Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 20, 2018

C. Bao
X. Li
Y. Zhai
W. Shang
CERNET Center/Tsinghua
University

October 17, 2017

dIVI: Dual-Stateless IPv4/IPv6 Translation
draft-xli-behave-divi-08

Abstract

This document presents the concepts and the implementations of address-sharing dual-stateless IPv4/IPv6 translation (dIVI).

The dIVI is an extension of 1:1 stateless IPv4/IPv6 translation with features of IPv4 address sharing and dual translation. The major benefits of those extensions are using public IPv4 addresses more effectively and removing the requirements of DNS64 and ALG, without loosing the stateless, end-to-end address transparency and bidirectional-initiated communications.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The experiences for the IPv6 deployment in the past 10 years strongly indicate that for a successful transition, the communication between IPv4 and IPv6 address families should be supported.

Recently, the stateless and stateful IPv4/IPv6 translation methods are developed and became the IETF standards. The original stateless IPv4/IPv6 translation (stateless 1:1 IVI) is scalable, maintains the end-to-end address transparency and support both IPv6 initiated and IPv4 initiated communications [RFC6052], [RFC6144], [RFC6145], [RFC6147], [RFC6219]. But it can not use the IPv4 addresses effectively. The stateful IPv4/IPv6 translation can share the IPv4 addresses among IPv6 hosts, but it only supports IPv6 initiated communication [RFC6052], [RFC6144], [RFC6145], [RFC6146], [RFC6147]. In addition, both stateless and stateful IPv4/IPv6 translation technologies require the application layer gateway (ALG) for the applications which embed IP address literals.

In this document, we present concepts and the implementations of the dual-stateless IPv4/IPv6 translation (dIVI). The dIVI can solve the IPv4 address sharing and the ALG problems mentioned above, though still keeps the stateless, end-to-end address transparency and supporting of both IPv6 initiated and IPv4 initiated communications. The dIVI is in the family of stateless IPv4/IPv6 translation, along with IVI and dIVI-PD [I-D.xli-behave-divi-pd]. These techniques are used for the following scenarios with second translation extension defined in [RFC6144].

o Scenario 1: IPv4 hosts via an IPv6 network to the IPv4 Internet.

o Scenario 2: The IPv4 Internet via an IPv6 network to IPv4 hosts.

o Scenario 5: An IPv6 network via an IPv4 network to IPv4 hosts.

o Scenario 6: IPv4 hosts via an IPv4 network to an IPv6 network.

2. Applicability

The address sharing dual stateless IPv4/IPv6 translation (dIVI) is shown in the following figure.

```
                                                   -----     ------
                                                .-|CPE.0|---|Host.0|
                                               /   -----     ------
           ------                   -----     |
         /  The   \    ------     /  An   \   |    -----     ------
        |   IPv4   |--| 1:N  |---|   IPv6  |------|CPE.1|---|Host.1|
         \Internet/   | XLAT |    \Network/   |    -----     ------
           ------      ------       -----     |
                                              |\   -----     ------
                                              |  -|CPE.2|---|Host.2|
                                              |    -----     ------
                                              |         ......
                                               \   -----     ------
                                                 -|CPE.K|---|Host.K|
                                                   -----     ------
                                                        ......
```

Figure 1: dIVI

Where

o The 1:N XLAT (first translator) is the IPv4/IPv6 translator perform stateless 1:N translation between the IPv4 Internet and an IPv6 network.

o The CPE.0 to CPE.K (second translators) are 1:1 IPv6/IPv4 translators/IPv6 routers which also perform port mapping function for Host.x when it is communicating with the IPv4 Internet with IPv4 address sharing.

o The Host.1 to Host.N are the dual-stack hosts.

The IPv6 network routes the IPv6 more specific routes of the IPv4-translatable addresses (longer than /64) defined in Section 5 of this document to corresponding CPEs. In the case of prefix delegation (shorter than /64), the dual stateless IPv4/IPv6 translation with prefix delegation (dIVI-pd) defined in [I-D.xli-behave-divi-pd] can be used.

3. Terminologies

This document uses the terminologies defined in [RFC6144].

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

4. Port Mapping Algorithm and Address Format Extension

4.1. Port Mapping Algorithm

The dual stateless IPv4/IPv6 translation (dIVI) uses the port mapping algorithm defined in [I-D.bcx-behave-address-fmt-extension].

For given sharing ratio (R) and the maximum number of continue ports (M), the generalized modulus algorithm is defined as

1. The port number (P) of a given PSID (K) is composed of

   P = R*M*j + M*K + i

   Where
   o PSID: K=0 to R-1
   o Port range index: j = (1024/M)/R to ((65536/M)/R)-1, if the well-known port numbers (0-1023) are excluded.
   o Port continue index: i=0 to M-1

2. The PSID (K) of a given port number (P) is determined by

   K = (floor(P/M)) % R

   Where
   o % is modular operator
   o floor(arg) is a function returns the largest integer not greater than arg

3. The well-known port number (0-1023) can be used, if additional port mapping rule is defined.

For example, for R=128, M=4

```
          Port range-1             | Port rang-2              | Port
 -------------------------------------------------------------------
 PSID=0   | 1024, 1025, 1026, 1027, | 1536, 1537, 1538, 1539, | 2048
 PSID=1   | 1028, 1029, 1030, 1031, | 1540, 1541, 1542, 1543, | ....
 PSID=2   | 1032, 1033, 1034, 1035, | 1544, 1545, 1546, 1547, | ....
 PSID=3   | 1036, 1037, 1038, 1039, | 1548, 1549, 1550, 1551, | ....
 ...
 PSID=127 | 1532, 1533, 1534, 1535, | 2044, 2045, 2046, 2047, | ....
```

Figure 2: Example

4.2. Address Format Extensions

The dual stateless IPv4/IPv6 translation (dIVI) uses the address format extension defined in [I-D.bcx-behave-address-fmt-extension].

```
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|PL| 0-------------32--40--48--56--64--72--80--88--96--104-112-120-|
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|32|     prefix    |v4(32)         | u | PSID  | 0              | Q |
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|40|     prefix        |v4(24)     | u |(8)| PSID | 0           | Q |
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|48|     prefix            |v4(16) | u | (16)  | PSID | 0       | Q |
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|56|     prefix                |(8)| u |  v4(24)   | PSID | 0   | Q |
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|64|     prefix                    | u |   v4(32)      |PSID |0| Q |
+--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Figure 3: Address Format

Where PL designates the prefix length.

The PSID prefix length (Q) is encoded in the last octet (bits 120-127) to indicate the number of ports can be used. When Q=0, the extended address format will become the address format defined in [RFC6052]. The relations between Q, the sharing ratio (R), the maximum continue port range (M) and the number of ports can be shown in the following figure.

Q	Ratio	Maximum M	# of Ports
0	1:1	65,536	65,536
1	1:2	32,786	32,786
2	1:4	16,384	16,384
3	1:8	8,192	8,192
4	1:16	4,096	4,096
5	1:32	2,048	2,048
6	1:64	1,024	1,024
7	1:128	512	512
8	1:256	256	256
9	1:512	128	128
10	1:1,024	64	64
11	1:2,048	32	32
12	1:4,096	16	16

Figure 4: Port Range

4.3. Stateless Translation Algorithm with Address Sharing

For the stateless translation, the IPv6 nodes are required to follow the port number range defined by the extended IPv4-translatable address format when communicating with the IPv4 Internet. The port number handling algorithm is:

- o If the packets are from the IPv4 Internet to an IPv6 network, the IPv4 source addresses are translated to the IPv4-converted addresses and the source port numbers are unchanged before and after translation; the IPv4 destination addresses are translated to the extended IPv4-translatable addresses based on the destination port number and the destination port numbers are unchanged before and after translation. Note that this means that only a specific IPv6 node can receive the packets for a specific port number.

- o If the packets are from an IPv6 network to the IPv4 Internet, the IPv6 source addresses and the source port numbers are checked, if the source port number matches the port number range defined by the extended IPv4-translatable address format, the IPv6 source addresses (which are the IPv4-translatable addresses) are translated to the IPv4 addresses and the source port numbers are unchanged before and after translation; the destination IPv6 addresses (which are the IPv4-converted addresses) are translated to the IPv4 destination addresses and the destination port numbers are unchanged before and after translation. However, if the source port number does not match the port number range defined by the extended IPv4-translatable address format, the packets will be dropped.

4.4. Partial-state Translation Algorithm with Address Sharing

Stateless translation requires that IPv6 nodes generate source port number in the range defined by the extended IPv4-translatable address. If this condition does not hold, the partial-state translation algorithm can be used, which can map the random source port number to the extended IPv4-translatable address defined source port number range. Due to the requirement of the states in the translator for the port mapping (no states required for the address mapping), it is named partial state. The technical details of the port mapping state maintaining algorithm is an implementation issue (see Section 6.2).

5. Dual Stateless Translation

In general dual stateful IPv4/IPv6 translations (IPv4-IPv6-IPv4) are

considered harmful, since the two translators cannot synchronize the parameter to translate the IPv4 address to its original format. However, the dual stateless IPv4/IPv6 translation (IPv4-IPv6-IPv4) can translate the IPv4 address to its original format based on algorithm. There are several reasons for using dual stateless IPv4/IPv6 translation.

- o The second translator (CPE) performs the port-set mapping algorithm discussed in Section 4.1, therefore, there is no need to modify the host with IPv4 address sharing.

- o ALG cannot be developed for some applications, or has not been developed during the early transition stage.

- o DNS64 is not allowed (due to DNSSec, etc.).

- o All the IPv6 packet processing tools (routing, policy based routing, packet filtering, traffic shaping based on layer 3 and layer 4 can be used for dual stateless translation, while new tools are required for encapsulation and tunneling.

5.1. Header Translation and MTU Handling

The general header and ICMP translation specifications are defined in [RFC6145].

Special MTU and fragmentation actions must be taken in the case of dual translation.

5.2. Port Mapping Algorithm in CPE

The port mapping algorithm is straightforward. The port mapping device maintains a database of allowed port numbers defined by the extended IPv4-translatable address format. If the packets from the host contains the source port number which do not match the port number range defined by the extended IPv4-translatable address format, the CPE will map the source port number to an allowed one and keep the record in the database for mapping back the returning packets and all the packets in the same session.

The port number database can be refreshed via the corresponding transport layer flags for TCP or via timeout for UDP sessions.

6. Implementation Considerations

6.1. Host implementation

For the wireless mobile Internet environment, it is not difficult to modify the operating system of the mobile device, therefore it possible to integrate the port mapping algorithm and the second IPv4/IPv6 translation function in the mobile device, which is an IPv6-only host to the network and has a dual-stack socket API for the applications running on this host.

6.2. Port Mapping in the Core Translator

The port mapping can be performed in the core translator, in this case, there is no need to have the second translator (CPE.x) and no need to modify the IPv6 host for the port restriction requirements.

7. Security Considerations

There are no security considerations in this document.

8. IANA Considerations

This memo adds no new IANA considerations.

Note to RFC Editor: This section will have served its purpose if it correctly tells IANA that no new assignments or registries are required, or if those assignments or registries are created during the RFC publication process. From the author's perspective, it may therefore be removed upon publication as an RFC at the RFC Editor's discretion.

9. Acknowledgments

The authors would like to acknowledge the following contributors in the different phases of the address-sharing IVI and dIVI development: Maoke Chen, Hong Zhang, Weifeng Jiang, Yuncehng Zhu and Guoliang Han.

The authors would like to acknowledge the following contributors who provided helpful inputs: Dan Wing, Fred Baker, Dave Thaler, Randy Bush, Kevin Yin, Wojciech Dec and Rajiv Asati.

10. References

10.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, DOI 10.17487/RFC2473, December 1998, <https://www.rfc-editor.org/info/rfc2473>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <https://www.rfc-editor.org/info/rfc6052>.

[RFC6144] Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", RFC 6144, DOI 10.17487/RFC6144, April 2011, <https://www.rfc-editor.org/info/rfc6144>.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, DOI 10.17487/RFC6145, April 2011, <https://www.rfc-editor.org/info/rfc6145>.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <https://www.rfc-editor.org/info/rfc6146>.

[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, DOI 10.17487/RFC6147, April 2011, <https://www.rfc-editor.org/info/rfc6147>.

[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, DOI 10.17487/RFC6219, May 2011, <https://www.rfc-editor.org/info/rfc6219>.

10.2. Informative References

[I-D.bcx-behave-address-fmt-extension]
Bao, C. and X. Li, "Extended IPv6 Addressing for Encoding Port Range", draft-bcx-behave-address-fmt-extension-10 (work in progress), January 2017.

[I-D.xli-behave-divi-pd]
Sun, Q., Asati, R., Xie, C., Li, X., Dec, W., and C. Bao, "dIVI-pd: Dual-Stateless IPv4/IPv6 Translation with Prefix Delegation", draft-xli-behave-divi-pd-01 (work in progress), September 2011.

Appendix A. Address Format and Workflow Examples

The prefix we selected is 2001:da8:a4a6::/48. The formats of the IPv4-converted address and the IPv4-translatable address are:

```
                         IPv4-converted address
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |PL| 0-------------32--40--48--56--64--72--80--88--92--104-112-120-|
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |48|     prefix            |v4(16) | u | (16)  |      0            |
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

                         IPv4-translatable address
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |PL| 0-------------32--40--48--56--64--72--80--88--92--104-112-120-|
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |48|     prefix            |v4(16) | u | (16)  |PSID|    0     | Q |
 +--+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Figure 5: Address format

The sharing ratio R=16, so the PSID has 4 bits (Q=4) and each PSID can have (4096-1024/16)= 4032 concurrent port numbers to use. The continue port range M=4, so each PSID will have 1008 available port range.

The host in the IPv4 Internet is 202.112.35.254 and the corresponding IPv4-converted address is 2001:da8:a4a6:ca70:23:fe00::

The shared IPv4 address is 202.38.117.1 with sharing ratio 16, the corresponding IPv4-translatable addresses are

```
PSID=0    2001:da8:a4a6:ca26:75:100::4
PSID=1    2001:da8:a4a6:ca26:75:110::4
PSID=2    2001:da8:a4a6:ca26:75:120::4
PSID=3    2001:da8:a4a6:ca26:75:130::4
PSID=4    2001:da8:a4a6:ca26:75:140::4
PSID=5    2001:da8:a4a6:ca26:75:150::4
PSID=6    2001:da8:a4a6:ca26:75:160::4
PSID=7    2001:da8:a4a6:ca26:75:170::4
PSID=8    2001:da8:a4a6:ca26:75:180::4
```

```
PSID=9    2001:da8:a4a6:ca26:75:190::4
PSID=10   2001:da8:a4a6:ca26:75:1a0::4
PSID=11   2001:da8:a4a6:ca26:75:1b0::4
PSID=12   2001:da8:a4a6:ca26:75:1c0::4
PSID=13   2001:da8:a4a6:ca26:75:1d0::4
PSID=14   2001:da8:a4a6:ca26:75:1e0::4
PSID=15   2001:da8:a4a6:ca26:75:1f0::4
```

A.1. The address-sharing host on an IPv6 network initiates communication

An address-sharing Host.0 (202.38.117.1) in an IPv6 network behind CPE initiates communication with Host 202.112.35.254:80 in the IPv4 Internet

```
On the Host.0
Src#p= 202.38.117.1#1881 (random port)
Dst#p= 202.112.35.254#80 (server port)

On an IPv6 network
Src#p= [2001:da8:a4a6:ca26:75:0100::4]#8192 (CPE mapped port)
Dst#p= [2001:da8:a4a6:ca70:23:fe00::]#80 (server port)

On the IPv4 Internet
Src#p= 202.38.117.1#8192 (CPE mapped port)
Dst#p= 202.112.35.254#80 (server port)
```

Figure 6: Workflow Example 1

The returning packets reverse the Src and Dst, the CPE maps the "CPE mapped port (8192)" back to the original "random port (1881)".

A.2. A host in the IPv4 Internet initiates communication

A host 202.112.35.254 in the IPv4 Internet initiates communication to the address-sharing Host.0 (202.38.117.1#4096)

```
On the IPv4 Internet
Src#p= 202.112.35.254#36567 (random port)
Dst#p= 202.38.117.1#4096 (server port)

On an IPv6 network
Src#p= [2001:da8:a4a6:ca70:23:fe00::]#36567 (random port)
Dst#p= [2001:da8:a4a6:ca26:75:0100::4]#4096 (server port)

On the Host.0
Src#p= 202.112.35.254#36567 (random port)
Dst#p= 202.38.117.1#4096 (server port)
```

Figure 7: Workflow Example 2

The returning packets reverse the Src and Dst.

Authors' Addresses

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

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

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

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

Yu Zhai
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing 100084
CN

Phone: +86 10-62785983
Email: jacky.zhai@gmail.com

Wentao Shang
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing 100084
CN

Phone: +86 10-62785983
Email: wentaoshang@gmail.com