

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: March 25, 2012

T. Murakami
IP Infusion
O. Troan
cisco
S. Matsushima
SoftBank
September 22, 2011

IPv4 Residual Deployment on IPv6 infrastructure - protocol specification
[draft-murakami-software-4rd-01](#)

Abstract

This document specifies an automatic tunneling mechanism for providing IPv4 connectivity service to end users over a service provider's IPv6 network. Key aspects include stateless operation, sharing of IPv4 addresses, and an algorithmic mapping between IPv4 addresses and IPv6 tunnel endpoints.

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 25, 2012.

Copyright Notice

Copyright (c) 2011 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.	Terminology	4
4.	4rd Configuration	5
4.1.	Customer Edge Configuration	6
5.	Algorithmic mapping	6
5.1.	Mapping Rules	6
5.1.1.	From a CE IPv6 Prefix to a CE 4rd Prefix	6
5.1.2.	From a CE 4rd Prefix to a Port-set ID	7
5.1.3.	From a Port-Set ID to a Port Set	8
5.1.4.	From an IPv4 Address or IPv4 Address + Port to a CE IPv6 Address	10
6.	Encapsulation and Fragmentation Consideration	11
7.	BR and CE behaviors	12
8.	NAT considerations	14
9.	ICMP	15
10.	Security Considerations	15
11.	IANA Consideration	16
12.	Acknowledgements	16
13.	References	17
13.1.	Normative References	17
13.2.	Informative References	17
	Authors' Addresses	18

1. Introduction

4rd is a protocol mechanism to deploy IPv4 to sites via a service provider's (SP's) IPv6 network. Similar to Dual-Stack Lite [[I-D.ietf-softwire-dual-stack-lite](#)], 4rd is designed to allow IPv4 traffic to be delivered over an IPv6 network without the direct provisioning of IPv4 addresses. 4rd can provide an IPv4 prefix, an IPv4 address or a shared IPv4 address. Like 6rd [[RFC5969](#)], 4rd is operated in a fully stateless manner within the SP network. The motivation for a stateless alternative to Dual-Stack Lite is described in "Motivations for Stateless IPv4 over IPv6 Migration Solutions" [[I-D.operators-softwire-stateless-4v6-motivation](#)].

4rd relies on IPv6 and is designed to deliver production-quality dual-stack service while allowing IPv4 to be phased out within the SP network. The phasing out of IPv4 within the SP network is independent of whether the end user disables IPv4 service or not. Further, "Greenfield" IPv6-only networks may use 4rd in order to deliver IPv4 to sites via the IPv6 network in a way that does not require protocol translation between IPv4 and IPv6.

4rd utilizes an algorithmic mapping between the IPv6 and IPv4 addresses that are assigned for use within the SP network. This mapping provides automatic determination of IPv6 tunnel endpoints from IPv4 destination addresses, allowing the stateless operation of 4rd. 4rd views the IPv6 network as a link layer for IPv4 and supports an automatic tunneling abstraction similar to the Non-Broadcast Multiple Access (NBMA) [[RFC2491](#)] model.

The 4rd algorithmic mapping is also used to automatically provision IPv4 addresses and allocating a set of non-overlapping ports for each 4rd CE. The "SP-facing" (i.e., "WAN") side of the 4rd CE, operate as native IPv6 interface with no need for IPv4 operation or support. On the "end-user-facing" (i.e., "LAN") side of a CE, IPv6 and IPv4 are implemented as for any native dual-stack service delivered by the SP.

A 4rd domain consists of 4rd Customer Edge (CE) routers and one or more 4rd Border Relays (BRs). IPv4 packets encapsulated by 4rd follow the IPv6 routing topology within the SP network between CEs and among CEs and BRs. CE to CE traffic is direct, while BRs are traversed only for IPv4 packets that are destined to or are arriving from outside a given 4rd domain. As 4rd is stateless, BRs may be reached using anycast for failover and resiliency.

4rd does not require any stateful NATP [RFC3022] functions at the BRs or elsewhere within the SP network. Instead, 4rd allows for sharing of IPv4 addresses among multiple sites by automatically allocating a set of non-overlapping ports for each CE as part of the stateless

mapping function. It is expected that the CE will, in turn, perform local IPv4 Network Address and Port Translation (NAPT) [RFC3022] functions for the site as is commonly performed today, except avoiding ports outside of the allocated port set. Although 4rd is designed primarily to support IPv4 deployment to a customer site (such as a residential home network) by an SP, it can equally be applied to an individual host acting as a CE router.

[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 [RFC 2119](#) [RFC2119].

[3.](#) Terminology

4rd domain (Domain): A set of 4rd CEs and BRs connected to the same virtual 4rd link. A service provider may deploy 4rd with a single 4rd domain, or may utilize multiple 4rd domains. Each domain requires a separate 4rd prefix.

4rd Border Relay (BR): A 4rd-enabled router managed by the service provider at the edge of a 4rd domain. A Border Relay router has at least one of each of the following: an IPv6-enabled interface, a 4rd virtual interface acting as an endpoint for the

4rd IPv4 in IPv6 tunnel, and an IPv4 interface connected to the native IPv4 network. A 4rd BR may also be referred to simply as a "BR" within the context of 4rd.

4rd Customer Edge (CE): A device functioning as a Customer Edge router in a 4rd deployment. In a residential broadband deployment, this type of device is sometimes referred to as a "Residential Gateway" (RG) or "Customer Premises Equipment" (CPE). A typical 4rd CE serving a residential site has one WAN side interface, one or more LAN side interfaces, and a 4rd virtual interface. A 4rd CE may also be referred to simply as a "CE" within the context of 4rd.

CE IPv6 prefix:	The IPv6 prefix assigned to a CE by other means than 4rd itself, and used by 4rd to derive a CE 4rd prefix.
CE IPv6 address:	In the context of 4rd, the IPv6 address used to reach the 4rd function of a CE from other CE's and from BR's. The IID of this address differs from that of host interface address that start with the CE IPv6 prefix.
CE 4rd prefix:	The 4rd prefix of the CE. It is derived from the CE IPv6 prefix by a mapping rule according to Section 5.1 . Depending on its length, it is an IPv4 prefix, an IPv4 address, or a shared IPv4 address followed by a Port-set ID (Section 5.1.2).
Port-set ID:	In a CE 4rd prefix longer than 32 bits, bits that follow the first 32. It algorithmically identifies a set of ports exclusively assigned to the CE. As specified in Section 5.1.2 , the set can comprise up to 4

disjoint port ranges.

Domain IPv6 prefix: An IPv6 prefix assigned by an ISP to a 4rd domain.

Domain IPv4 prefix: A 4rd prefix assigned by an ISP to the 4rd domain.

IPv4 Embedded Address (EA) bits: The IPv4 EA-bits in the IPv6 address identify an IPv4 prefix, IPv4 address or part of IPv4 address and port set.

Shared IPv4 address: An IPv4 address that is shared among multiple nodes. Each node has a separate part of the transport layer port space.

[4.](#) 4rd Configuration

The IPv4 prefix, IPv4 address or shared IPv4 address for use at a customer site is created by extracting the IPv4 embedded address (EA-bits) from the IPv6 prefix delegated to the site. Combined with the 4rd IPv4 prefix, the IPv4 prefix, IPv4 address or shared IPv4 address is automatically created by the CE for the customer site when IPv6 service is obtained.

For a given 4rd domain, the BR and CE MUST be configured with a set of mapping rules and BR IPv6 addresses. The configured values for these elements MUST be consistent for all CEs and BRs within a given 4rd domain.

A mapping rule consist of the following elements: a Domain IPv6 prefix and prefix length, a Domain 4rd prefix and prefix length, CE IPv6 Prefix length, and a Domain IPv6 suffix and length. See section ([Section 5.1](#)) for a detailed description of mapping rules.

[4.1.](#) Customer Edge Configuration

The 4rd configuration elements are set to values that are the same across all CEs within a 4rd domain. The values may be configured in a variety of manners, including provisioning methods such as the

Broadband Forum's "TR-69" [TR069] Residential Gateway management interface, an XML-based object retrieved after IPv6 connectivity is established, a DNS record, an SMIPv2 MIB [[RFC2578](#)], or manual configuration by an administrator. A companion document [[I-D.mrugalski-dhc-dhcpv6-4rd](#)] describes how to configure the necessary parameters via IPv6 DHCP. A CE that allows IPv6 configuration by IPv6 DHCP SHOULD implement this option. Other configuration and management methods may use the format described by this option for consistency and convenience of implementation on CEs that support multiple configuration methods.

The only remaining provisioning information the CE requires in order to calculate the 4rd address and enable IPv6 connectivity is an IPv6 prefix for the CE. This CE IPv6 prefix is configured as part of obtaining IPv6 Internet access (i.e., configured via SLAAC, DHCPv6, DHCPv6 PD, or otherwise).

A single 4rd CE MAY be connected to more than one 4rd domain. Each domain a given CE operates within would require its own set of 4rd configuration elements and would generate its own 4rd address.

[5.](#) Algorithmic mapping

[5.1.](#) Mapping Rules

[5.1.1.](#) From a CE IPv6 Prefix to a CE 4rd Prefix

A 4rd mapping rule establishes a 1:1 mapping between CE IPv6 prefixes and CE 4rd prefixes.

```

<----- CE IPv6 prefix (max 128) ----->
+-----+-----+-----+-----+-----+
|      Domain IPv6 prefix      |      EA-bits      |
+-----+-----+-----+-----+-----+
<-- Domain IPv6 Prefix length -->:<-- EA-bits length -->:
                                   :
                                   :      ||      :
                                   :      \ /      :

```

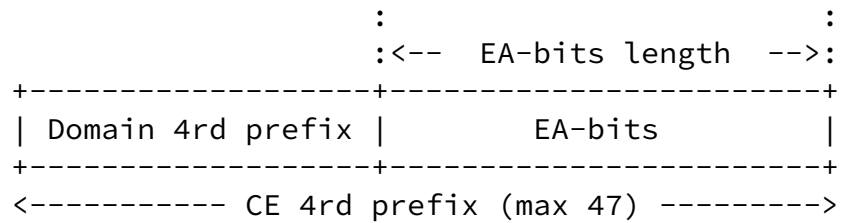


Figure 1: From a CE IPv6 Prefix to a CE 4rd Prefix

A CE derives its CE 4rd prefix from the CE IPv6 prefix, using parameters of the applicable mapping rule. If the domain has several mapping rules, the rule that applies is that whose Domain IPv6 prefix has the longest match with the CE IPv6 prefix. As shown in Figure 1, the CE 4rd prefix is created by concatenating the Domain 4rd prefix with the IPv4 EA-bits, where the IPv4 EA-bits is the remainder of the CE IPv6 prefix after the Domain IPv6 prefix (the length of the Domain IPv6 prefix is defined by the mapping rule).

[5.1.2.](#) From a CE 4rd Prefix to a Port-set ID

Depending on its length, a CE 4rd prefix is either an IPv4 prefix, a full IPv4 address, or a shared IPv4 address followed by a Port-set ID (Figure 2). If it includes a port set ID, this ID specifies which ports are assigned to the the CE for its exclusive use ([Section 5.1.3](#)).

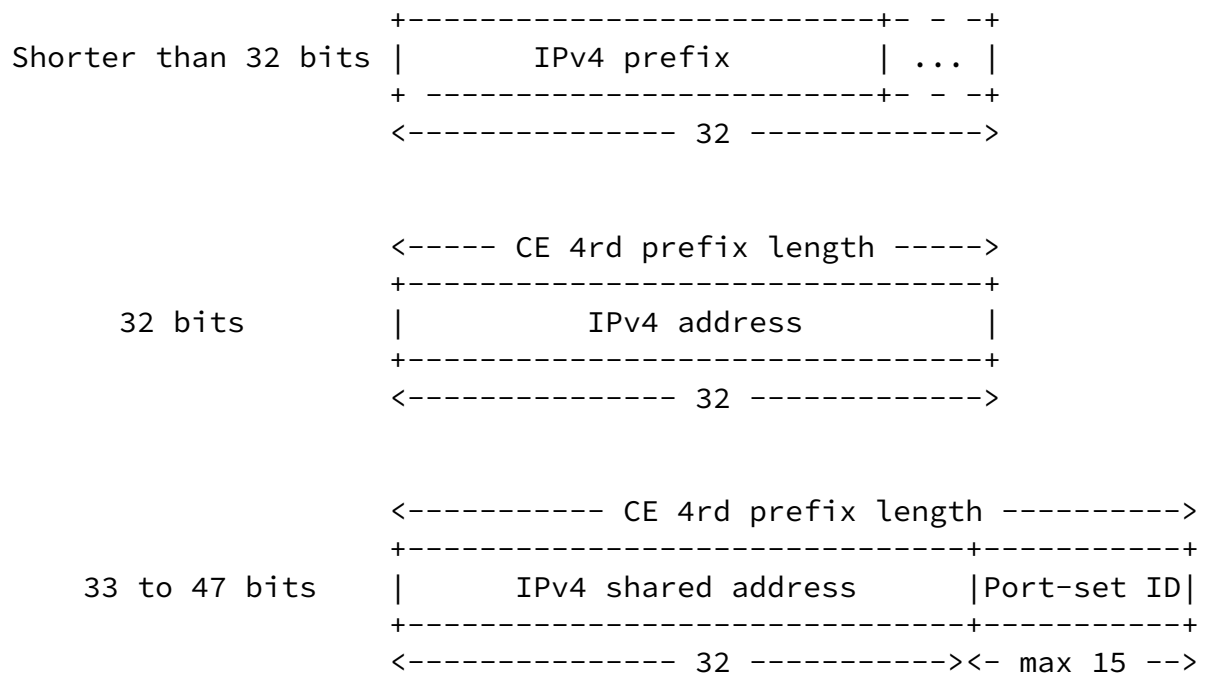


Figure 2: Variants of CE 4rd prefixes

5.1.3. From a Port-Set ID to a Port Set

The value of a Port-set ID specifies which ports can be used by a transport layer protocol (UDP, TCP, SCTP etc). Design constraint of the algorithm are the following:

Fairness with respect to special-value ports: No port-set must contain any well-known ports [IANA reference].

Fairness with respect to the number of ports For a Port-set-ID's having the same length, all sets must have the same number of ports.

Exhaustiveness For any Port-set-ID length, the aggregate of port sets assigned for all values must include all ordinary-value ports.

If the Port-set ID has 1 to 12 bits, the set comprises 4 port ranges. As shown in Figure 3, each port range is defined by its port prefix, made of a range-specific "head" followed by the Port-set ID. Head values are in binary 1, 01, 001, and 0001. They are chosen to exclude ports 0-4095 and only them.

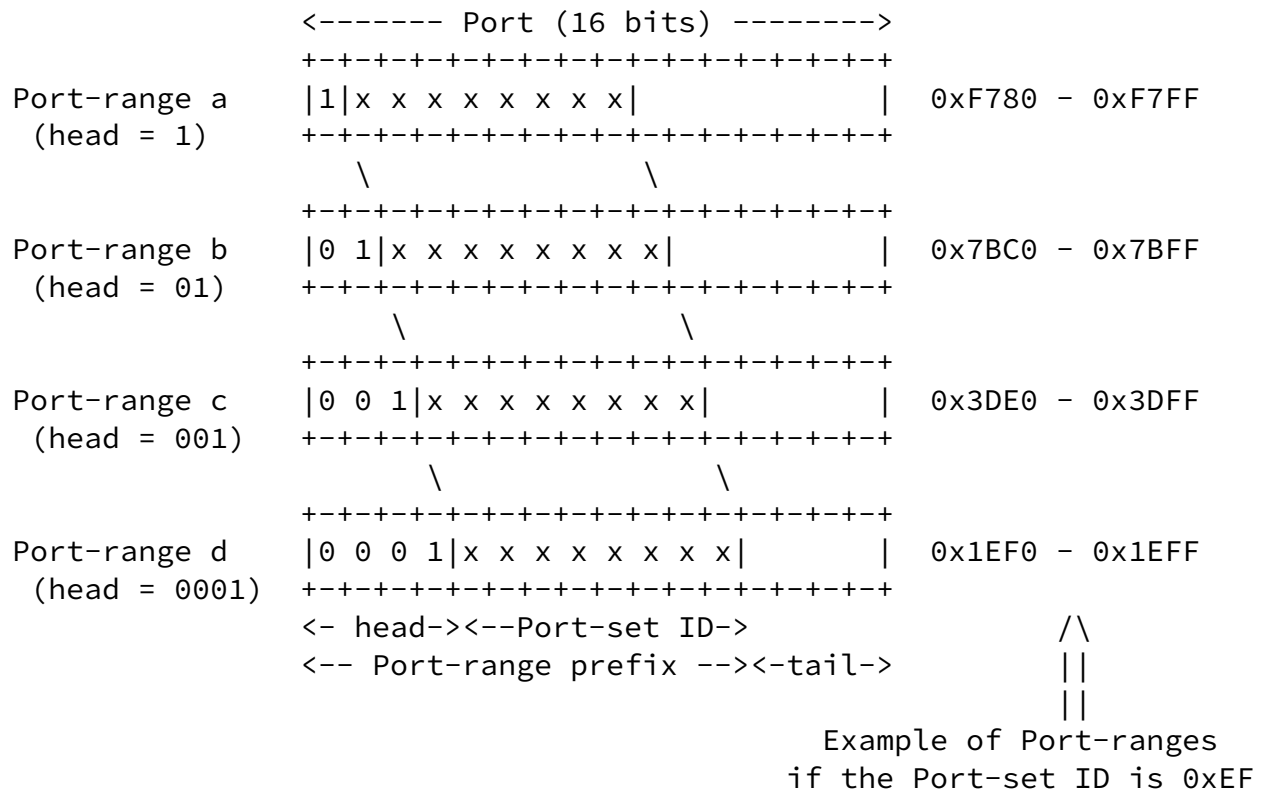


Figure 3: From Port-set ID to Port ranges

In the Port-set ID has 13 bits, only the 3 port ranges are assigned, having heads 1, 01, and 001. If it has 14 bits, only the 2 port ranges having heads 1 and 01 are assigned. If it has 15 bits, only the port range having head 1 is assigned. (In these three cases, the smallest port range has only one element).

5.1.4. From an IPv4 Address or IPv4 Address + Port to a CE IPv6 Address

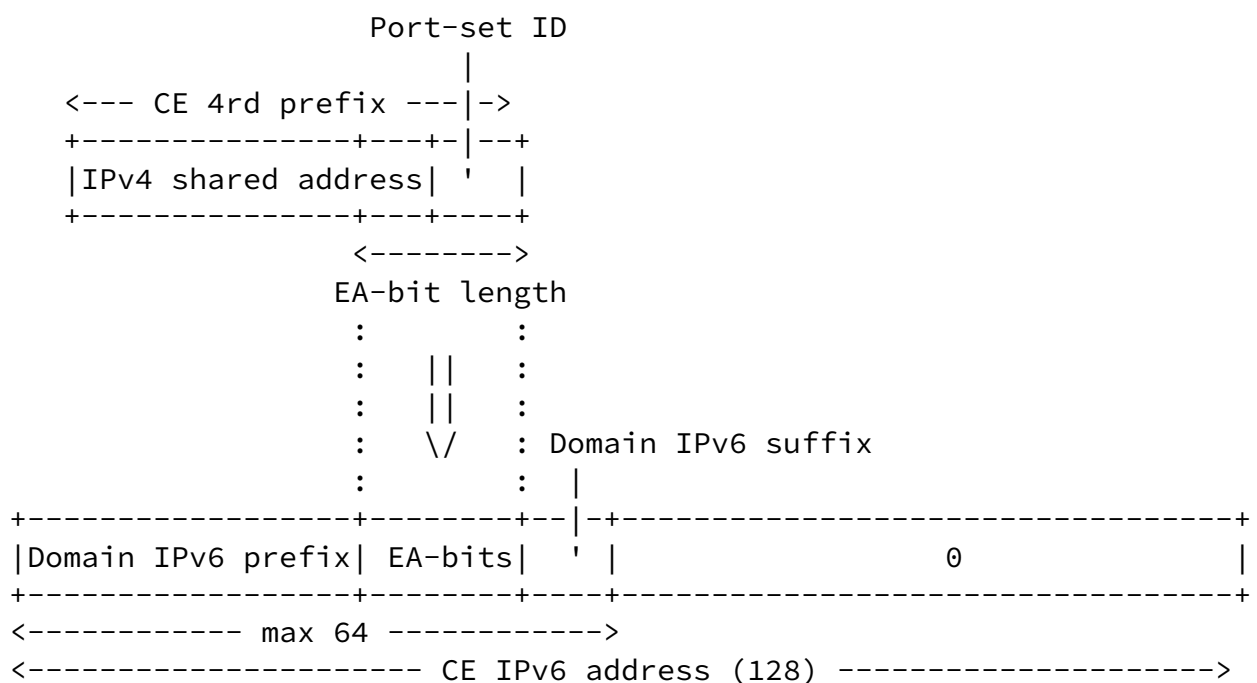


Figure 4: From 4rd Prefix to IPv6 address (shared IPv4 address case)

In order to find whether a CE IPv6 address can be derived from an IPv4 address, or an IPv4 address + a port, a mapping rule has to be found that matches the IPv4 information:

- o If a mapping rule has a length L of CE IPv4 prefixes which does not exceed 32 bits, there is a match if the IPv4 address starts with the Domain 4rd prefix. The CE 4rd prefix is then the first L bits of the IPv4 address.
- o If a mapping rule has a length L of CE IPv4 prefixes which exceeds 32 bits, the match can only be found with the IPv4 address and the port. For this, the port is examined to determine which port-range head it starts with: 1, 01,001, or 0001. The N bits that follow this head are taken as Port-set ID, where N is the length of Port set ID of the mapping rule. The CE 4rd prefix is then

made of the IPv4 address followed by the Port-set ID.

If a match has been found, the CE IPv6 prefix is then made of the Domain IPv6 prefix followed by bits of the CE 4rd prefix that follow the Domain 4rd prefix, followed by the Domain IPv6 prefix of the mapping rule if there is one, and followed by 0's up to 128 bits to make a complete IPv6 address ([\[RFC4291\]](#)). Figure 4 illustrates this process in the case of a shared IPv4 address.

[6.](#) Encapsulation and Fragmentation Consideration

Maximum transmission unit (MTU) and fragmentation issues for IPv4 in IPv6 tunneling are discussed in detail in [Section 7.2 of \[RFC2473\]](#). 4rd's scope is limited to a service provider network. IPv6 Path MTU discovery MAY be used to adjust the MTU of the tunnel as described in [Section 7.2 of \[RFC2473\]](#), or the 4rd Tunnel MTU might be explicitly configured.

The use of an anycast source address could lead to any ICMP error message generated on the path being sent to a different BR.

Therefore, using dynamic tunnel MTU [Section 7.2 of \[RFC2473\]](#) is subject to IPv6 Path MTU blackholes.

Multiple BRs using the same anycast source address could send fragmented packets to the same 4rd CE at the same time. If the fragmented packets from different BRs happen to use the same fragment ID, incorrect reassembly might occur. For this reason, a BR using an anycast source address MUST NOT fragment the IPv6 encapsulated packet unless BR's having identical rules are required to use disjoint ranges of fragment ID.

If the MTU is well-managed such that the IPv6 MTU on the CE WAN side interface is set so that no fragmentation occurs within the boundary of the SP, then the 4rd Tunnel MTU should be set to the known IPv6 MTU minus the size of the encapsulating IPv6 header (40 bytes). For example, if the IPv6 MTU is known to be 1500 bytes, the 4rd Tunnel MTU might be set to 1460 bytes. Absent more specific information, the 4rd Tunnel MTU SHOULD default to 1280 bytes.

Alternatively, if BR's having identical rule are required to use disjoint ranges of fragment ID, a BR using an anycast source address

SHOULD fragment the IPv6 encapsulated packet correctly.

For 4rd domain traversal, IPv4 packets are encapsulated in IPv6 packets whose Next header is set to 4 (i.e. IPv4). If fragmentation of IPv6 packets is needed, it is performed according to [\[RFC2460\]](#). Absent more specific information, the path MTU of a 4rd Domain has to be set to 1280 [\[RFC2460\]](#).

In domains where IPv4 addresses are not shared, IPv6 destinations are derived from IPv4 addresses alone. Thus, each IPv4 packet can be encapsulated and decapsulated independently of each other. 4rd processing is completely stateless.

On the other hand, in domains where IPv4 addresses are shared, BR's and CE's can have to encapsulate IPv4 packets whose IPv6 destinations depend on destination ports. Precautions are needed, due to the fact

that the destination port of a fragmented datagram is available only in its first fragment. A sufficient precaution consists in reassembling each datagram received in multiple packets, and to treat it as though it would have been received in single packet. This function is such that 4rd is in this case stateful at the IP layer. (This is common with DS-lite and NAT64/DNS64 which, in addition, are stateful at the transport layer.) At Domain entrance, this ensures that all pieces of all received IPv4 datagrams go to the right IPv6 destinations.

Another peculiarity of shared IPv4 addresses is that, without precaution, a destination could simultaneously receive from different sources fragmented datagrams that have the same Datagram ID (the Identification field of [\[RFC0791\]](#)). This would disturb the reassembly process. To eliminate this risk, CE MUST rewrite the datagram ID to an unique value among CEs having same shared IPv4 address upon sending the packets over 4rd tunnel. This value SHOULD be generated locally within the port-range assigned to a given CE. Note that replacing a Datagram ID in an IPv4 header implies an update of its Header-checksum field, by adding to it the one's complement difference between the old and the new values.

[7.](#) BR and CE behaviors

(a) BR reception of an IPv4 packet

- Step 1 BR looks up an appropriate mapping rule with a specific Domain 4rd prefix which has the longest match with an IPv4 destination address in the received IPv4 packet. If the mapping rule is not found, the received packet should be discarded. If the length of CE 4rd prefix associated with the mapping rule does not exceed 32 bits, BR proceeds to step 2. If the length of CE 4rd prefix exceeds 32 bits, BR checks that the received packet contains a complete IPv4 datagram. If the packet is fragmented, BR should reassemble the packet. Once BR can obtain the complete IPv4 datagram, BR proceeds to step 2 as though the datagram has been received in a single packet.
- Step 2 BR generates a CE IPv6 address from the IPv4 destination address or the IPv4 destination address and the destination port based on the mapping rule found in step 1. If the CE IPv6 address can be successfully generated, BR

encapsulates the IPv4 packet in IPv6 and forwards the IPv6 packet via the IPv6 interface. If the length of the IPv6 encapsulated packet exceeds the MTU of the IPv6 interface, the fragmentation should be done in IPv6.

(b) BR reception of an IPv6 packet

- Step 1 If the received IPv6 packet is fragmented, the reassembly should be done in IPv6 at first. Once BR obtains a complete IPv6 packet, BR looks up an appropriate mapping rule with a specific Domain 4rd prefix which has the longest match with an IPv4 source address in the encapsulated IPv4 packet. If the mapping rule is not found, the received IPv6 packet should be discarded. BR derives a CE IPv6

address from the IPv4 source address or the IPv4 source address and the source port in the encapsulated IPv4 packet based on the mapping rule. If the CE IPv6 address is equal to the IPv6 source address in the received IPv6 packet, BR decapsulates the IPv4 packet and then forward it via the IPv4 interface.

(c) CE reception of an IPv4 packet

Step 1 CE looks up an appropriate mapping rule with a specific Doamin 4rd prefix which has the longest match with an IPv4 destination address in the received IPv4 packet. If the mapping rule is found, the CE 4rd prefix must be checked. If the length does not exceeds 32 bits, CE proceeds to step 2. If the length exceeds 32 bits, CE checks that the received IPv4 packet contains a complete IPv4 datagram. If the packet is fragmented, CE should reassemble the packet. Once CE can obtain the complete IPv4 datagram, CE proceeds to step 2 as though the datagram has been received in a single packet. If the mapping rule is not found, CE proceeds to step 2.

Step 2 If the mapping rule is found in step 1, CE derives a IPv6 destination address from the IPv4 destination address or the IPv4 destination address and the destination port

based on the mapping rule. If the IPv6 destination address can be derived successfully, CE encapsulates the IPv4 packet in IPv6 whose destination address is set to the derived IPv6 address. If the mapping rule is not found in step 1, CE encapsulates the IPv4 packet in IPv6 whose destination address is set to BR IPv6 address. Then CE forwards the IPv6 packet via IPv6 interface. If the length of the IPv6 packet exceeds the MTU of the IPv6 interface, the fragmentation should be done in

IPv6. Moreover, if using IPv4 shared address, a Datagram ID in the received IPv4 header must be over-written before encapsulating the IPv4 packet in IPv6. In case of shared IPv4 address, the Datagram ID must be unique among CEs sharing the same IPv4 address. Hence, CE should assign the unique value and set this value to the datagram ID in IPv4 header. This value may be generated from the port-range assigned to the CE to keep the uniqueness among CEs sharing same IPv4 address.

(d) CE reception of an IPv6 packet

Step 1 If the received IPv6 packet is fragmented, the reassembly should be done in IPv6 at first. Once CE obtains a complete IPv6 packet, CE looks up an appropriate mapping rule with a specific Domain 4rd prefix which has the longest match with an IPv4 source address in the encapsulated IPv4 packet. If the mapping rule is found, CE derives a CE IPv6 address from the IPv4 source address or the IPv4 source address and the source port based on the mapping rule and then checks that the IPv6 source address of the received IPv6 packet is matched to it. If the mapping rule is not found, CE checks that the IPv6 source address is matched to BR IPv6 address. In case of success, CE decapsulates the IPv4 packet and forward it via the IPv4 interface.

[8.](#) NAT considerations

NAT44 should be implemented in CPE which has 4rd CE function. The NAT44 must conform that best current practice documented in

[[RFC4787](#)], [[RFC5508](#)] and [[RFC5382](#)]. When there are restricted available port numbers in a given 4rd CE described in [Section 5.1.3](#), the NAT44 must restrict mapping ports within the port-set.

9. ICMP

ICMP message should be supported in 4rd domain. Hence, the NAT44 in 4rd CE must implement the behavior for ICMP message conforming to the best current practice documented in [[RFC5508](#)].

If a 4rd CE receives an ICMP message having ICMP identifier field in ICMP header, NAT44 in the 4rd CE must rewrite this field to a specific value assigned from the port-set described in [Section 5.1.3](#). BR and other CEs must handle this field similar to the port number in tcp/udp header upon receiving the ICMP message with ICMP identifier field.

If a 4rd BR and CE receives an ICMP error message without ICMP identifier field for some errors that is detected inside a IPv6 tunnel, a 4rd BR and CE should replay the ICMP error message to the original source. This behavior should be implemented conforming to the [section 8 of \[RFC2473\]](#). The 4rd BR and CE obtain the original IPv6 tunnel packet storing in ICMP payload and then decapsulate IPv4 packet. Finally the 4rd BR and CE generate a new ICMP error message from the decapsulated IPv4 packet and then forward it.

If a 4rd BR receives an ICMP error message on its IPv4 interface, the 4rd BR should replay the ICMP message to an appropriate 4rd CE. If IPv4 address is not shared, the 4rd BR generates a CE IPv6 address from the IPv4 destination address in the ICMP error message and encapsulates the ICMP message in IPv6. If IPv4 address is shared, the 4rd BR derives an original IPv4 packet from the ICMP payload and generates a CE IPv6 address from the source address and the source port in the original IPv4 packet. If the 4rd BR can generate the CE IPv6 address, the 4rd BR encapsulates the ICMP error message in IPv6 and then forward it to its IPv6 interface.

10. Security Considerations

Spoofing attacks: With consistency checks between IPv4 and IPv6 sources that are performed on IPv4/IPv6 packets received by BR's and CE's ([Section 7](#)), 4rd does not introduce any opportunity for spoofing attack that would not pre-exist in IPv6.

Denial-of-service attacks: In 4rd domains where IPv4 addresses are shared, the fact that IPv4 datagram reassembly may be necessary introduces an opportunity for DOS attacks ([Section 4.4](#)). 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 enablement in both clients and servers so that, where 4rd is supported, it is less and less used.

Routing-loop attacks: This attack may exist in some automatic-tunneling scenarios are documented in [[I-D.ietf-v6ops-tunnel-loops](#)]. They cannot exist with 4rd because each BRs checks that the IPv6 source address of a received IPv6 packet is a CE address [Section 5.1](#).

Attacks facilitated by restricted port set: From hosts that are not subject to ingress filtering of [[RFC2827](#)], some attacks are possible by intervening with faked packets during ongoing transport connections ([[RFC4953](#)], [[RFC5961](#)], [[RFC6056](#)]). The attacks depend on guessing which ports are currently used by target hosts. Using unrestricted port set which mean that are IPv6 is exactly preferable. To avoid this attacks using restricted port set, NAT44 filtering behavior SHOULD be "Address-Dependent Filtering".

[11.](#) IANA Consideration

This document makes no request of IANA.

[12.](#) Acknowledgements

This draft is based on original idea described in [[I-D.despres-softwire-sam](#)]. The authors would like to thank Remi Despres, Mark Townsley, Wojciech Dec and Olivier Vautrin.

[13.](#) References

[13.1.](#) Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [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.
- [RFC2491] Armitage, G., Schuster, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", [RFC 2491](#), January 1999.
- [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.

[13.2.](#) Informative References

- [I-D.despres-software-sam]
Despres, R., "Stateless Address Mapping (SAM) - a Simplified Mesh-Software Model",
[draft-despres-software-sam-01](#) (work in progress),
July 2010.
- [I-D.ietf-software-dual-stack-lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", [draft-ietf-software-dual-stack-lite-11](#) (work in progress), May 2011.
- [I-D.ietf-v6ops-tunnel-loops]
Nakibly, G. and F. Templin, "Routing Loop Attack using IPv6 Automatic Tunnels: Problem Statement and Proposed Mitigations", [draft-ietf-v6ops-tunnel-loops-07](#) (work in

progress), May 2011.

[I-D.mrugalski-dhc-dhcpv6-4rd]

Mrugalski, T., "DHCPv6 Options for IPv4 Residual Deployment (4rd)", [draft-mrugalski-dhc-dhcpv6-4rd-00](#) (work in progress), July 2011.

[I-D.operators-softwire-stateless-4v6-motivation]

Murakami, et al.

Expires March 25, 2012

[Page 17]

Internet-Draft

IPv4 Residual Deployment

September 2011

Boucadair, M., Matsushima, S., Lee, Y., Bonness, O., Borges, I., and G. Chen, "Motivations for Stateless IPv4 over IPv6 Migration Solutions", [draft-operators-softwire-stateless-4v6-motivation-02](#) (work in progress), June 2011.

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

[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.

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

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

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

[RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-
Protocol Port Randomization", [BCP 156](#), [RFC 6056](#),
January 2011.

Murakami, et al.

Expires March 25, 2012

[Page 18]

Internet-Draft

IPv4 Residual Deployment

September 2011

Authors' Addresses

Tetsuya Murakami
IP Infusion
1188 East Arques Avenue
Sunnyvale
USA

Email: tetsuya@ipinfusion.com

Ole Troan
cisco
Oslo
Norway

Email: ot@cisco.com

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

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

