Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 11, 2011

M. Boucadair
C. Jacquenet
France Telecom
J. Song
Q. Niu
ZTE Corporation
October 8, 2010

Procedure to bypass DS-Lite AFTR
draft-boucadair-softwire-cgn-bypass-03.txt

Abstract

This document proposes a solution to avoid the use of two stateful DS-Lite AFTR devices when both end-points are located behind different AFTR devices. For this purpose a new IPv6 extension header, called Tunnel Endpoint Extension Header (TEEH), is defined. The proposed procedure encourages the use of IPv6 between DS-Lite AFTR nodes as a means to avoid the unnecessary crossing of AFTR devices. A Flow Label based solution is also described.

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

Copyright Notice

Table of Contents

1. Introduction

1.1. Purpose

The main purpose of this document is to investigate solutions to avoid the solicitation of some of the (AFTR-embedded) NAT capabilities along the path between two hosts located behind AFTR devices.

The advantages of this procedure include:

- Better one-way delay: No need to check the payload in the originating AFTR and no need to execute ALG operations twice;
- Optimised routing path;
- Better use of available AFTR resources;
- Enhance robustness: an AFTR device is withdrawn from the data path. The stateful nature of DS-Lite AFTR devices will affect the overall performance of the communication. This performance may be even more affected when two AFTR devices need to be crossed to establish the communication.

1.2. Terminology

Within this memo, the term AFTR is used to refer to both following schemes:

- an AFTR function embedded in a router, and/or
- a standalone AFTR with limited routing capabilities (redirection capabilities to the AFTR are being enabled in an external router).

An outbound AFTR is referred to as Source AFTR.

An inbound AFTR is called a Target AFTR.

In the example illustrated in Figure 1, if we suppose that A initiates a communication towards B, AFTR1 is the Source AFTR and AFTR2 is the Target AFTR.

"===" in the figure represents flows and not links. Furthermore, AFTRs may not be part of the optimal routing path between A and B.

```
 +-+                   +-----+            +-----+                  +-+
 |A|====IPv4-in-IPv6==>|AFTR1|============|AFTR2|===IPv4-in-IPv6===>|B|
 +-+                   +-----+            +-----+                  +-+
                     Source AFTR        Target AFTR
```

Figure 1: Source and Target AFTR

1.3. Contribution of this Draft

This document proposes a solution to avoid invoking NAT capabilities when several DS-Lite AFTR devices [I-D.ietf-softwire-dual-stack-lite] are involved in the data path. This document encourages the use of IPv6 for forwarding traffic between two AFTR devices.

This memo focuses primarily on the AFTR devices deployed in the same administrative domain. AFTRs located in distinct administrative domains are out of scope.

This document does not make any assumption on the services that may require the establishment of direct communications between hosts located behind AFTR devices. Examples of services would be P2P or hosting FTP/HTTP/SIP server behind a DS-Lite CPE.

In order to offload AFTR devices, application-specific solutions (e.g., [I-D.carpenter-behave-referral-object] [I-D.boucadair-mmusic-altc], [I-D.boucadair-dispatch-ipv6-atypes]) may be required to be implemented in order to prefer native IPv6 communications rather than crossing AFTR devices.

The implementation of the proposed procedure is not motivated in a context where the percentage of traffic involving two AFTR devices is minor (e.g., 1%). Nevertheless, as a side effect, Tunnel Endpoint Extension Header (TEEH) (Section 3) may be used to withdraw an AFTR from the data path, when both participants are managed by the same AFTR.

When TEEH is not supported, Two alternatives solutions are described in Section 5 and Appendix A.

2. Overall Scenarios

This section provides an overview of targeted scenarios.

Figure 2 illustrates the communication between two hosts that are located behind an AFTR device. Two NAT operations are required to be performed for the establishment of successful communication between A and B. The stateful nature of a DS-Lite AFTR device will presumably affect the overall performance of the communication. This performance may be even more affected when two AFTR devices need to be crossed to establish the communication.

Prior to sending datagrams to B, A has retrieved the IPv4 public address of B owing to DNS resolution, third party referral, etc.

```
+-+                   +------+           +------+                   +-+
|A|====IPv4-in-IPv6==>|AFTR1|============|AFTR2|====IPv4-in-IPv6===>|B|
+-+                   +------+           +------+                   +-+
                       NAT44               NAT44
```

Figure 2: Nominal behaviour

A first optimisation scenario is shown in Figure 3 where NAT capabilities of the Source AFTR are not solicited. A second optimisation scenario is shown in Figure 4 where NAT capabilities of the Target AFTR are not solicited. The latter is not a valid scenario since the destination is seen with a public IPv4 address which is managed by the Target AFTR (consequently, a NAT44 state must be instantiated in the Target AFTR). The last configuration, illustrated in Figure 5, aims at avoiding the use of NAT capabilities in both Source and Target AFTRs. This configuration is impossible to implement since the remote destination must always be seen with an external public IPv4 address (and/or an IPv6 one). Having an external IPv4 address means that a AFTR has assigned an IPv4 address and port number for that host. Therefore, all the incoming IPv4 traffic must cross that AFTR.

```
+-+                   +------+           +------+                   +-+
|A|====IPv4-in-IPv6==>|AFTR1|============|AFTR2|====IPv4-in-IPv6===>|B|
+-+                   +------+           +------+                   +-+
                      No_NAT44             NAT44
```

Figure 3: Avoid Source NAT44

```
+-+                   +------+           +------+                   +-+
|A|====IPv4-in-IPv6==>|AFTR1|============|AFTR2|====IPv4-in-IPv6===>|B|
+-+                   +------+           +------+                   +-+
                       NAT44             No_NAT44
```

Figure 4: Avoid Target NAT44

```
 +-+                   +------+            +------+                +-+
 |A|====IPv4-in-IPv6==>|AFTR1|============|AFTR2|====IPv4-in-IPv6===>|B|
 +-+                   +------+            +------+                +-+
                       No_NAT44            No_NAT44
```

Figure 5: Avoid all NAT44

3. Tunnel Endpoint Extension Header

TEEH is a new IPv6 extension header which is used to inform the remote party about the destination IPv6 address to be used when issuing a response. Particularly, TEEH is used by the Source AFTR to inform the Target AFTR about the IPv6 address of a customer's device attached to the Source AFTR. Therefore, the Target AFTR acts as an inbound AFTR for that customer's device.

The format of the Tunnel Endpoint header is shown in Figure 6.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Hdr Ext Len  |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                                                               |
.                                                               .
.                       IPv6 Tunnel Endpoint                    .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

[NOTE: the format of TEEH may change in the next version of the document to include other information such as the scope for instance]

Figure 6: Tunnel Endpoint Extension Header

The description of the fields is as follows:

- Next Header (8-bit): Identifies the type of header immediately following the TEEH header.
- Hdr Ext Len (8-bit, unsigned integer): Length of the Tunnel Endpoint header in 8-octet units, not including the first 8 octets.
- IPv6 Tunnel Endpoint: Encloses an IPv6 address that should be used as source of the encapsulated IPv4-in-IPv6 response. This field must be padded to ensure that the TEEH length is a multiple of 8 octets.

When TEEH is included in a received IPv4-in-IPv6 datagram, the answer SHOULD be sent to the IPv6 address conveyed in the TEEH.

When TEEH is inserted by a AFTR in an IPv4-in-IPv6 datagram sent to a customer’s device, the IPv6 address included in the TEEH SHOULD be used as destination IPv6 address of subsequent IPv4-in-IPv6 messages.

4. AFTR Bypass Procedure

4.1. Overview

Each CPE (which embeds a B4 function) is notified of the IPv6 reachability information of (one of) the available DS-Lite AFTRs (e.g., using [I-D.ietf-softwire-ds-lite-tunnel-option]). In addition, the CPE must support at least one encapsulation scheme to convey privately-addressed IPv4 traffic into IPv6 datagrams. The CPE behaves as defined in [I-D.ietf-softwire-dual-stack-lite].

A dedicated IPv6 prefix (pref6_aftr) is used to convey the traffic between AFTR nodes.

The following configuration tasks should be undertaken:

o Each AFTR is provided with an IPv4 address pool (IPv4@) for its NAT operations;

o An IPv4-Converted IPv6 prefix [I-D.ietf-behave-address-format] is also assigned to each AFTR. This IPv6 prefix embeds the IPv4 net: pref6_aftr+IPv4@.

o This IPv6 prefix is injected in a routing protocol (IGP/MP-BGP/ i-BGP, or softwire full mesh is used between AFTRs). This route announcement is assumed to be performed by the AFTR itself or by the router which is responsible for redirecting the traffic to a AFTR. When pref6_aftr+IPv4@ is found on routing table, it is used as a "hint" to detect that the IPv4 address is provisioned on a AFTR device.

An operational mode to bypass an AFTR is described in Section 4.2.

4.2. Operational Mode

IPv4-in-IPv6 encapsulated datagrams issued by a CPE are received by an AFTR device (Step 1). This AFTR de-capsulates the datagram and retrieves the destination IPv4 address. Then, it proceeds to a route lookup to check whether a route towards "pref6_aftr+destination IPv4@" is installed. If not, it proceeds with traditional NAT operations. Otherwise (i.e., a route is found. This means that the destination is located behind an AFTR), no NAT44 state is instantiated by the Source AFTR. The datagram is then encapsulated in IPv6 datagram with an IPv6 destination address equal to "pref6_aftr+destination IPv4 @::x" (refer to [I-D.ietf-behave-address-format] for more information on how to build IPv4-Converted IPv6 addresses).

As for the source IPv6 address of the encapsulated datagram, two schemes may be envisaged:

(1) Maintain the same source IPv6 address as per the datagram received from the customer’s device. The deployment of this alternative requires the activation of security association to secure the exchange between the Source and Target AFTR. A trust relationship must be configured.

(2) A new extension header (called TEEH for Tunnel Endpoint Extension Header, defined in Figure 6) is inserted to indicate where to send the response back. The value of the extension header is an IPv6 address of the source CPE (as stored in the Source AFTR).

The datagram is forwarded to the next hop until being delivered to a Target AFTR (Step 2).

- If a NAT entry is instantiated on that AFTR, the datagram is processed. Additionally, the source IPv6 address of the received datagram or the content of the TEEH is stored by the AFTR. This information will be used to send back the response.

In addition to re-writing destination IPv4 address+port (i.e., DNAPT for Destination NAPT), the IPv4 source address and the port number are also modified (referred to as SNAPT for Source NAPT). The translation of the source IPv4 address is required to avoid overlapping private IPv4 addressing in the destination home realm. A public IPv4 address belonging to the Target AFTR pool is used to enforce SNAPT. This SNAPT operation does not alter the number of sessions that may be maintained by a given AFTR.

The resulting IPv4 datagram is then encapsulated in IPv6 and forwarded to its final destination (i.e., B in Figure 7) (Step 3).

An AFTR must be configured to accept TEEH only when it is issued by other AFTR devices. A filtering rule based on the source IPv6 address MAY be configured.

- Otherwise, the datagram is rejected/dropped/silently discarded.

Figure 7 illustrates the occurred flow exchanges.

```
   +------------------+       +------------------+        +---------------------+
   |Src=@IPv6_CPE_A   |       |Src=@IPv6_aftr_s |        | Src=@IPv6_dslite2 |
   |Dst=@IPv6_dslite1|       |Dst=@IPv6_1.2.3.4|        | Dst=@IPv6_CPE_B    |
   |                  |       |TEEH=@IPv6_CPE_A |        |                    |
   | +-------------+ |       | +-------------+ |        | +---------------+ |
   | |Src=10.1.1.1 | |       | |Src=10.1.1.1 | |        | |Src=1.2.3.95   | |
   | |Dst=1.2.3.4  | |       | |Dst=1.2.3.4  | |        | |Dst=192.168.1.1| |
   | +-------------+ |       | +-------------+ |        | +---------------+ |
   +------------------+       +------------------+        +---------------------+
            |                         |                          |
+-+         v           +-----+       v         +-----+          v          +-+
|A|====IPv4-in-IPv6==>|AFTR1|====IPv4-inIPv6==>|AFTR2|====IPv4-in-IPv6===>|B|
+-+        (1)          +-----+      (2)        +-----+         (3)         +-+
                           ^                        ^
                           |                        |
                     +---------------+    +---------------------------------+
                     | No NAT state |    |           NAT state             |
                     +---------------+    |DNAPT: 10.1.1.1/pa:1.2.3.95/pb  |
                                          |SNAPT: 192.186.1.1/pc:1.2.3.4/pd|
                                          +---------------------------------+
```

pa, pb, pc and pd are port numbers. Only an excerpt of the NAT table is shown, IPv6 addresses are also maintained in the NAT table.

Figure 7: Outbound traffic

As for the response, B encapsulates IPv4 traffic in IPv6 datagrams that are forwarded to the AFTR as illustrated in Figure 8 and Figure 9 (Step 4). The AFTR then proceeds to NAT operations (both DNAPT and SNAPT). The resulting IPv4 traffic is then encapsulated in IPv6 and corresponding IPv6 datagrams are then forwarded to the IPv6 address of the remote destination as maintained in the NAT tables (Step 5). TEEH may be inserted to indicate the destination IPv6 address to be used for the subsequent messages (see Figure 8). Figure 9 shows the exchanged flows when TEEH is not used.

```
                  +------------------+                 +-------------------+
                  | Src=@IPv6_aftr2_s|                 | Src=@IPv6_CPE_B   |
                  | Dst=@IPv6_CPE_A  |                 | Dst=@IPv6_dslite2 |
                  |TEEH=@IPv6_aftr2_s|                 |                   |
                  | +-------------+  |                 | +---------------+ |
                  | |Src=1.2.3.4  |  |                 | |Src=192.168.1.1| |
                  | |Dst=10.1.1.1 |  |                 | |Dst=1.2.3.95   | |
                  | +-------------+  |                 | +---------------+ |
                  +------------------+                 +-------------------+
                          |                                      |
+-+                       v                      +-----+         v         +-+
|A|<===================IPv4-inIPv6===============|AFTR2|<===IPv4-in-IPv6====|B|
+-+                       (5)                    +-----+        (4)        +-+
```

Figure 8: Incoming traffic with Option Header

```
                  +-----------------+                 +-------------------+
                  |Src=@IPv6_aftr2_s|                 | Src=@IPv6_CPE_B   |
                  |Dst=@IPv6_CPE_A  |                 | Dst=@IPv6_dslite2 |
                  |                 |                 |                   |
                  | +-------------+ |                 | +---------------+ |
                  | |Src=1.2.3.4  | |                 | |Src=192.168.1.1| |
                  | |Dst=10.1.1.1 | |                 | |Dst=1.2.3.95   | |
                  | +-------------+ |                 | +---------------+ |
                  +-----------------+                 +-------------------+
                          |                                      |
+-+                       v                      +-----+         v         +-+
|A|<===================IPv4-inIPv6===============|AFTR2|<===IPv4-in-IPv6====|B|
+-+                       (5)                    +-----+        (4)        +-+
```

Figure 9: Incoming traffic without Option Header

For the remaining exchanges, either A uses the IPv6 address of AFTR2 to send subsequent messages owing to the presence of TEEH option (see Figure 8. The experienced behaviour is illustrated in Figure 10) or it uses the default behavior and it sends all IPv4 traffic to its attached AFTR1 (as illustrated in Figure 7).

A CPE must be configured to accept incoming IPv4-in-IPv6 traffic with a source address belonging to an IPv6 prefix used to address AFTR devices.

```
                 +-----------------+                     +--------------------+
                 |Src=@IPv6_CPE_A  |                     | Src=@IPv6_dslite2 |
                 |Dst=@IPv6_dslite2|                     | Dst=@IPv6_CPE_B   |
                 |                 |                     |                    |
                 | +-------------+ |                     | +---------------+ |
                 | |Src=10.1.1.1 | |                     | |Src=1.2.3.95   | |
                 | |Dst=1.2.3.4  | |                     | |Dst=192.168.1.1| |
                 | +-------------+ |                     | +---------------+ |
                 +-----------------+                     +--------------------+
                          |                                       |
+-+                       v                   +-----+             v         +-+
|A|====================IPv4-inIPv6==============>|AFTR2|====IPv4-in-IPv6===>|B|
+-+                      (6)                  +-----+            (7)        +-+
```

Figure 10: Withdraw Source CGN

As a result, NAT operations are enforced in one AFTR instead of two nodes. One AFTR is withdrawn from the path.

5. Flow Label Based Alternative

This alternative aims at avoiding two NAT operations without withdrawing an AFTR from the path and without adding a new IPv6 extension header.

Outbound flow exchanges are illustrated in Figure 11. Inbound flow exchanges are shown in Figure 12.

IPv6 is used to convey traffic between AFTR nodes. IPv4-Converted IPv6 addresses are used to detect whether the destination is also managed by an AFTR. No NAT state is then instantiated in the Source AFTR. Two AFTRs are maintained in the path but only one AFTR maintains a NAT state.

AFTR assigns a sequence number (or index) for every softwire between the AFTR and CPE. Sequence numbers must be generated by an AFTR to uniquely identify a given softwire.

The source AFTR sends the sequence number filled in flow label field of the IPv6 header to the target AFTR for indicting where to send the response back.

```
   +-----------------+       +-----------------+        +--------------------+
   |Src=@IPv6_CPE_A  |       |Src=@IPv6_aftr1_s|        | Src=@IPv6_dslite2  |
   |Dst=@IPv6_dslite1|       |Dst=@IPv6_1.2.3.4|        | Dst=@IPv6_CPE_B    |
   |                 |       |Flow Lab= a      |        |                    |
   | +-------------+ |       | +-------------+ |        | +----------------+ |
   | |Src=10.1.1.1 | |       | |Src=10.1.1.1 | |        | |Src=1.2.3.95    | |
   | |Dst=1.2.3.4  | |       | |Dst=1.2.3.4  | |        | |Dst=192.168.1.1| |
   | +-------------+ |       | +-------------+ |        | +----------------+ |
   +-----------------+       +-----------------+        +--------------------+
            |                         |                          |
+-+         v          +-----+        v          +-----+         v          +-+
|A|====IPv4-in-IPv6==>|AFTR1|====IPv4-inIPv6==>|AFTR2|====IPv4-in-IPv6===>|B|
+-+        (1)         +-----+       (2)         +-----+        (3)         +-+
                          ^                         ^
                          |                         |
                    +---------------+    +---------------------------------+
                    | No NAT state |    |            NAT state             |
                    +---------------+    |DNAPT: 10.1.1.1/pa:1.2.3.95/pb  |
                                         |SNAPT: 192.186.1.1/pc:1.2.3.4/pd|
                                         |Flow label: a                    |
                                         +---------------------------------+
```

Figure 11: Outbound traffic

These steps are followed:

o Step 1: A encapsulates its IPv4 datagram in IPv6 one and forwards the encapsulated IPv4-in-IPv6 datagram to its outbound AFTR.

o Step 2: Once that datagram is received by AFTR1, it de- capsulates it and retrieves the IPv4 datagram. Moreover, the destination IPv4 address is returned. AFTR1 proceeds to a routing look up to check whether a route to pref6_aftr+destination IPv4@ is installed. If the answer is positive (i.e., the destination is managed by an AFTR), AFTR1 does not proceed to any NAT44 operation. The IPv4 datagram is then encapsulated in an IPv6 one and forwarded to AFTR2 (destination IPv6 address of the encapsulated datagram is pref6_aftr+IPv4@). The sequence number a of softwire between AFTR1 and A is filled in the Flow Label field of the IPv6 packet.

o Step 3: AFTR2 receives that datagram. It de-capsulates the received datagram and retrieves the enclosed IPv4 one. AFTR2 checks if a NAT state is already instantiated towards the destination IPv4 address/port number. If the answer is positive, then it proceeds to DNAPT and SNAPT. AFTR2 keeps the sequence number a in the NAT table. The resulting datagram is then forwarded to the IPv6 address of B (stored in AFTR2).

- o Step 4: B replies as per DS-Lite specification. o

- o Step 5: AFTR2 de-capsulates the received datagram and proceeds to DNAPT and SNAPT. The resulting IPv4 datagram is then encapsulated in an IPv6 one and the sequence number a is filled in the Flow Label field. The IPv6 packet is forwarded to AFTR1. o

- o Step 6: AFTR1 finds the softwire according sequence number a carried in the Flow Label field, then it forwards the packet to A.

```
   +-----------------+      +-----------------+       +--------------------+
   |Src=@IPv6_CPE_A  |      |Src=@IPv6_aftr2_s|       | Src=@IPv6_CPE_B    |
   |Dst=@IPv6_dslite1|      |Dst=@IPv6_aftr1_s|       | Dst=@IPv6_dslite2  |
   |                 |      |Flow Lab= a      |       |                    |
   | +-------------+ |      | +-------------+ |       | +----------------+ |
   | |Src=1.2.3.4  | |      | |Src=1.2.3.4  | |       | |Src=192.168.1.1| |
   | |Dst=10.1.1.1 | |      | |Dst=10.1.1.1 | |       | |Dst=1.2.3.95   | |
   | +-------------+ |      | +-------------+ |       | +----------------+ |
   +-----------------+      +-----------------+       +--------------------+
            |                         |                         |
+-+         v         +-----+         v         +-----+         v         +-+
|A|<===IPv4-in-IPv6==|AFTR1|<===IPv4-inIPv6====|AFTR2|<===IPv4-in-IPv6====|B|
+-+        (6)        +-----+        (5)        +-----+        (4)        +-+
                          ^                         ^
                          |                         |
                   +---------------+    +---------------------------------+
                   | No NAT state |    |         NAT state               |
                   +---------------+    |DNAPT: 10.1.1.1/pa:1.2.3.95/pb  |
                                        |SNAPT: 192.186.1.1/pc:1.2.3.4/pd|
                                        |Flow label: a                   |
                                        +---------------------------------+
```

Figure 12: Inbound traffic

6. IANA Considerations

TBC.

7. Security Considerations

B4 element MUST be configured to accept incoming IPv4-in-IPv6 datagrams not issued by its outbound AFTR. All deployed AFTRs SHOULD share a security association to secure the use of the TEEH option.

8. Acknowledgements

The author would like to thank P. Levis, M. Kassi Lahlou, E. Burgey and D. Binet for their feedback and comments.

9. References

9.1. Normative References

[I-D.ietf-behave-address-format]
Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", draft-ietf-behave-address-format-10 (work in progress), August 2010.

[I-D.ietf-softwire-dual-stack-lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", draft-ietf-softwire-dual-stack-lite-06 (work in progress), August 2010.

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

9.2. Informative References

[I-D.boucadair-dispatch-ipv6-atypes]
Boucadair, M., Noisette, Y., and A. Allen, "The atypes media feature tag for Session Initiation Protocol (SIP)", draft-boucadair-dispatch-ipv6-atypes-00 (work in progress), July 2009.

[I-D.boucadair-mmusic-altc]
Boucadair, M., Kaplan, H., Gilman, R., and S. Veikkolainen, "Session Description Protocol (SDP) Alternate Connectivity (ALTC) Attribute", draft-boucadair-mmusic-altc-01 (work in progress), September 2010.

[I-D.carpenter-behave-referral-object]
Carpenter, B., Boucadair, M., Halpern, J., Jiang, S., and K. Moore, "A Generic Referral Object for Internet Entities", draft-carpenter-behave-referral-object-01 (work in progress), October 2009.

[I-D.ietf-softwire-ds-lite-tunnel-option]
Hankins, D. and T. Mrugalski, "Dynamic Host Configuration

Protocol for IPv6 (DHCPv6) Option for Dual- Stack Lite", draft-ietf-softwire-ds-lite-tunnel-option-05 (work in progress), September 2010.

Appendix A. Alternative Solution

This alternative aims at avoiding two NAT operations without withdrawing a AFTR from the path.

Outbound flow exchanges are illustrated in Figure 13. Inbound flow exchanges are shown in Figure 14.

IPv6 is used to convey traffic between AFTR nodes. IPv4-Converted IPv6 addresses are used to detect whether the destination is also managed by an AFTR. No NAT state is then instantiated in the Source AFTR. Two AFTR are maintained in the path but only one AFTR maintains a NAT state.

```
   +-----------------+      +-----------------+       +-------------------+
   |Src=@IPv6_CPE_A  |      |Src=@IPv6_aftr1_s|       | Src=@IPv6_dslite2 |
   |Dst=@IPv6_dslite1|      |Dst=@IPv6_1.2.3.4|       | Dst=@IPv6_CPE_B   |
   |                 |      |                 |       |                   |
   | +-------------+ |      | +-------------+ |       | +---------------+ |
   | |Src=10.1.1.1 | |      | |Src=10.1.1.1 | |       | |Src=1.2.3.95   | |
   | |Dst=1.2.3.4  | |      | |Dst=1.2.3.4  | |       | |Dst=192.168.1.1| |
   | +-------------+ |      | +-------------+ |       | +---------------+ |
   +-----------------+      +-----------------+       +-------------------+
            |                         |                        |
+-+         v         +-----+         v        +-----+         v          +-+
|A|====IPv4-in-IPv6==>|AFTR1|====IPv4-inIPv6==>|AFTR2|====IPv4-in-IPv6===>|B|
+-+        (1)        +-----+        (2)       +-----+        (3)         +-+
                          ^                        ^
                          |                        |
                   +--------------+    +--------------------------------+
                   | No NAT state |    |           NAT state            |
                   +--------------+    |DNAPT: 10.1.1.1/pa:1.2.3.95/pb  |
                                       |SNAPT: 192.186.1.1/pc:1.2.3.4/pd|
                                       +--------------------------------+
```

Figure 13: Outbound traffic

The following steps are followed

o Step 1: A encapsulates it IPv4 datagram in IPv6 one and forwards the encapsulated IPv4-in-IPv6 datagram to its outbound AFTR. The

IPv6 address/FQDN of its outbound AFTR is provisioned using DHCP for instance.

- Step 2: Once that datagram is received by the AFTR1, its de-capsulates it and retrieves the IPv4 datagram. Moreover, the destination IPv4 address is returned. AFTR1 proceeds to a routing look up to check whether a route to pref6_aftr+destination IPv4@ is installed. If the answer is positive (i.e., the destination is managed by an AFTR), AFTR1 does not proceeds to any NAT44 operation. The IPv4 datagram is then encapsulated in an IPv6 ones and forwarded to AFTR2 (destination IPv6 address of the encapsulated datagram is pref6_aftr+IPv4@). The source IPv6 address used by AFTR1 must identify unambiguously A.

- Step 3: AFTR2 receives that datagrams. It de-capsulates the received datagram and retrieves the enclosed IPv4 one. AFTR2 checks if a NAT state is already instantiated towards the destination IPv4 address/port number. If the answer is positive, then it proceeds to DNAPT and SNAPT. The resulting datagram is then forwards to the IPv6 address of B (stored in AFTR2).

- Step 4: B replies as per DS-Lite specifications.

- Step 5: AFTR2 de-capsulates the received datagrams and proceeds to DNAPT and SNAPT. The resulting IPv4 datagram is then encapsulated in an IPv6 one and forwarded to AFTR1.

- Step 6: AFTR1 checks its swapping states and forwards the packet to A.

```
   +------------------+      +------------------+       +--------------------+
   |Src=@IPv6_CPE_A   |      |Src=@IPv6_aftr2_s|       | Src=@IPv6_CPE_B    |
   |Dst=@IPv6_dslite1|      |Dst=@IPv6_aftr1_s|       | Dst=@IPv6_dslite2 |
   |                  |      |                  |       |                    |
   | +-------------+ |      | +-------------+ |       | +---------------+ |
   | |Src=1.2.3.4  | |      | |Src=1.2.3.4  | |       | |Src=192.168.1.1| |
   | |Dst=10.1.1.1 | |      | |Dst=10.1.1.1 | |       | |Dst=1.2.3.95   | |
   | +-------------+ |      | +-------------+ |       | +---------------+ |
   +------------------+      +------------------+       +--------------------+
            |                          |                         |
+-+         v          +-----+         v          +-----+        v          +-+
|A|<===IPv4-in-IPv6==|AFTR1|<===IPv4-inIPv6====|AFTR2|<===IPv4-in-IPv6====|B|
+-+        (6)         +-----+        (5)          +-----+       (4)         +-+
                            ^                          ^
                            |                          |
                  +---------------+    +----------------------------------+
                  | No NAT state |    |          NAT state                |
                  +---------------+    |DNAPT: 10.1.1.1/pa:1.2.3.95/pb   |
                                       |SNAPT: 192.186.1.1/pc:1.2.3.4/pd|
                                       +----------------------------------+
```

Figure 14: Inbound traffic

Authors' Addresses

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange-ftgroup.com

Christian Jacquenet
France Telecom
Rennes 35000
France

Email: christian.jacquenet@orange-ftgroup.com

Jun Song
ZTE Corporation
No.68,Zijinghua Road, Yuhuatai District
Nanjing, Jiangsu Province
China

Email: song.jun@zte.com.cn

Qibo Niu
ZTE Corporation
No.68,Zijinghua Road, Yuhuatai District
Nanjing, Jiangsu Province
China

Email: niu.qibo@zte.com.cn