Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 8, 2010

X. Li
C. Bao
CERNET Center/Tsinghua University
C. Metz
Cisco Systems, Inc.

March 7, 2010

Stateless/Partial-state 1:N Network Address and Protocol Translation between IPv4 and IPv6 nodes
draft-xli-behave-xlate-partial-state-01

Abstract

This document presents concepts and implementations of stateless/ partial-state 1:N network address and protocol translation between IPv4 and IPv6 nodes.

Stateless 1:N translation keeps the features of stateless, end-to-end address transparency and bidirectional-initiated communications of the original stateless translation (1:1 IVI), in addition it can utilize the IPv4 addresses more effectively. However, it requires the modification of the IPv6 end systems or deploying home gateways. By introducing "partial state" in the translator, this requirement is not necessary.

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 becoming the IETF standards [I-D.ietf-behave-v6v4-framework], [I-D.ietf-behave-v6v4-xlate], [I-D.ietf-behave-v6v4-xlate-stateful]. 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 [I-D.ietf-behave-v6v4-framework], [I-D.ietf-behave-v6v4-xlate]. But it can not use the IPv4 addresses effectively. The IPv4 address depletion problem makes the deployment of the stateless 1:1 IVI challenging, in particular as the number of IPv6 hosts increases. The stateful IPv4/IPv6 translation can share the IPv4 addresses among IPv6 hosts, but it only supports IPv6 initiated communication [I-D.ietf-behave-v6v4-framework], [I-D.ietf-behave-v6v4-xlate-stateful]. Rely on session initiated states, the stateful translation cannot support the end-to-end address transparency and costs more compared with the stateless translation.

We then try to find a translation scheme which keeps end-to-end address transparency can utilize IPv4 address effectively. This turns into stateless and partial-state translators. Stateless 1:N translation is an extensions of the stateless translation, which keeps stateless, end-to-end address transparency and bidirectional-initiated communications. By limiting useable port range for different IPv6 addresses, several IPv6 hosts can share a single IPv4 address using limited port range. The partial-state 1:N translator is an further extensions of the stateless 1:N translation. It tracks and maps port range of IPv6 hosts using a simplified scheme of stateful translation. Therefore, the modification of IPv6 hosts is not required. In addition, the partial-state 1:N translation has the following features:

1. Less state and complexity than full-blown stateful.

2. Supports IPv4-initiated connectivity.

3. Require less work to log translation bindings.

The stateless/partial-state 1:N translation are solutions for the following scenarios [I-D.ietf-behave-v6v4-framework].

- o Scenario 1: An IPv6 network to the IPv4 Internet.
- o Scenario 2: The IPv4 Internet to an IPv6 network.
- o Scenario 5: An IPv6 network to an IPv4 network.
- o Scenario 6: An IPv4 network to an IPv6 network.

2. Terminologies

This document uses the terminologies defined in
[I-D.ietf-behave-v6v4-framework],
[I-D.ietf-behave-v6v4-xlate-stateful],
[I-D.ietf-behave-address-format].

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].

3. Stateless 1:N Translation

In order to provide IPv4 connectivity for multiple IPv6 hosts sharing a single IPv4 address, the port number multiplexing technique is used. This is to say that a single IPv4 address can be shared for multiple IPv6 hosts under the condition that these individual hosts can only use a subset of the 65,536 port numbers when communicating with the IPv4 Internet. For example, if the port multiplexing ratio is 128, each host with IPv4- translatable address can use 512 concurrent port numbers when communicating with IPv4 Internet. Note that there is no port number restriction when these IPv6 hosts communicate with the IPv6 Internet.

3.1. Address-sharing algorithm

The stateless 1:N translation is shown in the following figure.

```
                                              .-------|Host0| A1/(P%N)+0
                                             /
         ------                   -----     |
       /  The   \     ------    /  An   \   |
      |  IPv4    |--|1:N   |---|  IPv6   |------------|Host1| A1/(P%N)+1
       \Internet/   |XLATE |    \Network/   |
         ------      ------       -----     |
                                            |\
                                            |  -------|Host2| A1/(P%N)+2
                                            |
                                            |
                                             \
                                               -------|HostK| A1/(P%N)+K
```

Figure 1: Stateless 1:N translation

In the above figure, the Host0, Host1, Host2, ..., HostK are sharing the same IPv4 address A1, but port number range for different hosts are not overlapped. Therefore, when these IPv6 hosts communicate with the IPv4 Internet via the translator, it looks like a single host with IPv4 address A1 communicating with the IPv4 Internet.

We use the Modulus Operator to define the port number range. If the multiplexing ratio is N, then:

o For host K, the allowed port number (P) are P=j*N + K (j=0, 1, ..., N-1).

o For the destination port number (P), the packets will be sent to host K=(P%N) (% is the Modulus Operator).

For example: If N=256, then host K=5 is only allowed to use port numbers 5, 261, 517, 773, ..., 65,285 as the source port, while the packets with these port numbers as the destination port number will be send to host K=5.

3.2. Extended address format

In order to perform the stateless translation between the IPv4 and IPv6, both IPv4-converted and IPv4-translatable address are required [I-D.ietf-behave-v6v4-framework], [I-D.ietf-behave-address-format].

The IPv4-converted addresses are used to represent IPv4 addresses in IPv6, as shown in the following figure.

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

Figure 2: IPv4-converted address format

There is no port number coding required for the IPv4-converted address.

The IPv4-translatable addresses are used to represent IPv6 addresses in IPv4, We use 16-bit suffix to encode the range of the port number as shown in the following figure.

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

Figure 3: Extended IPv4-translatable address format

Where, we use reserved 16-bits (Coding) to encode the port number range based on the Modulus Operator.

The most significant 4 bits define the multiplexing ratio and the least significant 12 bits define the index of the host, as shown in the following figure.

(4 bits)	Index Range(12 bits)	Multx ratio	# of Ports
0	000-000	1	65,536
1	000-001	2	32,768
2	000-003	4	16,384
3	000-007	8	8,192
4	000-00f	16	4,096
5	000-01f	32	2,048
6	000-03f	64	1,024
7	000-07f	128	512
8	000-0ff	256	256
9	000-1ff	512	128
A	000-3ff	1,024	64
B	000-7ff	2,048	32
C	000-fff	4,096	16

Figure 4: Transport layer port number coding

3.3. Transport address mapping algorithm

For the stateless 1:N translation, the IPv6 end systems 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 IPv4 to IPv6, the IPv4 source addresses are translated to the IPv4-converted addresses and the source port numbers are unchanged; 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.

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

3.4. Protocol translation

The protocol translation is defined in [I-D.ietf-behave-v6v4-xlate], except the address translation, which is defined in sections 3.2 and 4.2 of this document.

3.5. IPv6 end system requirements

The IPv6 end systems MUST follow the port number range defined by the extended IPv4-translatable addresses. The behavior of the IPv6 end system when communicating with the IPv4 Internet are:

- o If the IPv6 end system is used as a server, different well-known ports will be served by different IPv6 hosts.

- o If the IPv6 end system is used as a client, the end system must generate the source port numbers in the range defined by the extended IPv4-translatable address format. This can be done by the modification of IPv6 end systems.

4. Partial-state 1:N Translation

Stateless 1:N translation requires that IPv6 end system generate source port number in the range defined by the extended IPv4-translatable address. Then we introduce partial-state 1:N translation, which consists of session table and port number mapping algorithm in translator without the modification of IPv6 end systems.

4.1. Session tables

A partial-state translator has three session tables: one for TCP sessions, one for UDP sessions, and one for ICMP Query sessions. For TCP and UDP, the session table contains address and port number. For ICMP Query, the session table contains address and identifier. Each entry in the session tables keeps information on the state of the corresponding session.

- o UDP session is initiated based on port number mapping algorithm defined in section 4.2 and a timer that tracks the remaining lifetime of the UDP session. When the timer expires, the UDP session is deleted.

- o TCP Session is based on port number mapping algorithm defined in section 4.2 and TCP state machine. When the state machine reaches the termination state, the TCP session is deleted.

- o ICMP query session is initiated based on port number mapping algorithm defined in section 4.2 and a timer that tracks the remaining lifetime of the ICMP Query session. When the timer expires, the session is deleted.

4.2. Port number mapping algorithm

For source port number of the packet from IPv6 to IPv4:

- o If source port number is not in the range defined by the extended IPv4-translatable address, the translator will check if there is an entry in the session table.

  * If the entry exists, the translator will use that entry to map source port to the one in the session table.

  * If the entry does not exist, the translator will create an entry in the session table to map the source port to an allowed range.

- o If source port number is in the range defined by the extended IPv4-translatable address, the translator will not create an entry in the session table.

For destination port number of the packet from IPv4 to IPv6:

- o If the mapping entry exists in the session table, map the destination port number to the one in the session table.

- o If the mapping entry does not exist in the session table, keep the original destination port number.

For destination port number of the packet from IPv6 to IPv4 and for source port number of the packets from IPv4 to IPv6, no special algorithm is required and there is no port number change.

The reason we call this partial-state is that:

1. The address mapping is fully algorithm based, as defined in section 3.3. The states are used for port number mapping only, as defined in section 4.2.

2. There will be no session table created if the the source port number from IPv6 to IPv4 is in the range defined by the extended IPv4-translatable address.

3. For the destination port number of the packet from the IPv4 to IPv6, there will be no session table created

5. Operation considerations

5.1. Routing

The routing follows the general IPv4/IPv6 routing principle, i.e. "more specifics win", same as the original stateless 1:1 IVI. [I-D.xli-behave-ivi].

5.2. DNS

The DNS handling is referring to DNS64 [I-D.ietf-behave-dns64] and DNS46 [I-D.xli-behave-dns46-for-stateless].

5.3. ALG

The ALG related issue is discussed in [I-D.ietf-behave-v6v4-framework].

6. Deployment Considerations

The stateless 1:N translation requires that the IPv6 hosts served by the translator generate the port numbers in the range defined extended IPv4-translatable addresses.

6.1. Using Modified IPv6 Hosts in an IPv6 Network

Stateless translation can be deployed using modified IPv6 hosts. These IPv6 hosts are using extended IPv4-translatable addresses and the IPv6 hosts will generate the source port number in the range defined by extended IPv4-translatable addresses. In other words, the end systems maintain the port-number mapping states.

```
                                                    -----------
                                                 .-|Host0 (mdf)| A1/(P%N)+0
                        ------                  /   -----------
            ------     |State-|      -----     |
          /  The   \   |less  |    /  An   \   |    -----------
         |  IPv4    |--|1:N   |---|  IPv6   |------|Host1 (mdf)| A1/(P%N)+1
          \Internet/   |XLATE |    \Network/   |    -----------
            ------      ------       -----     |
                                               |\   -----------
                                               |  -|Host2 (mdf)| A1/(P%N)+2
                                               |    -----------
                                               |
                                                \   -----------
                                                  -|HostK (mdf)| A1/(P%N)+K
                                                    -----------
```

Figure 5: Using Modified IPv6 Hosts

6.2. Using Unmodified IPv6 Hosts in an IPv6 Network

Partial-state translation can be deployed using unmodified IPv6 hosts. These IPv6 hosts are using extended IPv4-translatable addresses and translator (XLATE) keeps the states for the port-number mapping.

```
                                                    -----------
                                                 .-|   Host0   | A1/(P%N)+0
                        ------                  /   -----------
            ------     |Partia|      -----     |
          /  The   \   |-state|    /  An   \   |    -----------
         |  IPv4    |--|1:N   |---|  IPv6   |------|   Host1   | A1/(P%N)+1
          \Internet/   |XLATE |    \Network/   |    -----------
            ------      ------       -----     |
                                               |\   -----------
                                               |  -|   Host2   | A1/(P%N)+2
                                               |    -----------
                                               |
                                                \   -----------
                                                  -|   HostK   | A1/(P%N)+K
                                                    -----------
```

Figure 6: Using Unmodified IPv6 Hosts

6.3. Mixed Environment in an IPv6 Network

In a mixed environment, partial-state translator can be deployed. If the IPv6 packets contain the port numbers which are not in the range defined by extended IPv4-translatable addresses, the states will be

created in the translator. Otherwise, no states created and maintained in the translator.

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, Yu Zhai, Wentao Shang, Weifeng Jiang and Yuncehng Zhu.

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

10. References

10.1. Normative References

[I-D.ietf-behave-address-format]
Huitema, C., Bao, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", draft-ietf-behave-address-format-04 (work in progress), January 2010.

[I-D.ietf-behave-dns64]
Bagnulo, M., Sullivan, A., Matthews, P., and I. Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-dns64-07 (work in progress), March 2010.

[I-D.ietf-behave-v6v4-framework]
Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-07 (work in progress), February 2010.

[I-D.ietf-behave-v6v4-xlate]
Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", draft-ietf-behave-v6v4-xlate-10 (work in progress), February 2010.

[I-D.ietf-behave-v6v4-xlate-stateful]
Bagnulo, M., Matthews, P., and I. Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-v6v4-xlate-stateful-08 (work in progress), January 2010.

[I-D.xli-behave-dns46-for-stateless]
Li, X. and C. Bao, "DNS46 for the IPv4/IPv6 Stateless Translator", draft-xli-behave-dns46-for-stateless-02 (work in progress), February 2010.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

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

10.2. Informative References

[CERNET] "CERNET Homepage: http://www.edu.cn/english_1369/index.shtml".

[CNGI-CERNET2]
"CNGI-CERNET2 Homepage: http://www.cernet2.edu.cn/index_en.htm".

[I-D.xli-behave-ivi]
Li, X., Bao, C., Chen, M., Zhang, H., and J. Wu, "The CERNET IVI Translation Design and Deployment for the IPv4/IPv6 Coexistence and Transition", draft-xli-behave-ivi-07 (work in progress), January 2010.

[RFC3849] Huston, G., Lord, A., and P. Smith, "IPv6 Address Prefix Reserved for Documentation", RFC 3849, July 2004.

[RFC5737] Arkko, J., Cotton, M., and L. Vegoda, "IPv4 Address Blocks

Reserved for Documentation", RFC 5737, January 2010.

[dIVI] "Test homepage for the dIVI: http://202.38.97.114:8056/test.html".

Authors' Addresses

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

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

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

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

Chris Metz
Cisco Systems, Inc."
3700 Cisco Way
San Jose CA 95134
USA

Phone:
Email: chmetz@cisco.com