AFTR) element to the Basic Bridging BroadBand (B4) element [I-D.ietf-softwire-lw4over6]. The resulting elements are called lwAFTR and lwB4 with NAPT, respectively. Lw4o6 also adopts some features from MAP-E. A+P scheme of public IPv4 address sharing is used by lwB4’s in assigning WAN side IPv4 public addresses with a distinct port set. As in MAP-E, encapsulation of IPv4 packets in IPv6 and decapsulation is according to [RFC2473].

3.1. MAP-T and 4rd Translation Architecture

In case IPv6 only network is multicast enabled, translation multicast architecture can be used. CE implements IGMP Proxy function [RFC4605] towards the LAN side and MLD Proxy on its WAN interface. IPv4 hosts send their join requests (IGMP Membership Report messages) to CE. CE as a MLD proxy sends aggregated MLD Report messages upstream towards BR. CE replies MLD membership query messages with MLD membership report messages based on IGMP membership state in the IGMP/MLD proxy.

BR is MLD querier on its WAN side. On its interface to IPv4 network BR may either have IGMP client or PIM. PIM being able to support both IPv4 and IPv6 multicast should be preferred. BR receives MLD join requests, extracts IPv4 multicast group address and then joins the group upstream, possibly by issuing a PIM join message.

IPv4 multicast data received by the BR as a leaf node in IPv4 multicast distribution tree is translated into IPv6 multicast data by the translator using [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd] and then sent downstream to the IPv6 part of the multicast tree to all downstream routers that are members. IPv6 data packet eventually gets to the CE. At the CE, a reverse [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd] operation takes place by the translator and then IPv4 multicast data packet is sent to the member hosts on the LAN interface. [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd] are modified to handle multicast addresses.

In order to support SSM, IGMPv3 MUST be supported by the host, CE and BR. For ASM, BR MUST be the Rendezvous Point (RP).

MAP-T and 4rd Translation Multicast solution uses the multicast 46 translator in not one but two places in the architecture: at the CE router and at the Border Relay. IPv4 multicast data received at 4rd BR goes through a [I-D.ietf-softwire-4rd] header-mapping into IPv6 multicast data at the BR and another [I-D.ietf-softwire-4rd] header-mapping back to IPv4 multicast data at the CE router. Encapsulation variant of [I-D.ietf-softwire-4rd] is not used. In case of MAP-T, IPv4 data packet is translated using v4 to v6 header translation using multicast addresses instead of the mapping algorithm used in [I-D.ietf-softwire-map-t].

All the elements of MAP-T and 4rd translation-based multicast support system are shown in Figure 1.

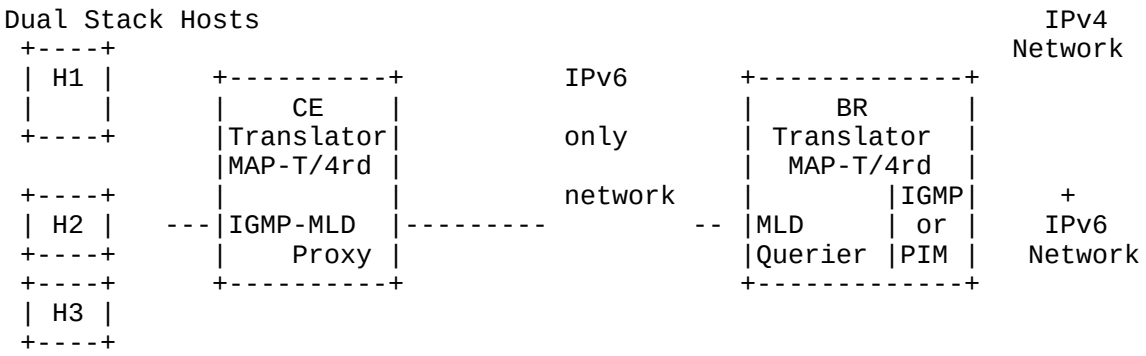


Figure 1: Architecture of MAP-T and 4rd Translation Multicast

In case IPv6 only network is multicast enabled, translation multicast architecture can also be used for lw4o6 multicast. lwB4 implements IGMP Proxy function [RFC4605] towards the LAN side and MLD Proxy on its WAN interface. IPv4 hosts send their join requests (IGMP Membership Report messages) to lwB4. lwB4 as a MLD proxy sends aggregated MLD Report messages upstream towards lwAFTR. lwB4 replies MLD membership query messages with MLD membership report messages based on IGMP membership state in the IGMP/MLD proxy.

lwAFTR is MLD querier on its WAN side. On its interface to IPv4 network lwAFTR may either have IGMP client or PIM. PIM being able to support both IPv4 and IPv6 multicast should be preferred. lwAFTR receives MLD join requests, extracts IPv4 multicast group address and then joins the group upstream, possibly by issuing a PIM join message.

For multicast data, [I-D.ietf-softwire-dslite-multicast] uses encapsulation of IPv4 multicast data in IPv6 multicast data packet but in this document we use translation. IPv4 multicast data received by the lwAFTR as a leaf node in IPv4 multicast distribution tree is translated into IPv6 multicast data by the translator and then sent downstream to the IPv6 part of the multicast tree to all downstream routers that are members. IPv6 data packet eventually gets to the lwB4. At the lwB4, a reverse translation operation takes place by the translator and then IPv4 multicast data packet is sent to the member hosts on the LAN interface. The translation algorithm in [I-D.ietf-softwire-map-t], [I-D.ietf-softwire-4rd] are modified to handle multicast addresses.

In order to support SSM, IGMPv3 MUST be supported by the host, lwB4 and lwAFTR. For ASM, lwAFTR MUST be the Rendezvous Point (RP).

MAP-T and 4rd Translation Multicast solution uses the multicast 46 translator in not one but two places in the architecture: at the lwB4 router and at the lwAFTR. IPv4 data packet is translated using v4 to v6 header translation using multicast addresses instead of the mapping algorithm used in [I-D.ietf-softwire-map-t].

All the elements of lw4o6 translation-based multicast support system are shown in Figure 2.

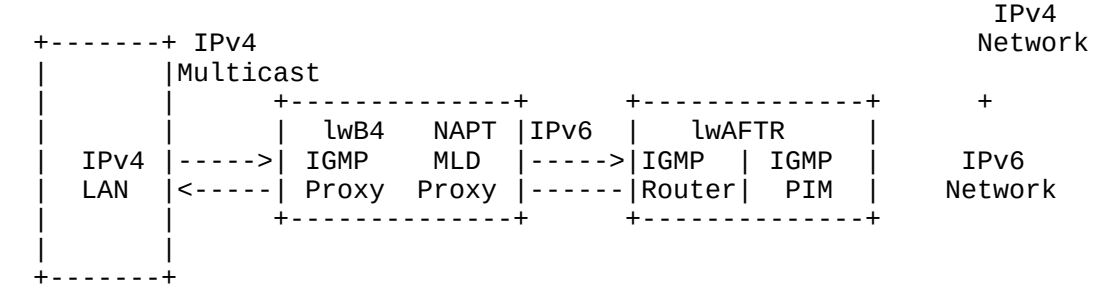


Figure 2: Architecture of lw4o6 Multicast Translation

4. MAP-T and 4rd Translation Multicast Operation

In this section we specify how the host can subscribe and receive IPv4 multicast data from IPv4 content providers based on the architecture defined in Figure 1 in two parts: address translation and protocol translation. Translation details are given in Appendix A.

4.1. Address Translation

IPv4-only host, H1 will join IPv4 multicast group by sending IGMP Membership Report message upstream towards the IGMP Proxy in Figure 1. MLD Proxy first creates a synthesized IPv6 address of IPv4 multicast group address using IPv4-embedded IPv6 multicast address format [I-D.ietf-mboned-64-multicast-address-format]. ASM_MPREFIX64 for any source multicast groups and SSM_MPREFIX64 for source specific multicast groups are used. Both are /96 prefixes.

SSM_MPREFIX64 is set to ff3x:0:8000::/96, with ’x’ set to any valid scope. ASM_MPREFIX64 values are formed as shown in Figure 3. Flag field 1 (ff1) field is defined in [RFC7371] bits M bit MUST BE set to 1. "scop" field is defined in [RFC3956]. Flag field 2 (ff2) is a set of 4 flags rrrM where r bits MUST be set to zero. M bit is set to 1 to indicate that a multicast IPv4 address is embedded in the low-

order 32 bits of the multicast IPv6 address. "sub-group-id" field MUST follow the recommendations specified in [RFC3306] if unicast-based prefix is used or the recommendations specified in [RFC3956] if embedded-RP is used. The default value is all zeros.

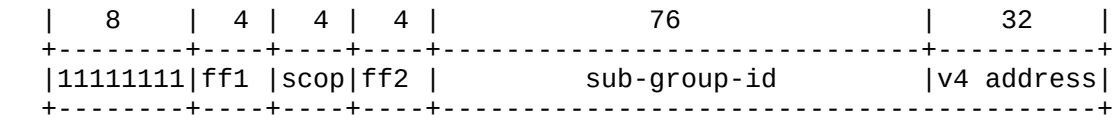


Figure 3: ASM_MPREFIX64 Formation

Each translator in the upstream BR is assigned a unique ASM_MPREFIX64 prefix. CE (MLD Proxy in CE) can learn this value by means out of scope with this document. With this, CE can easily create an IPv6 multicast address from the IPv4 group address a.b.c.d that the host wants to join.

Source-Specific Multicast (SSM) can also be supported similar to the Any Source Multicast (ASM) described above. In case of SSM, IPv4 multicast addresses use 232.0.0.0/8 prefix. IPv6 SSM_MPREFIX64 is set to FF3x:0:8000::/96 where 'x' is any valid scope.

Since SSM translation requires a unique address for each IPv4 multicast source, an IPv6 unicast prefix must be configured to the translator in the upstream BR to represent IPv4 sources. This prefix is prepended to IPv4 source addresses in translated packets.

The join message from the host for the group ASM_MPREFIX64:a.b.c.d or SSM_MPREFIX64:a.b.c.d or an aggregate join message will be received by MLD querier at the BR. BR as multicast anchor checks the group address and recognizes ASM_MPREFIX64 or SSM_MPREFIX64 prefix. It next checks the last 32 bits is an IPv4 multicast address in range 224/8 - 239/8. If all checks succeed, IGMPv4 Client joins a.b.c.d using IGMP on its IPv4 interface.

Joining IPv4 groups can also be done using PIM since PIM supports both IPv4 and IPv6. The advantage of using PIM is that there is no need to enable IGMP support in neighboring IPv4 routers. The advantage of using IGMP is that IGMP is a simpler protocol and it is supported by a wider range of routers. The use of PIM or IGMP is left as an implementation choice.

Address translation described above for MAP-T applies to lw4over6 multicast translation where the entities involved are lwB4 replaces Customer Edge device and lwAFTR replaces BR Figure 2.

4.2. Protocol Translation

The hosts will send their subscription requests for IPv4 multicast groups upstream to the default router, i.e. Costumer Edge device. After subscribing the group, the host can receive multicast data from the CE. The host implements IGMP protocol's host part.

Customer Edge device is IGMP Proxy facing the LAN interface. After receiving the first IGMP Report message requesting subscription to an IPv4 multicast group, a.b.c.d, MLD Proxy in the CE's WAN interface synthesizes an IPv6 multicast group address corresponding to a.b.c.d and sends an MLD Report message upstream to join the group.

When MAP-T or 4rd BR receives IPv4 multicast data for an IPv4 group a.b.c.d it [I-D.ietf-softwire-4rd] translates/encapsulates IPv4 packet into IPv6 multicast packet and sends it to IPv6 synthesized address corresponding to a.b.c.d using ASM_MPREFIX64 or SSM_MPREFIX64. The header mapping described in [I-D.ietf-softwire-4rd] Section 4.2 (using Table 1) is used except for mapping the source and destination addresses. In this document we use the multicast address translation described in Section 4.1 and propose it as a complementary enhancement to the translation algorithm in [I-D.ietf-softwire-4rd].

The IP/ICMP translation translates IPv4 packets into IPv6 using minimum MTU size of 1280 bytes (Section 4.3 in [I-D.ietf-softwire-4rd]) but this can be changed for multicast. Path MTU discovery for multicast is possible in IPv6 so 4rd BR can perform path MTU discovery for each ASM group and use these values instead of 1280. For SSM, a different MTU value MUST be kept for each SSM channel. Because of this 8 bytes added by IPv6 fragment header in each data packet can be tolerated.

Since multicast address translation does not preserve checksum neutrality, [I-D.ietf-softwire-4rd] translator/encapsulator at 4rd BR must however modify the UDP checksum to replace the IPv4 addresses with the IPv6 source and destination addresses in the pseudo-header which consists of source address, destination address, protocol and UDP length fields before calculating the new checksum.

IPv6 multicast data must be translated back to IPv4 at the 4rd CE (e.g. using Table 2 in Section 4.3 of [I-D.ietf-softwire-4rd]). Such a task is much simpler than the translation at 4rd BR because IPv6 header is much simpler than IPv4 header and IPv4 link on the LAN side of 4rd CE is a local link. The packet is sent on the local link to IPv4 group address a.b.c.d for IPv6 group address of ASM_MPREFIX64:a.b.c.d or SSM_MPREFIX64:a.b.c.d.

In case an IPv4 multicast source sends multicast data with the don't fragment (DF) flag set to 1, [I-D.ietf-softwire-4rd] header mapping sets the D bit in IPv6 fragment header before sending the packet downstream as in Fig. 3 in Section 4.3 of [I-D.ietf-softwire-4rd]. This feature of [I-D.ietf-softwire-4rd] preserves the semantics of DF flag at the BR and CE.

Because MAP-T/4rd is stateless, Multicast MAP-T/4rd should stay faithful to this as much as possible. Border Relay acts as the default multicast querier for all CEs that have established multicast communication with it. In order to keep a consistent multicast state between a CE and BR, CE MUST use the same IPv6 multicast prefixes (ASM_MPREFIX64/SSM_REFIX64) until the state becomes empty. After that point, the CE may obtain different values for these prefixes, effectively changing to a different 4rd BR.

Protocol translation described above for MAP-T applies to lw4over6 multicast translation where the entities involved are lwB4 replaces Customer Edge device and lwAFTR replaces BR Figure 2.

4.3. Learning Multicast Prefixes for IPv4-embedded IPv6 Multicast Addresses

CE can be pre-configured with Multicast Prefix64 of ASM_MPREFIX64 and SSM_MPREFIX64 that are supported in their network. However automating this process is also desired.

A new router advertisement option, a Multicast ASM Translation Prefix option, can be defined for this purpose. The option contains IPv6 ASM multicast translation prefix, ASM_MPREFIX64. A new router advertisement option, a Multicast SSM Translation Prefix option, can be defined for this purpose. The option contains IPv6 SSM multicast prefix translation prefix SSM_MPREFIX64.

After the host gets the multicast prefixes, when an application in the host wishes to join an IPv4 multicast group the host MUST use ASM_MPREFIX64 or SSM_MPREFIX64 and then obtain the synthesized IPv6 group address before sending MLD join message.

Source-specific multicast (SSM) group membership message payloads in IGMPv3 and MLDv2 contain address literals and their translation requires another multicast translation prefix option. IPv4 source addresses in IGMPv3 Membership Report message are unicast addresses of IPv4 sources and they have to be translated into unicast IPv6 source addresses in MLDv2 Membership Report message. A new router advertisement option, a Multicast Translation Unicast Prefix option can be defined for this purpose. The option contains IPv6 unicast Network-Specific Prefix U_PREFIX64. The host can be configured by

its default router using router advertisements containing the prefixes [I-D.sarikaya-softwire-6man-raoptions]. 64:ff9b::/96 is the global value called well-known prefix that is assigned to U_PREFIX64 [RFC6052]. Organization specific values called Network-Specific Prefixes can also be used. Since multicast is potentially inter-domain, the use of well-known prefix for U_PREFIX64 is recommended. DHCP servers can also configure hosts with ASM_MPREFIX64, SSM_MPREFIX64 and U_PREFIX64 as in [I-D.ietf-softwire-multicast-prefix-option].

Note that U_PREFIX64 is also used in multicast data packet address translation. Source-specific multicast source address in multicast data packets coming from SSM sources MUST be translated using U_PREFIX64.

4.4. Supporting IPv4 Multicast at CE Router and lwB4

When MAP-E CE router is a NAT or NAPT box assigning private IPv4 addresses to the hosts, IP Multicast requirements for a Network Address Translator (NAT) and a Network Address Port Translator (NAPT) stated in [RFC5135] apply to IGMP messages and IPv4 multicast data packets. The same applies to lwB4s in lw4over6.

On receiving multicast data packets, lwB4 or CE router MUST NOT modify destination IP address or destination port of the packets. Multicast UDP datagrams MUST be forwarded to the local LAN towards the host that is a member of this group.

IGMP membership reports received at lwB4 or CE router may be sent upstream individually for any source multicast but for source specific multicast, e.g. IGMPv3, membership reports MUST be sent after IGMP aggregation.

5. Security Considerations

Multicast control and data message security can be provided by the security architecture, mechanisms, and services described in [RFC4301], [RFC4302] and [RFC4303]. and in [RFC4607] for source specific multicast.

6. IANA Considerations

TBD.

7. Acknowledgements

TBD.

8. References

8.1. Normative References

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

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC1112] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, August 1989.

[RFC2113] Katz, D., "IP Router Alert Option", RFC 2113, February 1997.

[RFC2711] Partridge, C. and A. Jackson, "IPv6 Router Alert Option", RFC 2711, October 1999.

[RFC3171] Albanna, Z., Almeroth, K., Meyer, D., and M. Schipper, "IANA Guidelines for IPv4 Multicast Address Assignments", RFC 3171, August 2001.

[RFC4605] Fenner, B., He, H., Haberman, B., and H. Sandick, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding ("IGMP/MLD Proxying")", RFC 4605, August 2006.

[RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, August 2006.

[RFC3307] Haberman, B., "Allocation Guidelines for IPv6 Multicast Addresses", RFC 3307, August 2002.

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, January 1999.

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

[RFC2765] Nordmark, E., "Stateless IP/ICMP Translation Algorithm (SIIT)", RFC 2765, February 2000.

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

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

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

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC5135] Wing, D. and T. Eckert, "IP Multicast Requirements for a Network Address Translator (NAT) and a Network Address Port Translator (NAPT)", BCP 135, RFC 5135, February 2008.

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

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

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

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

[RFC7371] Boucadair, M. and S. Venaas, "Updates to the IPv6 Multicast Addressing Architecture", RFC 7371, September 2014.

[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-13 (work in progress), March 2015.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Qiong, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-13 (work in progress), November 2014.

[I-D.ietf-softwire-map-t]
Li, X., Bao, C., Dec, W., Troan, O., Matsushima, S., and T. Murakami, "Mapping of Address and Port using Translation (MAP-T)", draft-ietf-softwire-map-t-08 (work in progress), December 2014.

[I-D.ietf-softwire-4rd]
Despres, R., Jiang, S., Penno, R., Lee, Y., Chen, G., and M. Chen, "IPv4 Residual Deployment via IPv6 - a Stateless Solution (4rd)", draft-ietf-softwire-4rd-10 (work in progress), December 2014.

[I-D.ietf-mboned-64-multicast-address-format]
Boucadair, M., Qin, J., Lee, Y., Venaas, S., Li, X., and M. Xu, "IPv6 Multicast Address With Embedded IPv4 Multicast Address", draft-ietf-mboned-64-multicast-address-format-06 (work in progress), September 2014.

[I-D.ietf-softwire-multicast-prefix-option]
Boucadair, M., Qin, J., Tsou, T., and X. Deng, "DHCPv6 Option for IPv4-Embedded Multicast and Unicast IPv6 Prefixes", draft-ietf-softwire-multicast-prefix-option-08 (work in progress), March 2015.

8.2. Informative references

[RFC3306] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6 Multicast Addresses", RFC 3306, August 2002.

[RFC3956] Savola, P. and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", RFC 3956, November 2004.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[I-D.ietf-softwire-dslite-multicast]
Qin, J., Boucadair, M., Jacquenet, C., Lee, Y., and Q. Wang, "Delivery of IPv4 Multicast Services to IPv4 Clients over an IPv6 Multicast Network", draft-ietf-softwire-dslite-multicast-09 (work in progress), March 2015.

[I-D.sarikaya-softwire-6man-raoptions]
Sarikaya, B., "IPv6 RA Options for Translation Multicast Prefixes", draft-sarikaya-softwire-6man-raoptions-01 (work in progress), February 2013.

[I-D.perreault-mboned-igmp-mld-translation]
Perreault, S. and T. Tsou, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Translation ("IGMP/MLD Translation")", draft-perreault-mboned-igmp-mld-translation-01 (work in progress), April 2012.

Appendix A. Group Membership Message Translation Details

IGMP Report messages (IGMP type number 0x12 and 0x16, in IGMPv1 and IGMPv2 and 0x22 in IGMPv3) are translated into MLD Report messages (MLDv1 ICMPv6 type number 0x83 and MLDv2 type number 0x8f). IGMP Query message (IGMP type number 0x11) is translated into MLD Query message (ICMPv6 type number 0x82) [I-D.perreault-mboned-igmp-mld-translation].

Destination address in ASM, i.e. IGMPv1, IGMPv2 and MLDv1, is the multicast group address so the destination address in IGMP message is translated into the destination address in MLD message using [I-D.ietf-mboned-64-multicast-address-format].

Destination address in SSM, i.e. IGMPv3 and MLDv2 is translated as follows: it could be all nodes on link, which has the value of 224.0.0.1 (IGMPv3) and ff02::1 (MLDv2), all routers on link, which has the value of 224.0.0.2 (IGMPv3) and ff02::2 (MLDv2), all IGMP/MLD-capable routers on link, which has the value of 224.0.0.22 (IGMPv3) and ff02::16 (MLDv2).

Source address of MLD message that CE sends is set to link-local IPv6 address of CE’s WAN side interface. Source address of MLD message that BR sends is set to link-local IPv6 address of BR’s downstream interface.

Multicast Address or Group Address field in IGMP message payloads is translated using [I-D.ietf-mboned-64-multicast-address-format] as described above into the corresponding field in MLD message.

Source Address in IGMPv3 message payloads is translated using U_PREFIX64, the IPv6 unicast prefix to be used by SSM source. [RFC6052] defines in Section 2.3 the address translation algorithm of embedding an IPv4 source address and obtaining an IPv6 source address using a network specific prefix like U_PREFIX64. At the BR on its upstream interface or at the CE on its LAN interface, IPv4 addresses are extracted from the IPv4-embedded IPv6 addresses.

Maximum Response Time (MRT) field in IGMPv2 and IGMPv3 queries are translated into Maximum Response Delay (MRD) in MLDv1 and MLDv2 queries, respectively. In the corresponding MLD message, MRD is set to 100 times the value of MRT. At the BR on its upstream interface or at the CE on its LAN interface, MRT value is obtained by dividing MRD into 100 and rounding it to the nearest integer.

IGMP messages are sent with a Router Alert IPv4 option [RFC2113]. The translated MLD message are sent with a Router Alert option in a Hop-By-Hop IPv6 extension header [RFC2711]. In both cases, 2-octet value is set to 0.

Authors' Addresses

Behcet Sarikaya
Huawei USA
5340 Legacy Dr. Building 175
Plano, TX 75024

Email: sarikaya@ieee.org

Hui Ji
China Telecom
NO19.North Street
Beijing, Chaoyangmen,Dongcheng District
P.R. China

Email: jihui@chinatelecom.com.cn

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 15, 2014

Q. Sun
C. Xie
China Telecom
Y. Lee
Comcast
M. Chen
FreeBit
July 14, 2013

Deployment Considerations for Lightweight 4over6
draft-sun-softwire-lightweigh-4over6-deployment-04

Abstract

Lightweight 4over6 is a mechanism which moves the translation function from tunnel lwAFTR (AFTR) to lwB4s (B4s), and hence reduces the mapping scale on the lwAFTR to per-customer level. This document discusses various deployment models of Lightweight 4over6. It also describes the deployment considerations and applicability of the Lightweight 4over6 architecture.

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 January 15, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Requirements Language 4
3. Deployment Model 5
4. Overall Deployment Considerations 7
4.1. Addressing and Routing 7
4.2. Port-set Management 7
4.3. lwAFTR Discovery 8
4.4. Impacts on Accouting 8
5. lwAFTR Deployment Consideration 9
5.1. Logging at the lwAFTR 9
5.2. MTU and Fragmentation Considerations 9
5.3. Reliability Considerations of lwAFTR 9
5.4. Placement of AFTR 10
5.5. Port set algorithm consideration 10
5.6. Path Consistency Consideration 10
6. lwB4 Deployment Consideration 12
6.1. NAT traversal issue 12
6.2. Static Port Forwarding Configuration 12
7. DS-Lite Compatibility Consideration 13
7.1. Case 1: Integrated Network Element with Lightweight 4over6 and DS-Lite AFTR Scenario 13
7.2. Case 2: DS-Lite Coexistent scenario with Separated AFTR . 14
8. Acknowledgement 15
9. References 16
Appendix 1. Appendix:Experimental Result 19
1.1. Experimental environment 19
1.2. Experimental results 20
1.3. Conclusions 21
Authors' Addresses 22

1. Introduction

Lightweight 4over6 [I-D.ietf-softwire-lw4over6] is an extension to DS-Lite which simplifies the AFTR module [RFC6333] with distributed NAT function among B4 elements. The lwB4 in Lightweight 4over6 is provisioned with an IPv6 address, an IPv4 address and a port-set. It performs NAPT on end user's packets with the provisioned IPv4 address and port-set. IPv4 packets are forwarded between the lwB4 and the lwAFTR over a Softwire using IPv4-in-IPv6 encapsulation. The lwAFTR maintains one mapping entry per subscriber with the IPv6 address, IPv4 address and port-set. Therefore, this extension removes the NAT44 module from the AFTR and replaces the session-based NAT table to a per-subscriber based mapping table. This should relax the requirement to create dynamic session-based log entries. This mechanism preserves the dynamic feature of IPv4/IPv6 address binding as in DS-Lite, so it has no coupling between IPv6 address and IPv4 address/port-set as any full stateless solution ([RFC6052] or [I-D.ietf-softwire-map]) requires. This document discusses deployment models of Lightweight 4over6. It also describes the deployment considerations and applicability of the Lightweight 4over6 architecture.

Terminology of this document follows the definitions and abbreviations of [I-D.ietf-softwire-lw4over6].

2. Requirements Language

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. Deployment Model

Lightweight 4over6 is suitable for operators who would like to free any correlation of the IPv6 address with IPv4 address and port-set (or port-range). In comparison to full stateless solutions like MAP [I-D.ietf-softwire-map] and 4rd [I-D.ietf-softwire-4rd], Lightweight 4over6 frees address planning of IPv6 delegation for CPE from mapping rule administration and management in the network. Thus, IPv6 addressing is completely flexible to fit other deployment requirements, e.g., auto-configuration, service classification, user management, QoS support, etc. The philosophy here is that bits of IPv6 address should be left for IPv6 usage first.

Lightweight 4over6 can be deployed in a residential network (depicted in Figure1). In this scenario, a lwB4 would acquire an IPv4 address and a port-set after a successful user authentication process and IPv6 provisioning process. Then, it establishes an IPv4-in-IPv6 softwire using the IPv6 address to deliver IPv4 services to its connected host via the lwAFTR in the network. The lwB4 can act as a CPE, or software located in the host. The lwAFTR supports Lightweight 4over6 which keeps the mapping between lwB4's IPv6 address and its allocated IPv4 address + port set. The supporting system may keep the binding information as well for logging and user management.

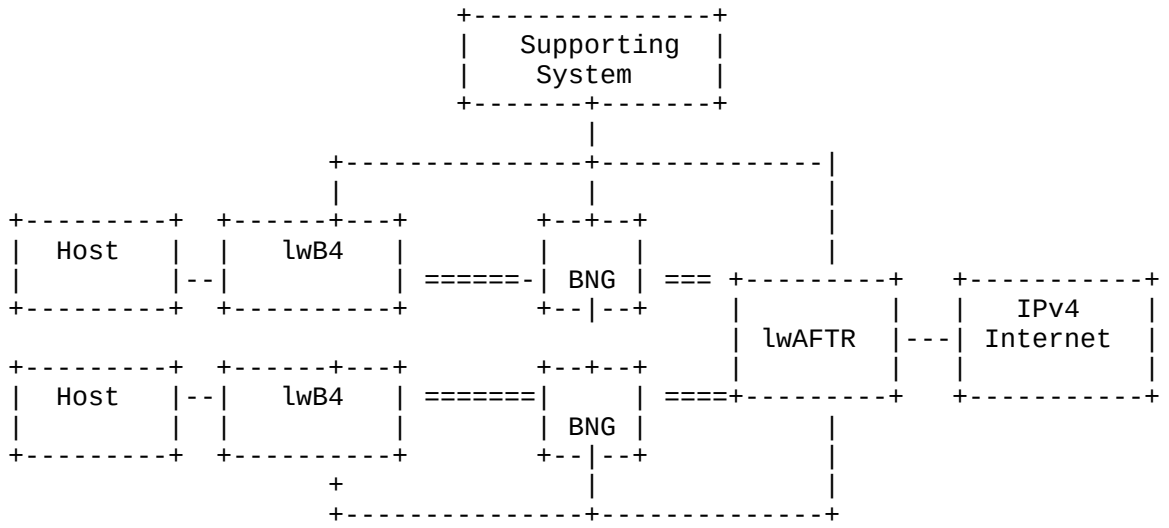


Figure 1 Deployment Model

There are two deployment models in practice: one is called bottom-up and the other is top-down. In bottom-up model, after port-restricted

IPv4 address is allocated to a given subscriber, the lwAFTR will report mapping records to the supporting system on creating a binding for traffic logging if necessary. Operators may use [I-D.ietf-behave-syslog-nat-logging] or [I-D.ietf-behave-ipfix-nat-logging] to report the port set allocated by lwAFTR. In this way, the lwAFTR can determine the binding by its own and there is little impact on existing network architecture. In top-down model, the Supporting system should firstly determine the binding information for each subscriber and then synchronize it with the lwAFTR. With this method, one binding record can be easily synchronized with multiple lwAFTRs and stateless failover can be achieved. However, new mechanism (e.g. [I-D.zhou-dime-4over6-provisioning]) needs to be introduced to notify each individual binding record between the Supporting system and the lwAFTR.

4. Overall Deployment Considerations

4.1. Addressing and Routing

In Lightweight 4over6, there is no inter-dependency between IPv4 and IPv6 addressing schemes. IPv4 address pools are configured centralized in lwAFTR for IPv6 subscribers. These IPv4 prefix must advertise to IPv4 Internet accordingly.

For IPv6 addressing and routing, there are no additional addressing and routing requirements. The existing IPv6 address assignment and routing announcement should not be affected. For example, in PPPoE scenario, a CPE could obtain a prefix via prefix delegation procedure, and the hosts behind CPE would get its own IPv6 addresses within the prefix through SLAAC or DHCPv6 statefully. This IPv6 address assignment procedure has nothing to do with restricted IPv4 address allocation.

4.2. Port-set Management

In Lightweight 4over6, each lwB4 will get its restricted IPv4 address and a port-set after successful user authentication process and IPv6 provisioning process. This port-set assignment can been achieved by DHCPv4-over-DHCPv6 [I-D.ietf-dhc-dhcpv4-over-dhcpv6] and PCP [I-D.ietf-pcp-port-set].

Operator may use DHCPv4 to provision IPv4 address to the lwB4. In a typical deployment, the DHCP server is a centralized DHCP server and lwAFTR is the DHCP relay agent to relay the dhcp messages to the server over unicast. Rarely DHCP server will collocate with the lwAFTR to provision IPv4 resources to the lwB4.

Operator may also use PCP Port-set Option to provision IPv4 address and port-set to the lwB4. In a typical deployment, PCP server will collocate with lwAFTR, and the subscriber’s binding can be determined by lwAFTR. The PCP request should be sent to the lwAFTR’s tunnel end-point address. It is not common that PCP server will be centralized deployed in which the lwAFTR is the PCP proxy to relay PCP requests.

It is also possible that subscriber’s binding is determined in AAA server. In this case, the BNGs will embed with a DHCPv4-over-DHCPv6 server function which allows them to locally handle any DHCPv4-over-DHCPv6 requests initiated by hosts. The AAA server will pass the subscriber’s binding to a BNG using the AAA attribute in [I-D.sun-softwire-lw4over6-radext] and in turn populates the mapping of the lwB4.

Some operators may offer different service level agreements (SLA) to users that some users may require more ports then others. In this deployment scenario, the operator can implement differentiated policies in provisioning system specified to a user's lwB4 or a group of lwB4s to allocate a certain range of port-set. The lwAFTR may also run multiple instances with different port-set sizes to build the mapping table.

4.3. lwAFTR Discovery

A Lightweight 4over6 lwB4 must discover the lwAFTR's IPv6 address before offering any IPv4 services. This IPv6 address can be learned through an out-of-band channel, static configuration, or dynamic configuration. In practice, Lightweight 4over6 lwB4 can use the same DHCPv6 option [RFC6334]to discover the FQDN of the lwAFTR.

When Lightweight 4over6 is deployed in the same place with DS-Lite, either different FQDNs can be configured for Lightweight 4over6 and DS-Lite separately or different DHCPv6 options can be used for Lightweight 4over6 [I-D.sun-softwire-lw4over6-dhcpv6] and DS-Lite. More detailed considerations on DS-Lite compatibility will be discussed in Section6.

4.4. Impacts on Accouting

In Lightweight 4over6, the accounting impact due to the tunneling protocol is the same with DS-Lite (see section 6.2 of [RFC6908]). However, since in Lightweight 4over6, the IPv4 service is only available after port-set allocation, if operators will regard IPv4 service as a on-demand value-added service, e.g. IPv6 connectivity is offered by default, while IPv4 connectivity will be offered untill a subscriber requires, etc., IPv4 service accounting should start after port-set allocation has completely.

5. lwAFTR Deployment Consideration

As Lightweight 4over6 is an extension to DS-Lite, both technologies share similar deployment considerations. For example: Interface consideration, Lawful Intercept Considerations, Blacklisting a shared IPv4 Address, AFTR's Policies, AFTR Impacts on Accounting Process, etc., in [RFC6908] can also be applied here. This document only discusses new considerations specific to Lightweight 4over6.

5.1. Logging at the lwAFTR

In Lightweight 4over6, operators only log one entry per subscriber. The log should include subscriber's IPv6 address used for the softwire, the public IPv4 address and the port-set. The port set algorithm implemented in Lightweight 4over6 lwAFTR should be synchronized with the one implemented in logging system. For example, if contiguous port set algorithm is adopted in the lwAFTR, the same algorithm should also been applied to the logging system.

Since the mapping in lwAFTR does not contain destination-specific information, operator should be aware that they will not be able to have destination-specific log.

5.2. MTU and Fragmentation Considerations

As Lightweight 4over6 is also a tunneling protocol, the same consideration regarding to the fragmentation and reassembly in DS-Lite [RFC6908] can also be applied. The only difference is that NAT functionality has been removed to lwB4 from lwAFTR in Lightweight 4over6. Therefore, on receiving an IPv4 fragmented packet after decapsulation in the lwB4, the lwB4 should further re-assemble the packets before doing NAT since the transport protocol information is only available in the first fragment.

5.3. Reliability Considerations of lwAFTR

Operators may deploy multiple lwAFTRs for robustness, reliability, and load balancing. In Lightweight 4over6, subscriber to IPv4 and port-set mapping must be pre-provisioned in the lwAFTR before providing IPv4 serives. For redundancy, the backup lwAFTR must either have the subscriber mapping already provisioned or notify the lwB4 to create a new mapping in the backup lwAFTR. The first option can be considered as Hot Standby mode, which requires state syncronization between multiple lwAFTRs. In Hot Standby mode, the bindings are replicated on-the-fly from the Primary lwAFTR to the Backup lwAFTR. When the Primary lwAFTR fails, the Backup lwAFTR will take over all the existing established sessions. In this mode, the internal hosts are not required to re-initiate the bindings with the

external hosts. In Lightweight 4over6, since the number of mapping states has been greatly reduced compared to DS-Lite, it is reasonable to adopt Hot Standby mode when there are only two lwAFTRs (one for Primary lwAFTR and one for Backup lwAFTR). However, if the number of lwAFTRs is larger than two, it is not scalable to deploy Hot Standby mode since each two of the lwAFTRs should to syncronize the binding states.

The second option is to use Cold Standby mode which does not require a Backup Standby lwAFTR to synchronize binding states. In failover, the lwAFTR has to notify the lwB4 to create a new binding, or fetch the binding by itself. [I-D.lee-softwire-lw4over6-failover] describes these two approaches for simple Cold Standby mode. For most deployment scenarios, we believe that Cold Standby mode should be sufficient enough and is thus recommended.

5.4. Placement of AFTR

The lwAFTR can be deployed in a "centralized model" or a "distributed model".

In the "centralized model", the lwAFTR could be located at the higher place, e.g. at the exit of MAN, etc. Since the lwAFTR has good scalability and can handle numerous concurrent sessions, we recommend to adopt the "centralized model" for Lightweight 4over6 as it is cost-effective and easy to manage.

In the "distributed model", lwAFTR is usually integrated with the BRAS/SR. Since newly emerging customers might be distributed in the whole Metro area, we have to deploy lwAFTR on all BRAS/SRs. This will cost a lot in the initial phase of the IPv6 transition period.

5.5. Port set algorithm consideration

If each lwB4 is given a set of ports, port randomization algorithm can only select port in the given port-set. This may introduce security risk because hackers can make a more predictable guess of what port a subscriber may use. Therefore, non-continuous port set algorithms (e.g. as defined in [I-D.ietf-softwire-map]) can be used to improve security.

5.6. Path Consistency Consideration

In Lightweight 4over6, if the binding state is not syncronized among multiple lwAFTRs, the lwAFTR in which the subscriber’s binding state is stored should be exactly the one to service the subscriber. Otherwise, there will be no match in lwAFTR. This requires the provionsion packets (either using DHCPv4-over-DHCPv6 or PCP Port-set)

should arrive at the same lwAFTR as the subsquent IP-in-IP traffic. If multiple lwAFTRs are using the same Tunnel End Point address and there are intermediate routers between lwB4 and lwAFTR, there might be a problem when intermediate routers perform ECMP based on L4 hash for the plain provionsion packets while doing L3 hash for subsequent IP-in-IP traffic. In this case, it is recommended that the privioning packet is sent over IPv6 tunnel so that intermediate routers can only process ECMP using L3 hash.

6. lwB4 Deployment Consideration

For lwB4 consideration, the DNS Deployment Considerations and B4 Remote Management in [RFC6908] can also be applied here. In this section, we only describe the considerations sepcific to Lightweight 4over6.

6.1. NAT traversal issue

In Lightweight 4over6, since the subscriber's source port will be restricted to the port-set allocated from the provisioning system, this will have impact on some NAT traversal mechanisms. For example, in UPnP 1.0, the external port number which can be used by remote peer is selected by UPnP client in end host. If the client randomly selects a port number which is not in that valid port-set, the UPnP process will fail. This is likely to happen because end-host does not know the port-set in lwB4. More detailed experimental results can be found in [I-D.deng-aplusp-experiment-results]. This problem will not exist in UPnP 2.0 because the UPnP client in the end-host will negotiate the external port number with the server. Another way is to implement a mechanism (e.g. [I-D.ietf-pcp-port-set], etc.) in end host to fetch the port-set from lwB4. The UPnP client can then select the port number within the port-set.

6.2. Static Port Forwarding Configuration

Currently, some external initiated applications rely on manual port configuration to reserve a port in the CPE. The restricted port-set in lwB4 will also have impacts on manual port forwarding configuration. It is recommended that the port-set allocated from the provioning system should be shown explicitly in the lwB4, which can be used as a hint for subscribers to add port forwarding mapping.

7. DS-Lite Compatibility Consideration

Lightweight 4over6 can be either deployed all alone, or combined with DS-Lite [RFC6333]. Since Lightweight 4over6 does not any have extra requirement on IPv6 addressing, it can use use the same addressing scheme with DS-Lite, together with routing policy, user management policy, etc. Besides, the bottom-up model has quite similar requirement and workflow on the supporting system with DS-Lite. Therefore, it is suitable for operators to deploy incrementally in existing DS-Lite network

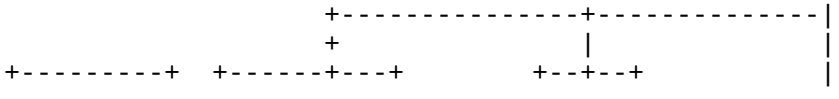
7.1. Case 1: Integrated Network Element with Lightweight 4over6 and DS-Lite AFTR Scenario

In this case, DS-Lite has been deployed in the network. Later in the deployment schedule, the operator decided to implement Lightweight 4over6 lwAFTR function in the same network element(depicted in Figure2). Therefore, the same network element needs to support both transition mechanisms.

There are two options to distinguish the traffic from two transition mechanisms.

The first one is to distinguish using the client's source IPv4 address. The IPv4 address from Lightweight 4over6 is public address as NAT has been done in the lwB4, and IPv4 address for DS-lite is private address as NAT will be done on AFTR. When the network element receives an encapsulated packet, it would de-capsulate packet and apply the transition mechanism based on the IPv4 source address in the packet. This requires the network element to examine every packet and may introduce significant extra load to the network element. However, both the B4 element and Lightweight 4over6 lwB4 can use the same DHCPv6 option [RFC6334] with the same FQDN of the AFTR and lwAFTR.

The second one is to distinguish using the destination's tunnel IPv6 address. One network element can run separated instances for Lightweight 4over6 and DS-Lite with different tunnel addresses. Then B4 element and Lightweight 4over6 lwB4 can use the same DHCPv6 option [RFC6334] with different FQDNs pointing to corresponding tunnel addresses. This requires the supporting system should distinguish different types of users when assigning the FQDNs in DHCPv6 process. Another option is to use a new DHCPv6 option [I-D.sun-softwire-lw4over6-dhcpv6] to discover lwAFTR's FQDN.



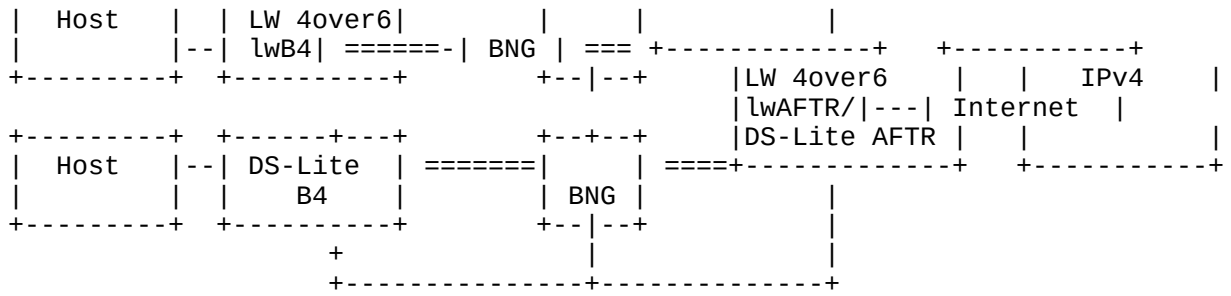


Figure 2 DS-Lite Coexistence scenario with Integrated AFTR

7.2. Case 2: DS-Lite Coexistent scenario with Separated AFTR

This is similar to Case 1. The difference is the lwAFTR and AFTR functions won't be co-located in the same network element (depicted in Figure3). This use case decouples the functions to allow more flexible deployment. For example, an operator may deploy AFTR closer to the edge and lwAFTR closer to the core. Moreover, it does not require the network element to pre-configure with the CPE's IPv6 addresses. An operator can deploy more AFTR and lwAFTR at needed. However, this requires the B4 and lwB4 to discover the corresponding network element. In this case, B4 element and Lightweight 4over6 lwB4 can still use [RFC6334] with different FQDNs pointing to corresponding tunnel end-point addresses, and the supporting system should distinguish different types of users.

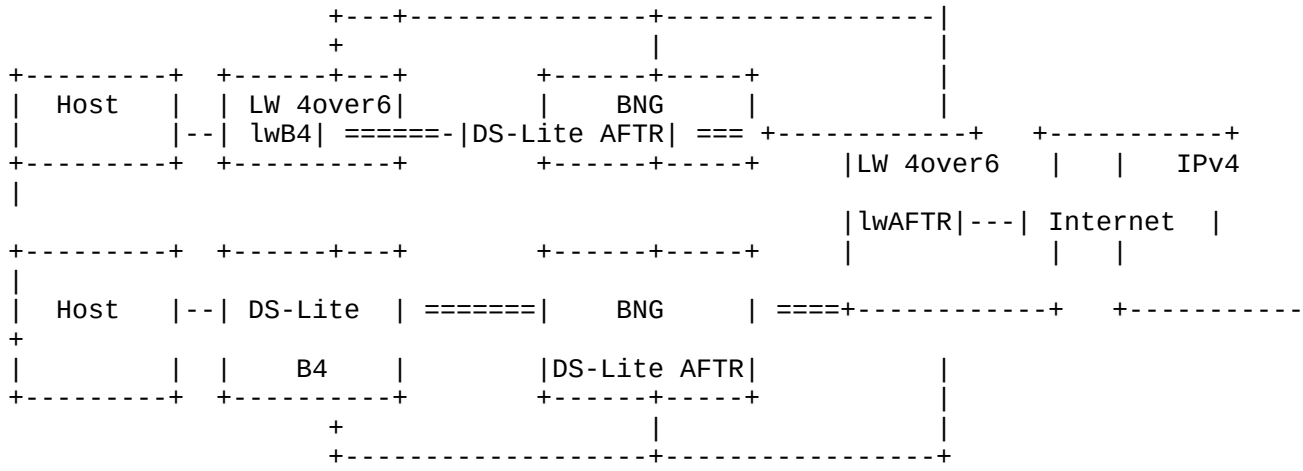


Figure 3 DS-Lite Coexistence scenario with Seperated AFTR

8. Acknowledgement

TBD

9. 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.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.deng-aplusp-experiment-results]
Deng, X., Boucadair, M., and F. Telecom, "Implementing A+P in the provider's IPv6-only network", draft-deng-aplusp-experiment-results-00 (work in progress), March 2011.

[I-D.ietf-behave-ipfix-nat-logging]
Sivakumar, S. and R. Penno, "IPFIX Information Elements for logging NAT Events", draft-ietf-behave-ipfix-nat-logging-00 (work in progress), March 2013.

[I-D.ietf-behave-syslog-nat-logging]
Chen, Z., Zhou, C., Tsou, T., and T. Taylor, "Syslog Format for NAT Logging", draft-ietf-behave-syslog-nat-logging-01 (work in progress), May 2013.

[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-ipv6-06 (work in progress), March 2013.

[I-D.ietf-pcp-base]
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-29 (work in progress), November 2012.

[I-D.ietf-pcp-port-set]
Sun, 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-01 (work in progress), May 2013.

[I-D.ietf-softwire-4rd]
Despres, R., Jiang, S., Penno, R., Lee, Y., Chen, G., and M. Chen, "IPv4 Residual Deployment via IPv6 - a Stateless Solution (4rd)", draft-ietf-softwire-4rd-06 (work in progress), July 2013.

[I-D.ietf-softwire-dslite-deployment]
Lee, Y., Maglione, R., Williams, C., Jacquenet, C., and M. Boucadair, "Deployment Considerations for Dual-Stack Lite", draft-ietf-softwire-dslite-deployment-08 (work in progress), January 2013.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 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-07 (work in progress), May 2013.

[I-D.lee-softwire-lw4over6-failover]
Lee, Y., Sun, Q., and C. Liu, "Simple Failover Mechanism for Lightweight 4over6", draft-lee-softwire-lw4over6-failover-00 (work in progress), July 2013.

[I-D.sun-softwire-lw4over6-dhcpv6]
Xie, C., Sun, Q., Lee, Y., Tsou, T., and P. Wu, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Lightweight 4over6", draft-sun-softwire-lw4over6-dhcpv6-00 (work in progress), July 2013.

[I-D.zhou-dime-4over6-provisioning]
Zhou, C. and T. Taylor, "Attribute-Value Pairs For Provisioning Customer Equipment Supporting IPv4-Over-IPv6 Transitional Solutions", draft-zhou-dime-4over6-provisioning-00 (work in progress), March 2013.

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

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052,

October 2010.

[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.

[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.

[RFC6908] Lee, Y., Maglione, R., Williams, C., Jacquenet, C., and M. Boucadair, "Deployment Considerations for Dual-Stack Lite", RFC 6908, March 2013.

1. Appendix:Experimental Result

We have deployed Lightweight 4over6 in our operational network of HuNan province, China. It is designed for broadband access network, and different versions of lwB4 have been implemented including a linksys box, a software client for windows XP, vista and Windows 7. It can be integrated with existing dial-up mechanisms such as PPPoE, etc. The major objectives listed below aimed to verify the functionality and performance of Lightweight 4over6:

- Verify how to deploy Lightweight 4over6 in a practical network.
- Verify the impact of applications with Lightweight 4over6.
- Verify the performance of Lightweight 4over6.

1.1. Experimental environment

The network topology for this experiment is depicted in Figure 2.

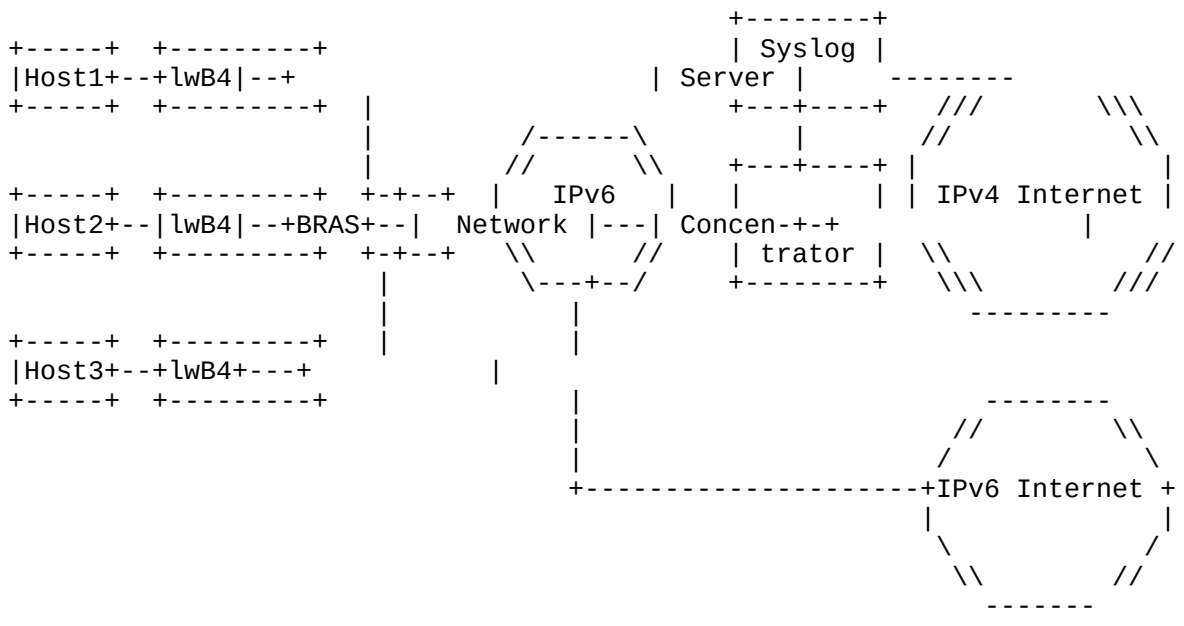


Figure 2 Lightweight 4over6 experiment topology

In this deployment model, lwAFTR is co-located with a extended PCP server to assign restricted IPv4 address and port set for lwB4. It also triggers subscriber-based logging event to a centrilized syslog server. IPv6 address pools for subscribers have been distributed to

BRASs for configuration, while the public available IPv4 address pools are configured by the centralized lwAFTR with a default address sharing ratio. It is rather flexible for IPv6 addressing and routing, and there is little impact on existing IPv6 architecture.

In our experiment, lwB4 will firstly get its IPv6 address and delegated prefix through PPPoE, and then initiate a PCP-extended request to get public IPv4 address and its valid port set. The lwAFTR will thus create a subscriber-based state accordingly, and notify syslog server with {IPv6 address, IPv4 address, port set, timestamp}.

1.2. Experimental results

In our trial, we mainly focused on application test and performance test. The applications have widely include web, email, Instant Message, ftp, telnet, SSH, video, Video Camera, P2P, online game, voip and so on. For performance test, we have measured the parameters of concurrent session numbers and throughput performance.

The experimental results are listed as follows:

Application Type	Test Result	Port Number Occupation
Web	ok IE, Firefox, Chrome	normal websites: 10~20 Ajex Flash webs: 30~40
Video	ok, web based or client based	30~40
Instant Message	ok QQ, MSN, gtalk, skype	8~20
P2P	ok utorrent,emule,xunlei	lower speed: 20~600 (per seed) higher speed: 150~300
FTP	need ALG for active mode, flashxp	2
SSH, TELNET	ok	1 for SSH, 3 for telnet
online game	ok for QQ, flash game	20~40

Figure 3 Lightweight 4over6 experimental result

The performance test for lwAFTR is taken on a normal PC. Due to limitations of the PC hardware, the overall throughput is limited to around 800 Mbps. However, it can still support more than one hundred million concurrent sessions.

1.3. Conclusions

From the experiment, we can have the following conclusions:

- Lightweight 4over6 has good scalability. As it is a lightweight solution which only maintains per-subscription state information, it can easily support a large amount of concurrent subscribers.

- Lightweight 4over6 can be deployed rapidly. There is no modification to existing addressing and routing system in our operational network. And it is simple to achieve traffic logging.

- Lightweight 4over6 can support a majority of current IPv4 applications.

Authors' Addresses

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

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

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

Email: yiu_lee@cable.comcast.com

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

Email: fibrib@gmail.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 9, 2014

C. Xie
Q. Sun
China Telecom
Y. Lee
Comcast
T. Tsou
Huawei Technologies (USA)
Y. Chen
Tsinghua University
July 8, 2013

Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Lightweight 4over6
draft-sun-softwire-lw4over6-dhcpv6-00

Abstract

Lightweight 4over6 [I-D.ietf-softwire-lw4over6] is an extension to DS-Lite which moves the Network Address Translation function from the DS-Lite AFTR to the B4. It can be deployed in a DS-Lite network to gradually reduce the load of Carrier Grade NAT in the AFTR. However, when DS-Lite and lw4over6 co-exist in the same network, B4 elemtns and lwB4 elements may want to signal the DHCPv6 server the type of AFTR (i.e. AFTR or lwAFTR) they request. In this memo, a new DHCPv6 option is proposed for lwB4 element to request the IPv6 address of its corresponding lwAFTR.

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 January 9, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Terminology 3
3. Application Scenario 3
4. The lwAFTR-Name DHCPv6 Option 6
5. IANA Considerations 7
6. Acknowledgements 7
7. References 7
7.1. Normative References 7
7.2. Informative References 8
Authors' Addresses 8

1. Introduction

Lightweight 4over6 (lw4o6) [I-D.ietf-softwire-lw4over6] defines a model for providing IPv4 access over an IPv6 network in which the Network Address Translation (NAT) function is performed by the Customer-Premises Equipment (CPE) instead of being centralized on a Carrier-Grade NAT (CGN). It removes the requirement for a Carrier Grade NAT function in the AFTR, and reduces the amount of centralized state in the AFTR.

The DHCPv4 over DHCPv6 Transport [I.D-ietf-dhc-dhcpv4-over-dhcpv6] and Dynamic Host Configuration Protocol (DHCP) Option for Port Set [I.D-sun-dhc-port-set-option] can be used for lwB4 to be provisoned with the public IPv4 address and port set. To discover the lwAFTR's FQDN, [I-D.ietf-softwire-lw4over6] re-uses the DS-Lite DHCPv6 option defined in [RFC6334]. However, for operators who have deployed DS-Lite and will deploy lw4over6 in the same network using the same DHCPv6 option, the DHCPv6 server will not be able to distinguish between DS-Lite subscriber and lw4over6 subscriber.

In this memo, we define a new DHCPv6 option for lwB4 [I-D.ietf-softwire-lw4over6] to request the DHCPv6 server its corresponding lwAFTR. This new DHCPv6 option enables the DHCPv6 server to distinguish between DS-Lite subscriber and lw4over6 subscriber and offer the requested resources to the subscribers. This removes the requirement to pre-provision the subscriber type (i.e., DS-Lite or lw4over6) in the provision system. This option is particularly helpful in a scenario where operators offer both DS-Lite and lw4over6 in the same network.

2. Terminology

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].

Terminology defined in [I-D.ietf-softwire-lw4over6] is used extensively in this document.

3. Application Scenario

There are several possible deployment scenarios in which DS-Lite and lw4over6 co-exist in the same network.

In scenario 1, DS-Lite and lw4over6 are deployed in the same AFTR (depicted in Figure1).

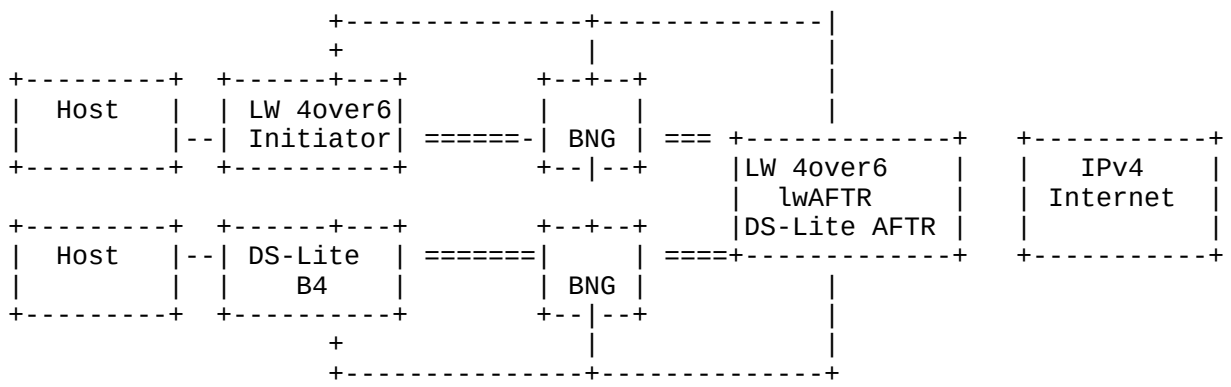


Figure 1: DS-Lite Coexistence scenario with Integrated AFTR

The AFTR needs to distinguish the traffic from two transition mechanisms. The first option is to distinguish using the client's source IPv4 address. Two transition mechanisms can share the same tunnel endpoint address. However, this requires the network element to examine every packet and may introduce significant overhead to the AFTR element. The second option is to use separate tunnel endpoint addresses for DS-Lite and lw4over6. This can be easily supported in the network element. The second option is more practical and recommended. This option requires the B4 element to discover the AFTR's FQDN and lwB4 element to discover lwAFTR's FQDN.

In scenario 2, DS-Lite AFTR and lw4over6 lwAFTR do not co-locate in the same network element (as depicted in Figure2) and are usually configured with different tunnel endpoint address. Similar to scenario 1 option 2, lwB4 also needs to discover a the lwAFTR's FQDN rather than the AFTR's FQDN.

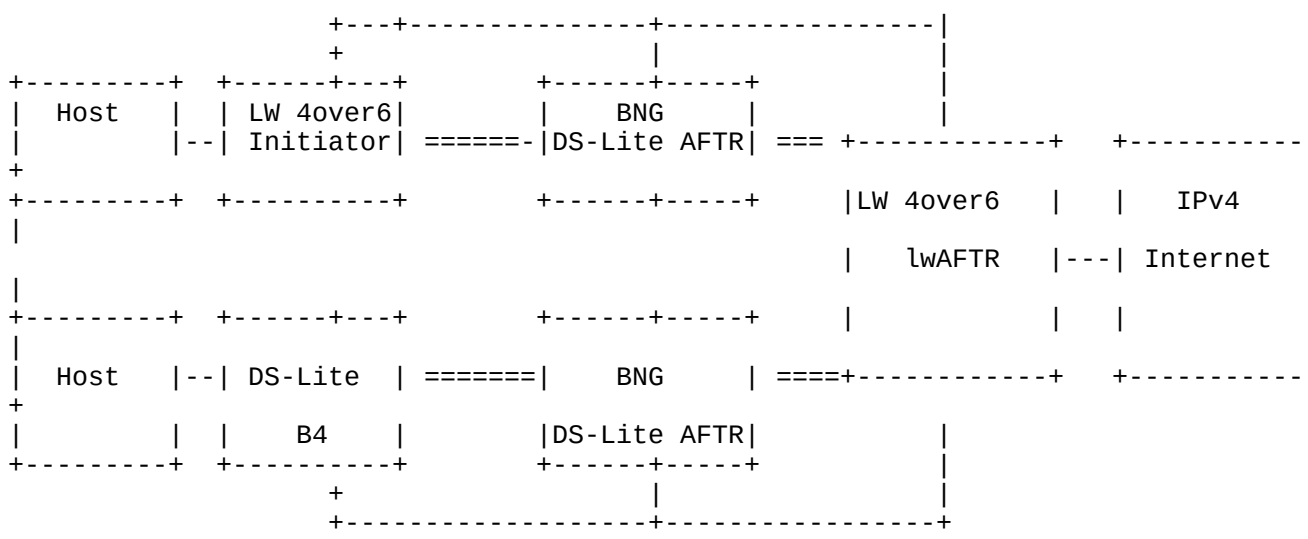


Figure 2: DS-Lite Coexistence scenario with Seperated AFTR

There are two possible solutions for an lw4over6 lwB4 to discover its the lwAFTR's IPv6 address.

1. Subscriber Type Pre-configuration

In this approach, the operator must pre-provision the subscriber type (e.g. Alice is lw4over6 subscriber and Bob is DS-Lite subscriber) in the provisioning system, this information will be used to instruct the DHCPv6 server to offer AFTR or lwAFTR to the subscriber in the DHCPv6 reply.

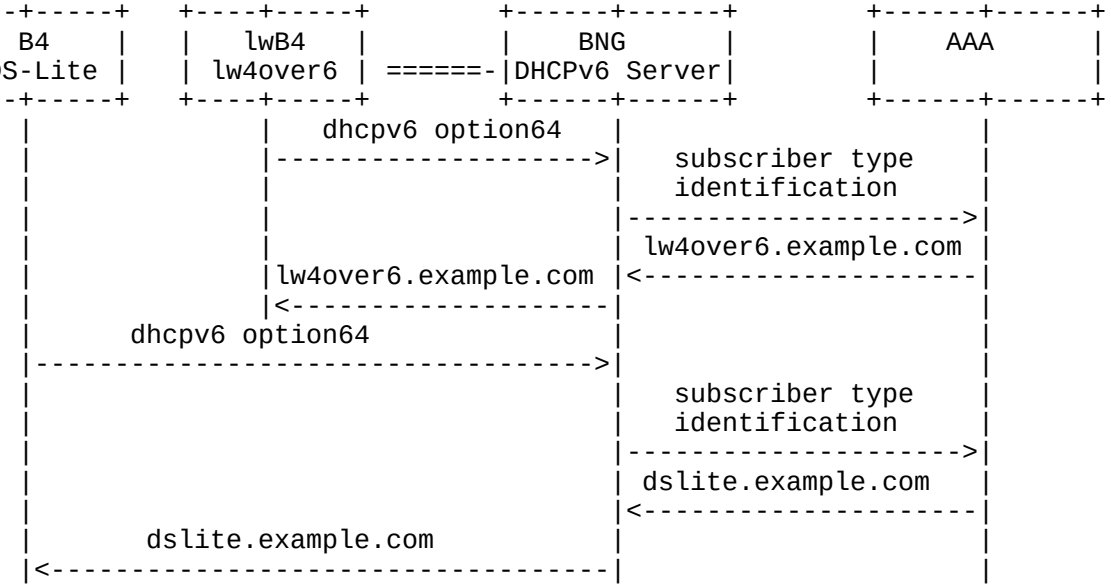


Figure 3: Workflow of Subscriber Type Pre-configuration

This approach requires operators to pre-provision static subscriber information in the provisioning system. This requires modification in the provisioning system to include this new subscriber information. Besides, when a subscriber migrates from DS-Lite to lw4over6, this will require update in the provisioning system.

2. lw4over6 DHCPv6 option

This approach is to use a new DHCPv6 option for lw4over6 lwB4.

The DHCPv6 server can identify a lw4over6 subscriber by the lw4over6 DHCPv6 request and offer lwAFTR's FQDN (depicted in the Figure4) to the lwB4 element.

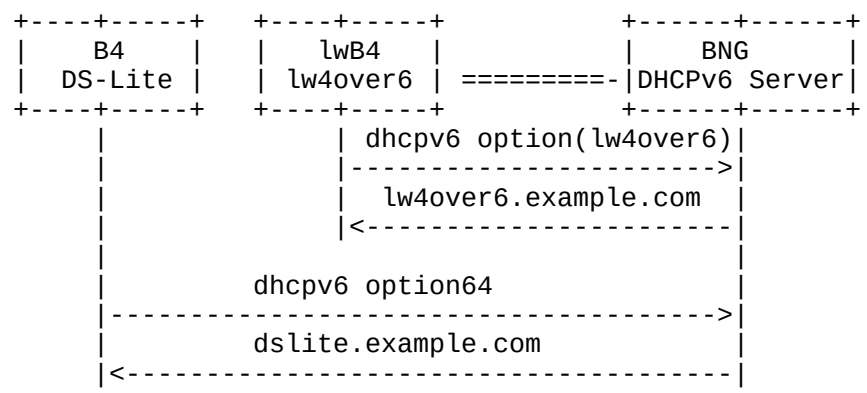


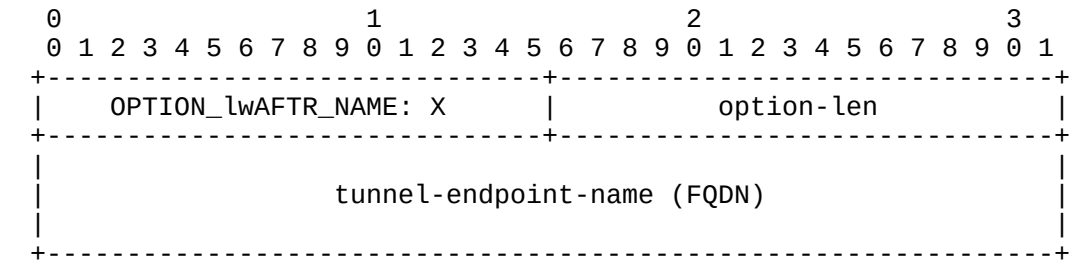
Figure 4: Workflow of lw4over6 DHCPv6 option

The new DHCPv6 option enables the DHCPv6 server to offer the lwAFTR's FQDN to lwB4, the provisioning system does not need to be upgraded to identify the subscriber's type. At migration, operators simply configure the B4 element to support NAT and this new DHCPv6 option, and this will be done.

Therefore, a new lw4over6 DHCPv6 option is recommended.

4. The lwAFTR-Name DHCPv6 Option

The format of lwAFTR-Name option is the same as DS-Lite AFTR-Name option with a new option-code. It is shown in Figure5.



OPTION_lwAFTR_NAME: TBD

option-len: Length of the tunnel-endpoint-name field in octets.

tunnel-endpoint-name: A fully qualified domain name of the lwAFTR tunnel endpoint.

Figure 5: Format of lwAFTR-Name DHCPv6 Option Format

The server behavior and the client behavior is exactly the same with DS-Lite AFTR-Name DHCPv6 Option ([RFC6334] section 4 and section5).

5. IANA Considerations

IANA is requested to allocate single DHCPv6 option code referencing this document, delineating OPTION_lwAFTR_NAME.

6. Acknowledgements

The authors would like to thank the following individuals who have participated in the drafting, review, and discussion of this memo: TO BE COMPLETED

7. References

7.1. Normative References

[I-D.ietf-pcp-port-set]
Sun, 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-00 (work in progress), March 2013.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 2013.

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

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

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

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

7.2. Informative References

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

Authors' Addresses

Chongfeng Xie
China Telecom
P.R.China

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

Qiong Sun
China Telecom
P.R.China

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

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

Email: yiu_lee@cable.comcast.com

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

Phone: +1 408 330 4424
Email: Tina.Tsou.Zouting@huawei.com

Yuchi Chen
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: peng-wu@foxmail.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 5, 2014

C. Xie
Q. Sun
China Telecom
Q. Sun
Tsinghua University
C. Zhou
Huawei Technologies
T. Tsou
Huawei Technologies (USA)
Z. Liu
Tsinghua University
March 4, 2014

Radius Extension for Lightweight 4over6
draft-sun-softwire-lw4over6-radext-01

Abstract

lightweight 4over6(lw4over6) [I-D.ietf-softwire-lw4over6] is an extension to DS-Lite in which the amount of state maintained in lwAFTR has been reduced to per-subscriber-level. The lwB4 needs to be provisioned with the public IPv4 address and port set it is allowed to use. The DHCPv4 over DHCPv6 Transport [I.D-ietf-dhc-dhcpv4-over-dhcpv6] and Dynamic Host Configuration Protocol (DHCP) Option for Port Set [I.D-sun-dhc-port-set-option] can be used for lwB4 to provison with the public IPv4 address and port set.

However, in many networks, the configuration information may be stored in Authentication Authorization and Accounting (AAA) servers while user configuration is mainly from Broadband Network Gateway (BNG). This document defines a Remote Authentication Dial In User Service (RADIUS) attribute that carries lightweight 4over6 configuration information from AAA server to BNG.

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 September 5, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Terminology 3
3. Lightweight 4over6 configuration process with RADIUS 3
4. Attributes 6
4.1. lw4o6_binding Attribute 6
5. Table of attributes 8
6. Security Considerations 9
7. IANA Considerations 9
8. Acknowledgements 9
9. References 9
9.1. Normative References 9
9.2. Informative References 10
Authors' Addresses 10

1. Introduction

Lightweight 4over6 (lw4over6) [I-D.ietf-softwire-lw4over6] defines a model for providing IPv4 access over an IPv6 network in which the Network Address Translation (NAT) function is performed by the Customer-Premises Equipment (CPE) instead of being centralized on a Carrier-Grade NAT (CGN). Lightweight 4over6 features keeping per-subscriber binding state in the service provider's network. This per-subscriber binding state is assigned by the provisioning system and should be synchronized between lwAFTRs. In lw4over6, there are multiple mechanisms to provision an lwB4 with the binding state,

including [I.D-ietf-dhc-dhcpv4-over-dhcpv6], [I-D.ietf-softwire-map-dhcp] , or [I-D.ietf-pcp-port-set], etc.

In many networks, user configuration information may be managed by AAA (Authentication, Authorization, and Accounting) servers. Current AAA servers communicate using the Remote Authentication Dial In User Service (RADIUS) [RFC2865] protocol. In a fixed line broadband network, the Broadband Network Gateways (BNGs) act as the access gateway of users. For lw4over6 case, the BNGs are assumed to embed a DHCPv4-over-DHCPv6 server function which allows them to locally handle any DHCPv4-over-DHCPv6 requests issued by hosts. The operators may per-configure subscriber’s binding state in AAA server which then passes the information to a BNG and in turn populates the mapping of the subscribe.

This document defines a new RADIUS attribute that can be used in lightweight 4over6 to carry subscriber’s binding state.

2. Terminology

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].

Terminology defined in [I-D.ietf-softwire-lw4over6] is used extensively in this document.

3. Lightweight 4over6 configuration process with RADIUS

The below Figure 1 illustrates how the RADIUS protocol and DHCPv4 -over-DHCPv6 cooperate to provide lwB4 with the binding state.

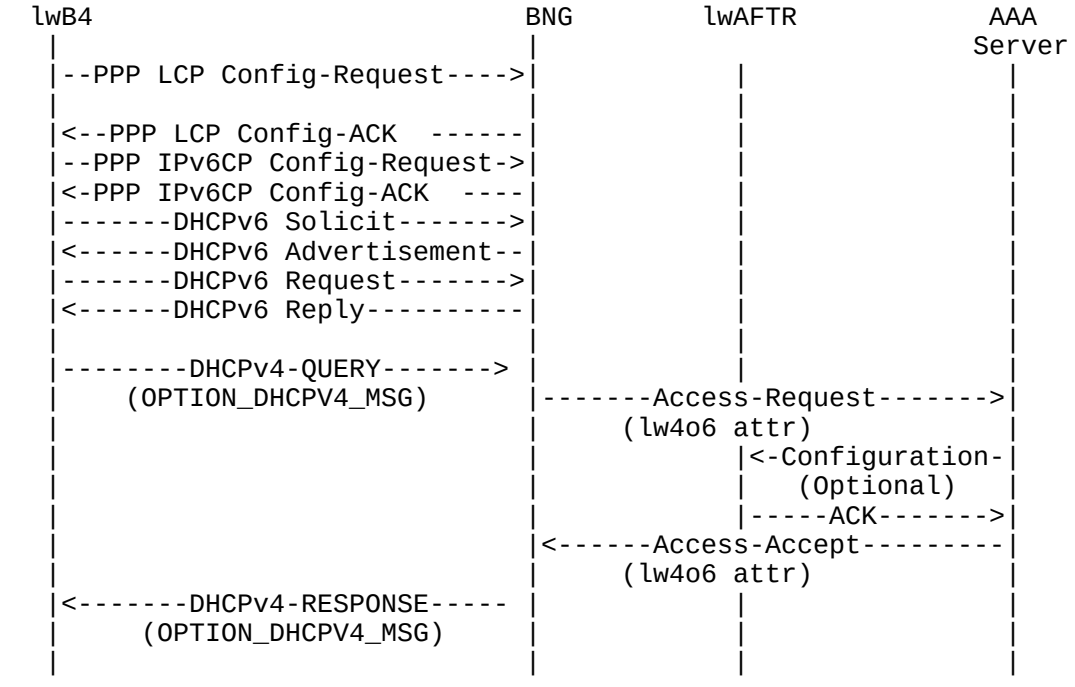


Figure 1: Lightweight 4over6 configuration process with RADIUS case 1

BNGs act as a client of RADIUS and as a Unified server. The lwB4 will firstly get the IPv6 address via DHCPv6 process. It then initiates a DHCPv4-QUERY message with OPTION_DHCPV4_MSG Option. Since the lwB4 has known the address of the Unified server in advance, it is recommanded to send the DHCPv4-QUERY message using unicast address. When receving the DHCPv4-QUERY from lwB4, the BNG SHOULD intercept the subscriber's IPv6 address and stored locally. Then, the BNG SHOULD initiate a RADIUS Access-Request message, in which the User-Name attribute (1) SHOULD be filled by the lwB4 MAC address, to the RADIUS server,the User-password attribute (2) SHOULD be filled by the shared lw4over6 password that has been preconfigured on the DHCPv6 server to get lw4over6 attribute. The IPv6 address in lw4o6 attribute should be filled by the subscriber's IPv6 address. The AAA server will then determine the IPv4 address and Port Set for the subscriber.

The subscriber's binding state should be syncronized between AAA server and lwAFTR. If the bindings are pre-configured statically in both AAA server and lwAFTR, the AAA server does not need to configure lwAFTR anymore. Otherwise, if the bindings are locally creately in

AAA server on-demand, it should inform the lwAFTR with the subscriber's binding state using [I-D.zhou-dime-4over6-provisioning] or COA requests.

Figure 2 illustrates how the RADIUS protocol and DHCPv6 cooperate to provide lwB4 and lwAFTR with tunnel configuration information.

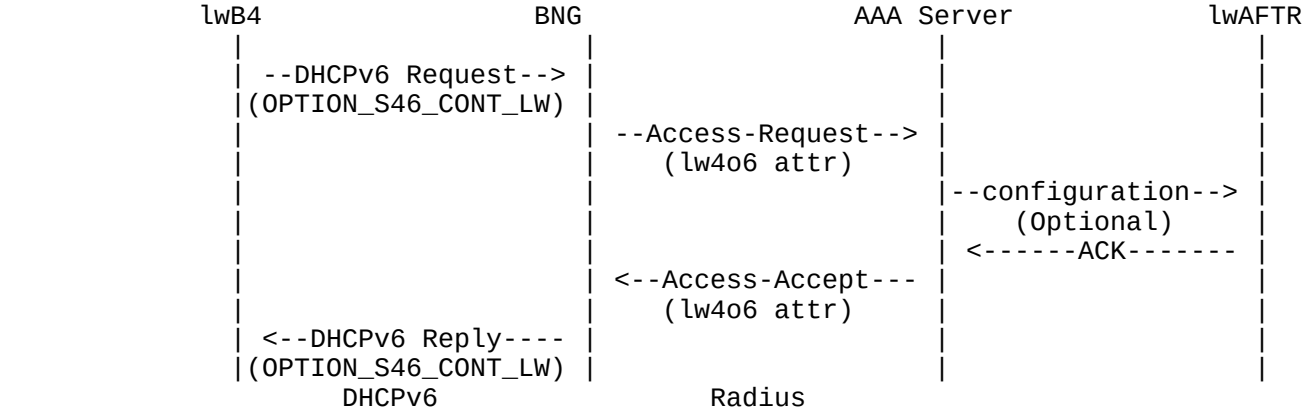


Figure 2: Lightweight 4over6 configuration process with RADIUS case 2

BNGs act as a RADIUS client and as a DHCPv6 server. Before the tunnel establishes, lwB4 MAY initiate a DHCPv6 Solicit message that includes an Option Request option[RFC3315] with OPTION_S46_CONT_LW option defined in [I-D.ietf-softwire-map-dhcp]. When BNG receives the SOLICIT, it SHOULD initiates radius Access-Request message, in which the User-Name attribute (1) SHOULD be filled by the lwB4 MAC address, to the RADIUS server,the User-password attribute (2) SHOULD be filled by the shared lw4over6 password that has been preconfigured on the DHCPv6 server to get lw4over6 attribute.

If the authentication request is approved by the AAA server, AAA server will determine the IPv6 address, IPv4 address and Port Set for the subscriber. The subscriber's binding state should be syncronized between AAA server and lwAFTR. If the bindings are pre-configured statically in both AAA server and lwAFTR, the AAA server does not need to configure lwAFTR anymore. Otherwise, if the bindings are locally creately in AAA server on-demand, it should inform the lwAFTR as mentioned above.

Similarly, BNGs can act as a RADIUS client and as a PCP server in case an lwB4 runs a PCP client (as depicted in Figure 3).

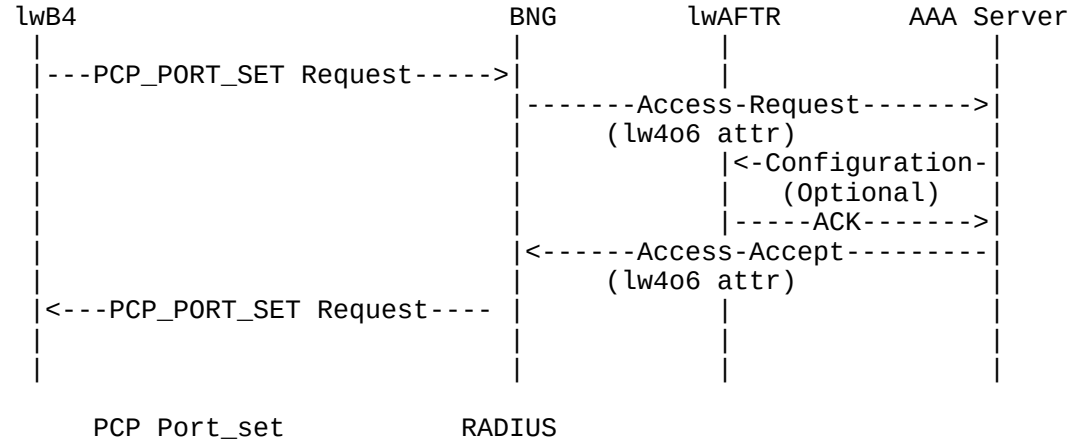


Figure 3: Lightweight 4over6 configuration process with RADIUS case 3

In the above-mentioned scenarios, Message-Authenticator (type 80) [RFC2865] SHOULD be used to protect both Access-Request and Access-Accept messages.

After receiving the lw4over6-binding attribute in the initial Access-Accept, the BNG SHOULD store the received lw4over6 configuration parameters locally. When the lw4over6 CE sends a DHCP or PCP Request message to request an extension of the lifetime for the assigned address, the BNG does not have to initiate a new Access-Request towards the AAA server to request the lw4o6 binding state. The BNG could retrieve the previously stored lw4o6 configuration parameters and use them in its reply. The BNG will then inform the AAA server with updated lifetime.

If the BNG does not receive the lw4over6-binding attribute in the Access-Accept or if the BNG receives an Access-Reject, the tunnel cannot be established.

4. Attributes

This section defines the lw4o6_binding attribute that is used in both above-mentioned scenarios. The attribute design follows [RFC6158] and refers to [RFC6929].

4.1. lw4o6_binding Attribute

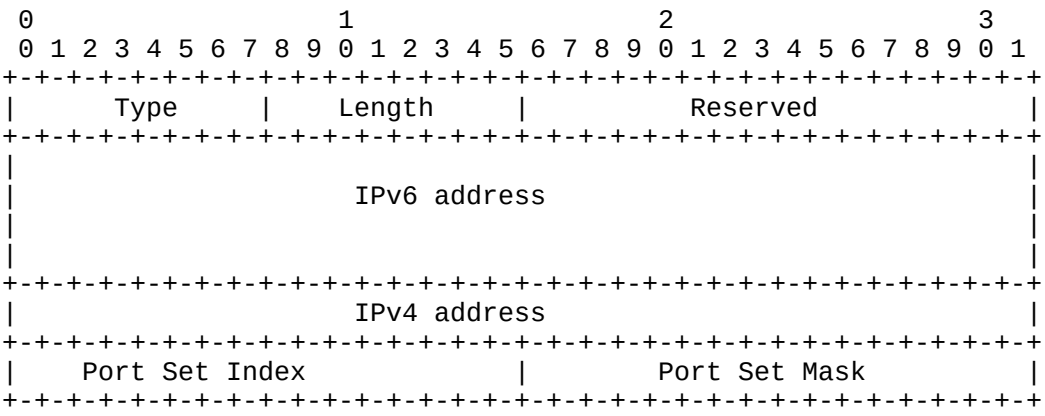
The lw4o6_binding RADIUS attribute contains the subscriber’s binding information including IPv6 address, IPv4 address and the port-set. The BNG SHALL use the binding entry returned in the RADIUS lw4o6_binding attribute to populate the requests.

If the BNG includes the lw4o6_binding attribute, but the AAA server does not recognize it, this attribute MUST be ignored by the AAA server.

If the BNG does not receive the lw4o6_binding attribute in the Access-Accept message and there is the unified server in BNG is not configured to allocate the port-set by itself, the unified SHOULD not response and the tunnel can not be established.

When the Access-Request message is triggered by a DHCP Rebind message, if the binding attribute received in the Access-Accept message is different from the currently used one for that session, the BNG MUST force the lwB4 to re-establish the tunnel using the new binding information received in the Access-Accept message.

The lw4o6_binding Attribute is structured as follows:



Type

TBD

Length

28

Port Set Index:
Port Set Index identifies a set of ports assigned to a device. The first k bits on the left of the 2-octet field is the Port Set Index value, with the rest of the field right padding zeros.

Port Set Mask:
Port Set Mask indicates the position of the bits used to build the mask. The first k bits on the left is padding ones while the remained (16-k) bits of the 2-octet field on the right is padding zeros.

IPv4 address
The translated IPv4 address for a subscriber.

IPv6 address
The IPv6 address for a subscriber.

Figure 4: Lightweight 4over6 Attribute

5. Table of attributes

The following table provides a guide to which attributes may be found in which kinds of packets, and in what quantity.

Request	Accept	Reject	Challenge	Accounting Request	#	Attribute
0-1	0-1	0	0	0-1	TBD1	lw4o6-binding
0-1	0-1	0	0	0-1	1	User-Name
0-1	0	0	0	0	2	User-Password
0-1	0-1	0	0	0-1	6	Service-Type
0-1	0-1	0-1	0-1	0-1	80	Message-Authenticator

The following table defines the meaning of the above table entries.

0	This attribute MUST NOT be present in packet.
0+	Zero or more instances of this attribute MAY be present in packet.
0-1	Zero or one instance of this attribute MAY be present in packet.
1	Exactly one instance of this attribute MUST be present in packet.

Figure 5: Lightweight 4over6 Attribute Table

6. Security Considerations

TO BE COMPLETED

7. IANA Considerations

This document has no IANA actions.

8. Acknowledgements

The authors would like to thank the following individuals who have participated in the drafting, review, and discussion of this memo: TO BE COMPLETED

9. References

9.1. Normative References

[I-D.ietf-pcp-port-set]
Sun, 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-00 (work in progress), March 2013.

[I-D.ietf-softwire-lw4over6]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-ietf-softwire-lw4over6-00 (work in progress), April 2013.

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

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

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

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

9.2. Informative References

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

Authors' Addresses

Chongfeng Xie
China Telecom
P.R.China

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

Qiong Sun
China Telecom
P.R.China

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

Qi Sun
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: sunqibupt@gmail.com

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: cathy.zhou@huawei.com

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

Phone: +1 408 330 4424
Email: Tina.Tsou.Zouting@huawei.com

ZiLong Liu
Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
Email: liuzilong8266@126.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: August 17, 2014

T. Tsou
Huawei Technologies (USA)
B. Li
C. Zhou
Huawei Technologies
J. Schoenwaelder
Jacobs University Bremen
R. Penno
Cisco Systems, Inc.
M. Boucadair
France Telecom

February 13, 2014

DS-Lite Failure Detection and Failover
draft-tsou-softwire-bfd-ds-lite-06

Abstract

In DS-Lite, the tunnel is stateless, not associated with any state information, and the CGN function at the AFTR is stateful. Currently, there is no failure detection and failover mechanism for both stateless tunnel and stateful CGN function, which makes it difficult to manage and diagnose if there is a problem. This draft analyzes the applicability of some of the possible solutions.

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 17, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Terminology 3
3. Failover Mechanisms 3
3.1. Anycast Approach 4
3.2. VRRP Approach 4
4. Solutions 4
4.1. Bidirectional Forwarding Detection (BFD) 4
4.1.1. DS-Lite Scenario 5
4.1.2. Parameters for BFD 5
4.1.3. Elements of Procedure 6
4.1.4. BFD for NAT failure detection 6
4.1.5. Implementation Considerations 6
4.2. Port Control Protocol (PCP) 7
4.3. ICMP Echo Request / Echo Reply (PING) 7
4.4. Comparison of Different Solutions 8
5. State Synchronization and Session Re-establishment 8
6. IANA Considerations 9
7. Security Considerations 9
8. Acknowledgements 9
9. References 9
9.1. Normative References 9
9.2. Informative References 10
Authors' Addresses 10

1. Introduction

In DS-Lite [RFC6333], the IPv4-in-IPv6 DS-Lite tunnel is stateless, no status information about the tunnel is available, and no keep-alive mechanism is available. It is difficult to know whether the tunnel is up or down; and if there is a link problem, the Basic Bridging BroadBand (B4) element can not automatically switch to another Address Family Transition Router (AFTR) so as to continue the network service automatically, without the involvement of operators. Besides, In DS-Lite [RFC6333], the CGN function at the AFTR is stateful and there is no mechanism to detect whether the NAT44 CGN is functioning in the AFTR. These will create problems for network operation and maintenance.

Possible solutions for failure detection include the usage of Bidirectional Forwarding Detection (BFD), the Port Control Protocol (PCP), and ICMP Echo Request / Echo Reply (PING). The properties of these solutions are discussed in this document and guidelines are provided how to implement failure detection and automatic failover.

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 RFC 2119 [RFC2119].

2. Terminology

AFTR: Address Family Transition Router.

B4: Basic Bridging BroadBand.

BBF: BroadBand Forum.

BFD: Bidirectional Forwarding Detection.

CPE: Customer Premise Equipment (i.e., the DS-Lite B4).

FQDN Fully Qualified Domain Name.

PCP Port Control Protocol.

3. Failover Mechanisms

The FQDN of the AFTR is sent to the B4 element via a DHCP option, as defined in [RFC6334]. Multiple IP addresses can be configured for the FQDN of an AFTR on the DNS server. If a B4 element detects a failure on the link to the AFTR, the B4 element MUST terminate the

current DS-Lite tunnel, choose another AFTR address in the list, and create a tunnel to the new AFTR. If necessary, the B4 element SHOULD re-configure the connectivity test tool accordingly and restart the test procedures.

3.1. Anycast Approach

Anycasts may also be used for failover. But there is an ICMP-error-message problem with anycast, that is, when a packet is sent from the AFTR to a B4 element, if one of the routers along the path generates an ICMP error message, e.g., Packet Too Big (PTB), then the error message may not be sent back to the source AFTR but to another AFTR.

There's also a problem with anycast for stateful CGN/AFTR. If there is an asymmetric path though the CGNs, then return path traffic will be dropped as there is no corresponding state table entry in the AFTR.

3.2. VRRP Approach

For active/passive HA in NAT gateways, it's quite common to have a single virtual address offered by VRRP (or a proprietary equivalent) that the upstream routers will use as their next hop. In the event that the master CGN fails, the standby takes over the virtual L3 address. If a VRRP based virtual address is used as the tunnel endpoint, then the clients wouldn't need to be aware of the failover.

4. Solutions

4.1. Bidirectional Forwarding Detection (BFD)

Bidirectional Forwarding Detection [RFC5880] (BFD) is a mechanism intended to detect faults in a bidirectional path. It is usually used in conjunction with applications like OSPF, IS-IS, for fast fault recovery and fast re-route [RFC5882]. BFD is being made mandatory for keep-alive for subscriber sessions, including DS-Lite, by the BroadBand Forum (BBF) [WT-146].

BFD can be used in DS-Lite, by creating a BFD session between the B4 element and the AFTR to provide tunnel status information. If a fault is detected, the B4 element can try to create a DS-Lite tunnel with another AFTR and terminate the existing one, so as to continue network service. BFD could also be used to detect the CGN state at the AFTR, but the detection should be based on per-user.

[I-D.vinokour-bfd-dhcp] proposes using a DHCP option to distribute BFD parameters to B4 elements. But in case of DS-Lite, some of the

key BFD parameters are already available (e.g., peer IP address), and other parameters can be negotiated by BFD signaling or statically configured, so that no extra DHCP option(s) need to be defined.

4.1.1. DS-Lite Scenario

In DS-Lite [RFC6333], the BFD packet SHOULD be sent through an IPv4-in-IPv6 tunnel, as shown in Figure 1. The IPv4 addresses of the B4 element and the AFTR SHOULD be the endpoints of a BFD session.

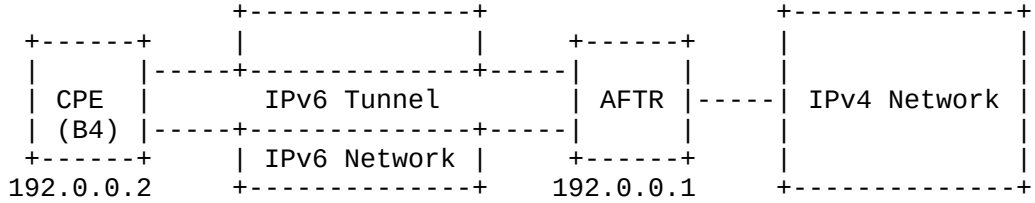


Figure 1: DS-Lite Scenario

4.1.2. Parameters for BFD

In order to set up a BFD session, the following parameters are needed, as shown in Section 4.1 of [RFC5880]:

- Peer IP address
- My Discriminator
- Your Discriminator
- Desired Min TX Interval
- Required Min RX Interval
- Required Min Echo RX Interval

B4's WAN-side IPv4 address is the well-known address 192.0.0.2, and the AFTR's well-known IPv4 address is 192.0.0.1, as defined in section 5.7 of [RFC6333]. The B4 element needs to create an IPv6 tunnel to an AFTR so as to get network connectivity to the AFTR, and send IPv4 BFD packets through the tunnel to manage it.

The other parameters listed above can be negotiated by BFD signaling, and initial values can be configured on B4 elements and AFTRs.

4.1.3. Elements of Procedure

When a B4 element gets online, it will be assigned an IPv6 prefix or address, and also the FQDN of the AFTR, as defined in [RFC6334]. The B4 element will create an IPv6 tunnel to the AFTR with which the B4 element can initiate a BFD session to the AFTR. BFD packets will be sent through the DS-Lite tunnel. As defined in section 4 of [RFC5881], BFD control packets MUST be sent in UDP packets with destination port 3784, and BFD echo packets MUST be sent in UDP packets with destination port 3785.

When sending out the first BFD packet, the B4 element can generate a unique local discriminator, and set the remote discriminator to zero. When the AFTR receives the first BFD packet from a B4 element, the AFTR will also generate a corresponding local discriminator, and put it in the response packet to the B4 element. This will finish the discriminator negotiation in the B4 to AFTR direction, without any manual configuration.

When an AFTR receives the first packet from a B4 element, the AFTR will get the IPv6 address and discriminator of the B4 element, so that the AFTR can initiate the BFD session in the other direction and a similar discriminator negotiation can be carried out.

4.1.4. BFD for NAT failure detection

B4 creates PCP mapping. BFD at AFTR uses an external public interface (or another external mapping) to send a BFD packet to the public PCP mapping created by B4. In this case, the AFTR BFD packet will have a public source IP of interface, which will go through the NAT, therefore exercising the NAT function. B4 will reply to the AFTR external interface.

4.1.5. Implementation Considerations

BFD is usually used for quick fault detection, at a very small time scale, e.g. milliseconds. But in DS-Lite, it may not be necessary to detect faults in such a short time. On the other hand, an AFTR may need to support tens of thousands of B4 elements, which means an AFTR will need to support the same number of BFD sessions. In order to meet performance requirements on an AFTR, it may be necessary to extend the time period between BFD packet transmissions to a longer time, e.g., 10s or 30s.

Compared to other solutions, BFD has a simple and fixed packet format, which is easy to implement by logic devices (e.g., ASIC, FPGA). Complicated protocols are usually processed by software which is relatively slow. An AFTR may need to support 10000-20000 users,

and if the protocol is handled by software, it will bring extra load to the AFTR.

4.2. Port Control Protocol (PCP)

[RFC6887]PCP is a NAT traversal tool. It can also be used for network connectivity test if PCP is supported in the network. A common use case of PCP is to create a pinhole so that external users can visit the servers located behind a NAT. The lifetime of the pinhole mapping is usually long, e.g., hours, and the lifetime will be refreshed periodically by the client before it is expired. For the purpose of network connectivity tests, a B4 element can create a mapping in the CGN via PCP, with a short life time, e.g., 10s of seconds, and keep on refreshing the mapping before it expires. If any refresh requests fail, the B4 element knows that something is wrong with the link or the PCP server or the CGN.

In order to detect the network connectivity of the DS-Lite tunnel, the encapsulation mode MUST be used for PCP: PCP packets are sent through the DS-Lite tunnel.

PCP can detect the failure of more components of the DS-Lite system. Besides failures of the link and the routing, it also covers NAT functions.

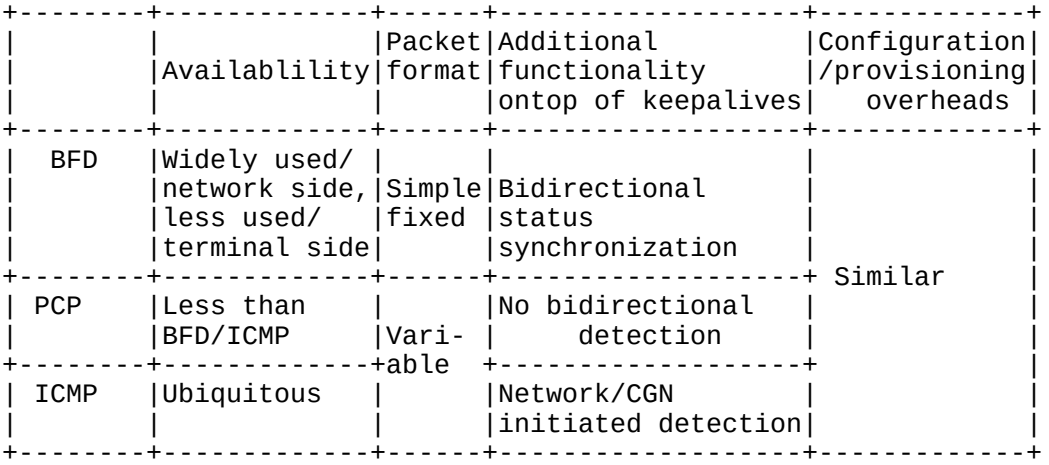
4.3. ICMP Echo Request / Echo Reply (PING)

PING is commonly implemented using the Echo Request and Echo Response messages of the Internet Control Message Protocol (ICMP) [RFC0792] [RFC4443]. In case of DS-Lite, a B4 element can send Echo Request packets to the AFTR periodically. If the B4 element does not receive Echo Response packets for a certain number (e.g., 3) of Echo Request packets, then the B4 element decides that a fault has been detected.

In order to test the connectivity of DS-Lite tunnel, Echo Request packets MUST be sent using ICMPv4, rather than ICMPv6.

Since ICMP is an integral part of any IP implementation, the usage of PING to detect tunnel failures does not require any special implementation efforts on the B4 elements. However, on AFTRs that process ICMP messages in software rather than in hardware, the usage of PING might lead to scalability issues.

4.4. Comparison of Different Solutions



	Availablility	Packet format	Additional functionality ontop of keepalives	Configuration /provisioning overheads
BFD	Widely used/ network side, less used/ terminal side	Simple fixed	Bidirectional status synchronization	Similar
PCP	Less than BFD/ICMP	Vari-able	No bidirectional detection	
ICMP	Ubiquitous		Network/CGN initiated detection	

Figure 2: Comparison of different solutions

Figure 2 gives a direct comparison among different solutions. Compared to other solutions, BFD has a simple and fixed packet format, which is easy to implement by logic devices (e.g., ASIC, FPGA). Complicated protocols are usually processed by software which is relatively slow. ICMP is widely used than PCP/BFD, while BFD is more widely used in the router and CGN side than in the terminal side. However, from the aspect of failure detection, BFD has explicit capability of bidirectional status synchronization to guarantee the consistency of the failure status of both sides. ICMP could actively initiate status detection from the network side or CGN side, while PCP could not. PCP has no capability of bidirectional detection. Considering the configuration/provisioning overheads, since there is normally TR-069 server at the network management side. So it is similar for each approach.

From the above comparison, BFD is selected as the failure detection approach in this document.

5. State Synchronization and Session Re-establishment

There should be a state sync mechanism between active AFTR and backup AFTR, to synchronize the state of each user between the two AFTRs. This mechanism is to guarantee that the traffic returning to the B4 is from the backup AFTR, if the service is shifted to that AFTR. The BFD link for both active AFTR and backup AFTR should be set up in the

initial state. When the active AFTR is detected in failure, the service will be shifted to the backup AFTR. If the backup AFTR is detected in failure, it will notify the network management server to fix the failure.

In the hot-standby case, the master AFTR and the backup AFTR will synchronize and backup the session. So there is no need to re-establish the TCP session in the event of an AFTR failure. But in the cold-standby case, if there is an active TCP session through the CGN function of an AFTR, and this AFTR fails, then the TCP session will need to be re-established by the client because only the capability is reserved but the session is not backup.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

In the DS-Lite [RFC6333] application, the B4 element may not be directly connected to the AFTR; there may be other routers between them. In such a deployment, there are potential spoofing problems, as described in [RFC5883]. Hence cryptographic authentication SHOULD be used with BFD as described in [RFC5880] if security is concerned.

8. Acknowledgements

The authors would like to thank Ian Farrer for his valuable comments.

9. References

9.1. Normative References

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

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

[RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection

(BFD)", RFC 5880, June 2010.

[RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, June 2010.

[RFC5882] Katz, D. and D. Ward, "Generic Application of Bidirectional Forwarding Detection (BFD)", RFC 5882, June 2010.

[RFC5883] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for Multihop Paths", RFC 5883, June 2010.

[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.

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

[WT-146] Kavanagh, A., Klamm, F., Boucadair, W., and R. Dec, "WT-146 Subscriber Sessions (work in progress)", Apr 2012.

9.2. Informative References

[I-D.vinokour-bfd-dhcp]
Vinokour, V., "Configuring BFD with DHCP and Other Musings", May 2008.

Authors' Addresses

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

Phone: +1 408 330 4424
Email: tina.tsou.zouting@huawei.com

Brandon Li
Huawei Technologies
M6, No. 156, Beiqing Road, Haidian District
Beijing 100094
China

Phone:
Email: brandon.lijian@huawei.com

Cathy Zhou
Huawei Technologies
China

Phone:
Email: cathy.zhou@huawei.com

Juergen Schoenwaelder
Jacobs University Bremen
Campus Ring 1
Bremen 28759
Germany

Phone:
Email: j.schoenwaelder@jacobs-university.de

Reinaldo Penno
Cisco Systems, Inc.
170 West Tasman Drivee
San Jose, California 95134
USA

Phone:
Email: repenno@cisco.com

Mohamed Boucadair
France Telecom
Rennes,35000
France

Phone:
Email: mohamed.boucadair@orange.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 16, 2015

X. Xu
Huawei Technologies
N. Sheth
Juniper Networks
R. Asati
Cisco Systems

February 12, 2015

BGP Tunnel Encapsulation Attribute for UDP
draft-xu-softwire-encaps-udp-02

Abstract

This document specifies a new Border Gateway Protocol (BGP) Tunnel Type of User Datagram Protocol (UDP) tunnels.

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 16, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Requirements Language 2
2. Terminology 2
3. BGP Tunnel Type Code for UDP 2
4. Security Considerations 3
5. IANA Considerations 3
6. Contributors 3
7. Acknowledgements 3
8. References 4
8.1. Normative References 4
8.2. Informative References 4
Authors' Addresses 4

1. Introduction

[RFC5512] specifies a method by which Border Gateway Protocol (BGP) speakers can signal tunnel encapsulation information to each other and accordingly it defines support for Generic Routing Encapsulation (GRE) [RFC2784], Layer Two Tunneling Protocol - Version 3 (L2TPv3) [RFC3931] and IP in IP [RFC2003] tunnel types. This document builds on [RFC5512] and defines support for the User Datagram Protocol (UDP) tunnel type which is applicable to the MPLS-in-UDP encapsulation [I-D.ietf-mpls-in-udp].

1.1. Requirements Language

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 RFC 2119 [RFC2119].

2. Terminology

This memo makes use of the terms defined in [RFC5512].

3. BGP Tunnel Type Code for UDP

To use either the Encapsulation Subsequent Address Family Identifier (SAFI) or the BGP Encapsulation Extended Community defined in [RFC5512] to signal the UDP tunnel type information across BGP speakers, a new Tunnel Type code (TBD) indicating the UDP tunnel type needs to be assigned by IANA. This document does not specify any UDP tunnel specific sub-TLV. Furthermore, the BGP Encapsulation Network Layer Reachability Information (NLRI) Format is not modified by this document.

4. Security Considerations

The security considerations mentioned in [RFC5512] is applicable to this new BGP Tunnel Type code for UDP tunnels as well. No new security risk is introduced by this new Tunnel Type code for UDP tunnels.

5. IANA Considerations

A new BGP Tunnel Type code indicating the UDP tunnel type needs to be assigned by IANA.

6. Contributors

Note that contributors are listed in alphabetical order according to their last names.

Yongbing Fan

China Telecom

Email: fanyb@gsta.com

Yiu Lee

Comcast

Email: Yiu_Lee@Cable.Comcast.com

Zhenbin Li

Huawei Technologies

Email: lizhenbin@huawei.com

7. Acknowledgements

Thanks to

8. References

8.1. Normative References

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

[RFC5512] Mohapatra, P. and E. Rosen, "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", RFC 5512, April 2009.

8.2. Informative References

[I-D.ietf-mpls-in-udp] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", draft-ietf-mpls-in-udp-11 (work in progress), January 2015.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[RFC3931] Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.

Authors' Addresses

Xiaohu Xu
Huawei Technologies
No.156 Beiqing Rd
Beijing 100095
CHINA

Phone: +86-10-60610041
Email: xuxiaohu@huawei.com

Nischal Sheth
Juniper Networks
1194 N. Mathilda Ave
Sunnyvale, CA 94089
USA

Email: nsheth@juniper.net

Rajiv Asati
Cisco Systems
7200 Kit Creek Road
Research Triangle Park,, NC 27709
USA

Email: rajiva@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 6, 2015

X. Xu
Huawei Technologies
R. Asati
Cisco Systems
L. Yong
Huawei USA
Y. Lee
Comcast
Y. Fan
China Telecom
I. Beijnum
Institute IMDEA Networks
December 3, 2014

Encapsulating IP in UDP
draft-xu-softwire-ip-in-udp-03

Abstract

Existing Softwire encapsulation technologies are not adequate for efficient load balancing of Softwire service traffic across IP networks. This document specifies additional Softwire encapsulation technology, referred to as IP-in-User Datagram Protocol (UDP), which can facilitate the load balancing of Softwire service traffic across IP networks.

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 June 6, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Conventions 3
2. Terminology 3
3. Encapsulation in UDP 3
4. Processing Procedures 5
5. Congestion Considerations 5
6. Security Considerations 6
7. IANA Considerations 7
8. Acknowledgements 7
9. References 8
9.1. Normative References 8
9.2. Informative References 8
Authors' Addresses 9

1. Introduction

To fully utilize the bandwidth available in IP networks and/or facilitate recovery from a link or node failure, load balancing of traffic over Equal Cost Multi-Path (ECMP) and/or Link Aggregation Group (LAG) across IP networks is widely used. [RFC5640] describes a method for improving the load balancing efficiency in a network carrying Softwire Mesh service [RFC5565] over Layer Two Tunneling Protocol - Version 3 (L2TPv3) [RFC3931] and Generic Routing Encapsulation (GRE) [RFC2784] encapsulations. However, this method requires core routers to perform hash calculation on the "load-balancing" field contained in tunnel encapsulation headers (i.e., the Session ID field in L2TPv3 headers or the Key field in GRE headers), which is not widely supported by existing core routers.

Most existing routers in IP networks are already capable of distributing IP traffic "microflows" [RFC2474] over ECMP paths and/or

LAG based on the hash of the five-tuple of User Datagram Protocol (UDP) [RFC0768] and Transmission Control Protocol (TCP) packets (i.e., source IP address, destination IP address, source port, destination port, and protocol). By encapsulating the Softwire service traffic into an UDP tunnel and using the source port of the UDP header as an entropy field, the existing load-balancing capability as mentioned above can be leveraged to provide fine-grained load-balancing of Softwire service traffic traffic over IP networks. This is similar to why LISP [RFC6830] uses UDP encapsulation. Therefore, this specification defines an IP-in-UDP encapsulation method for Software service (including both mesh and hub-spoke modes).

IPv6 flow label has been proposed as an entropy field for load balancing in IPv6 network environment [RFC6438]. However, as stated in [RFC6936], the end-to-end use of flow labels for load balancing is a long-term solution and therefore the use of load balancing using the transport header fields would continue until any widespread deployment is finally achieved. As such, IP-in-UDP encapsulation would still have a practical application value in the IPv6 networks during this transition timeframe.

Similarly, the IP-in-UDP encapsulation format defined in this document by itself cannot ensure the integrity and privacy of data packets being transported through the IP-in-UDP tunnels and cannot enable the tunnel decapsulators to authenticate the tunnel encapsulator. Therefore, in the case where any of the above security issues is concerned, the IP-in-UDP SHOULD be secured with IPsec [RFC4301] or DTLS [RFC6347]. For more details, please see Section 6 of Security Considerations.

1.1. Conventions

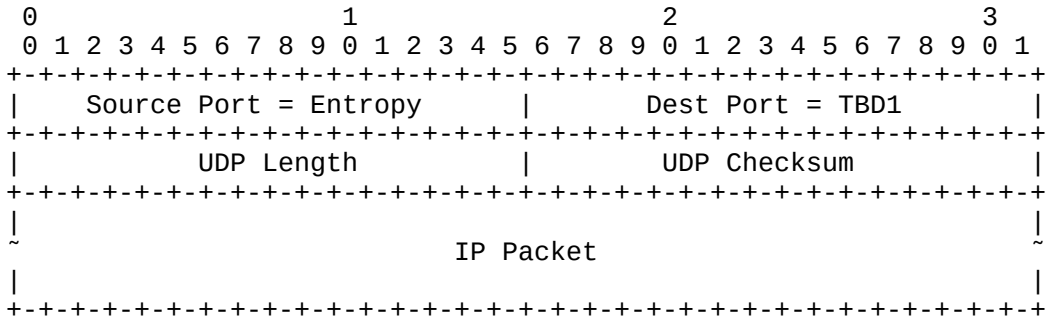
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Terminology

This memo makes use of the terms defined in [RFC5565].

3. Encapsulation in UDP

IP-in-UDP encapsulation format is shown as follows:



Source Port of UDP

This field contains a 16-bit entropy value that is generated by the encapsulator to uniquely identify a flow. What constitutes a flow is locally determined by the encapsulator and therefore is outside the scope of this document. What algorithm is actually used by the encapsulator to generate an entropy value is outside the scope of this document.

In case the tunnel does not need entropy, this field of all packets belonging to a given flow SHOULD be set to a randomly selected constant value so as to avoid packet reordering.

To ensure that the source port number is always in the range 49152 to 65535 (Note that those ports less than 49152 are reserved by IANA to identify specific applications/protocols) which may be required in some cases, instead of calculating a 16-bit hash, the encapsulator SHOULD calculate a 14-bit hash and use those 14 bits as the least significant bits of the source port field while the most significant two bits SHOULD be set to binary 11. That still conveys 14 bits of entropy information which would be enough as well in practice.

Destination Port of UDP

This field is set to a value (TBD1) allocated by IANA to indicate that the UDP tunnel payload is an IP packet. As for whether the encapsulated IP packet is IPv4 or IPv6, it would be determined according to the Version field in the IP header of the encapsulated IP packet.

UDP Length

The usage of this field is in accordance with the current UDP specification [RFC0768].

UDP Checksum

For IPv4 UDP encapsulation, this field is RECOMMENDED to be set to zero because the IPv4 header includes a checksum and use of the UDP checksum is optional with IPv4. For IPv6 UDP encapsulation, the IPv6 header does not include a checksum, so this field MUST contain a UDP checksum that MUST be used as specified in [RFC0768] and [RFC2460] unless one of the exceptions that allows use of UDP zero-checksum mode (as specified in [RFC6935]) applies.

IP Packet

This field contains one IP packet.

4. Processing Procedures

This IP-in-UDP encapsulation causes E-IP[RFC5565] packets to be forwarded across an I-IP [RFC5565] transit core via "UDP tunnels". While performing IP-in-UDP encapsulation, an ingress AFBR (e.g. PE router) would generate an entropy value and encode it in the Source Port field of the UDP header. The Destination Port field is set to a value (TBD1) allocated by IANA to indicate that the UDP tunnel payload is an IP packet. Transit routers, upon receiving these UDP encapsulated IP packets, could balance these packets based on the hash of the five-tuple of UDP packets. Egress AFBRs receiving these UDP encapsulated IP packets MUST decapsulate these packets by removing the UDP header and then forward them accordingly (assuming that the Destination Port was set to the reserved value pertaining to IP).

Similar to all other Softwire tunneling technologies, IP-in-UDP encapsualtion introduces overheads and reduces the effective Maximum Transmision Unit (MTU) size. IP-in-UDP encapsulation may also impact Time-to-Live (TTL) or Hop Count (HC) and Differentiated Services (DSCP). Hence, IP-in-UDP MUST follow the corresponding procedures defined in [RFC2003]. If an ingress AFBR performs fragmentation on an E-IP packet before encapsulating, it MUST use the same source UDP port for all fragmented packets so as to ensures these fragmented packets are always forwarded on the same path.

5. Congestion Considerations

Section 3.1.3 of [RFC5405] discussed the congestion implications of UDP tunnels. As discussed in [RFC5405], because other flows can share the path with one or more UDP tunnels, congestion control [RFC2914] needs to be considered. As specified in [RFC5405]:

> "IP-based traffic is generally assumed to be congestion-controlled, i.e., it is assumed that the transport protocols generating IP-based traffic at the sender already employ mechanisms that are sufficient to address congestion on the path. Consequently, a tunnel carrying IP-based traffic should already interact appropriately with other traffic sharing the path, and specific congestion control mechanisms for the tunnel are not necessary".

Since IP-in-UDP is only used to carry IP traffic which is generally assumed to be congestion controlled, it generally does not need additional congestion control mechanisms.

6. Security Considerations

The security problems faced with the IP-in-UDP tunnel are exactly the same as those faced with IP-in-IP [RFC2003] and IP-in-GRE tunnels [RFC2784]. In other words, the IP-in-UDP tunnel as defined in this document by itself cannot ensure the integrity and privacy of data packets being transported through the IP-in-UDP tunnel and cannot enable the tunnel decapsulator to authenticate the tunnel encapsulator. In the case where any of the above security issues is concerned, the IP-in-UDP tunnel SHOULD be secured with IPsec or DTLS. IPsec was designed as a network security mechanism and therefore it resides at the network layer. As such, if the tunnel is secured with IPsec, the UDP header would not be visible to intermediate routers anymore in either IPsec tunnel or transport mode. As a result, the meaning of adopting the IP-in-UDP tunnel as an alternative to the IP-in-GRE or IP-in-IP tunnel is lost. By comparison, DTLS is better suited for application security and can better preserve network and transport layer protocol information. Specifically, if DTLS is used, the destination port of the UDP header will be filled with a value (TBD2) indicating IP with DTLS and the source port can still be used as an entropy field for load-sharing purposes.

If the tunnel is not secured with IPsec or DTLS, some other method should be used to ensure that packets are decapsulated and forwarded by the tunnel tail only if those packets were encapsulated by the tunnel head. If the tunnel lies entirely within a single administrative domain, address filtering at the boundaries can be used to ensure that no packet with the IP source address of a tunnel endpoint or with the IP destination address of a tunnel endpoint can enter the domain from outside. However, when the tunnel head and the tunnel tail are not in the same administrative domain, this may become difficult, and filtering based on the destination address can even become impossible if the packets must traverse the public Internet. Sometimes only source address filtering (but not destination address filtering) is done at the boundaries of an

administrative domain. If this is the case, the filtering does not provide effective protection at all unless the decapsulator of an IP-in-UDP validates the IP source address of the packet.

7. IANA Considerations

One UDP destination port number indicating IP needs to be allocated by IANA:

Service Name: IP-in-UDP

Transport Protocol(s): UDP

Assignee: IESG <iesg@ietf.org>

Contact: IETF Chair <chair@ietf.org>.

Description: Encapsulate IP packets in UDP tunnels.

Reference: This document.

Port Number: TBD1 -- To be assigned by IANA.

One UDP destination port number indicating IP with DTLS needs to be allocated by IANA:

Service Name: IP-in-UDP-with-DTLS

Transport Protocol(s): UDP

Assignee: IESG <iesg@ietf.org>

Contact: IETF Chair <chair@ietf.org>.

Description: Encapsulate IP packets in UDP tunnels with DTLS.

Reference: This document.

Port Number: TBD2 -- To be assigned by IANA.

8. Acknowledgements

Thanks to Vivek Kumar, Carlos Pignataro and Mark Townsley for their valuable comments on the initial idea of this document. Thanks to Andrew G. Malis for his valuable comments on this document.

9. References

9.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

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

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC5405] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", BCP 145, RFC 5405, November 2008.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[RFC6935] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", RFC 6935, April 2013.

[RFC6936] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, April 2013.

9.2. Informative References

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2914] Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, September 2000.

[RFC3931] Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.

[RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.

[RFC5640] Filsfils, C., Mohapatra, P., and C. Pignataro, "Load-Balancing for Mesh Softwires", RFC 5640, August 2009.

[RFC6438] Carpenter, B. and S. Amante, "Using the IPv6 Flow Label for Equal Cost Multipath Routing and Link Aggregation in Tunnels", RFC 6438, November 2011.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

Authors' Addresses

Xiaohu Xu
Huawei Technologies
No.156 Beiqing Rd
Beijing 100095
CHINA

Phone: +86-10-60610041
Email: xuxiaohu@huawei.com

Rajiv Asati
Cisco Systems
7200 Kit Creek Road
Research Triangle Park,, NC 27709
USA

Email: rajiva@cisco.com

Lucy Yong
Huawei USA
5340 Legacy Dr
Plano, TX 75025
USA

Email: Lucy.yong@huawei.com

Yiu Lee
Comcast
One Comcast Center
Philadelphia, PA
USA

Phone: Email: Yiu_Lee@Cable.Comcast.com
Email: cpignata@cisco.com

Yongbing Fan
China Telecom
Guangzhou
CHINA

Email: fanyb@gsta.com

Iljitsch van Beijnum
Institute IMDEA Networks
Avda. del Mar Mediterraneo, 22
Leganes,, Madrid 28918
Spain

Email: iljitsch@muada.com