Network Working Group   P. Thubert
Internet-Draft   M. Molteni
Expires: December 18, 2002   Cisco Systems
                                                           June 19, 2002

IPv6 Reverse Routing Header and its application to Mobile Networks

draft-thubert-nemo-reverse-routing-header-00

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on December 18, 2002.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

Abstract

Already existing proposals enable Mobile Networks by extending Mobile IP to support Mobile Routers. In order to enable nested Mobile Networks, some involve the overhead of nested tunnels between the Mobile Routers and their Home Agents.

This proposal allows the building of a nested Mobile Network avoiding the nested tunnel overhead. This is accomplished by using a new routing header, called the reverse routing header, and by overlaying a layer 3 tree topology on the evolving Mobile Network.

Table of Contents

1. Introduction

This document assumes the reader is familiar with the Mobile Networks terminology defined in [2] and with Mobile IPv6 defined in [1].

Generally a NEMO may be either simple (a network with one mobile router) or nested, single or multi-homed. This proposal starts from the assumption that nested NEMOs will be the norm, and so presents a solution that avoids the tunnel within tunnel overhead of already existing proposals.

The solution is based on a single bi-directional tunnel between the first Mobile Router (MR) to forward a packet and its Home Agent (HA). By using IPsec ESP on that tunnel, home equivalent privacy is obtained without further encapsulation.

The solution uses a new Routing Header (RH), called the Reverse Routing Header (RRH), to provide an optimized path for the single tunnel. RRH is a variant of IPv4 Loose Source and Record Route (LSRR) [6] adapted for IPv6. RRH records the route out of the nested NEMO and can be trivially converted into a routing header for packets destined to the NEMO.

This version focuses on single-homed NEMOs. Hints for further optimizations and multi-homing are given in the appendixes.

Local Fixed Node (LFN) and Correspondent Node (CN) operations are left unchanged as in [1]. Specifically the CN can also be a LFN.

Section 3 proposes an example to illustrate the operation of the proposed solution, leaving detailed specifications to the remaining chapters. The reader may refer to Section 2.1 for the specific terminology.

1.1 Extending existing solutions

This proposal extends [1] to support simple and nested Mobile Networks.

This paper also builds on an other existing proposal, [3], which is based on nested tunnels, in order to address the following problems, introduced by that solution:

"Pinball" routing

Both inbound and outbound packets will flow via the HAs of all the MRs on their path within the NEMO, with increased latency, less resilience and more bandwidth usage.

Packet size

An extra IPv6 header is added per level of nesting to all the packets. The header compression suggested in [5] cannot be applied because both the source and destination (the intermediate MR and its HA), are different hop to hop.

2. Terminology and Assumptions

2.1 Terminology

Simple NEMO

One or more IP subnets attached to a MR and mobile as a unit, with respect to the rest of the Internet. A simple NEMO can be either single or multi-homed.

The IP subnets may have any kind of topology and may contain fixed routers. All the access points of the NEMO (to which further MRs may attach) are on the same layer 2 link of the MR.

We like to represent a simple single-homed NEMO as an hanger, because it has only one uplink hook and a bar to which multiple hooks can be attached. Graphically we use the question mark "?" to show the uplink hook (interface) connected to the MR, and the "=" sign to represent the bar:

```
                                   ?
                                  MR1
                                   |
                            ===============
```

Nested NEMO

A group of simple NEMOs recursively attached together and implementing nested Mobility as defined in [2].

```
                                    ?
                                   MR1
                                    |
                        ====?===============?====
                           MR2             MR3
                            |               |
                      ===========   ===?==========?===
                                      MR4         MR5
                                       |           |
                                  ==========  ============
```


IPv6 Mobile Host

A IPv6 Host, with support for MIPv6 MN, and the additional Nemo capability described in this draft.

Home prefix

Network prefix, which identifies the home link within the Internet topology.

NEMO prefix

Network prefix, common to all IP addresses in the NEMO when the MR is attached to the home link. It may or may not be a subset of the Home subnet prefix.

Inbound direction:

direction from outside the NEMO to inside

Outbound direction:

direction from inside the NEMO to outside

2.2 Assumptions

We make the following assumptions:

1. A MR has one Home Agent and one Home Address -> one primary CoA.

2. A MR attaches to a single Access Router as default router.

3. A MR may have more than one uplink interface.

4. An interface can be either wired or wireless. The text assumes that interfaces are wireless for generality.

5. Each simple NEMO may have more that one L2 Access Point, all of them controlled by the same Access Router, which we assume to be the Mobile Router.

Since an MR has only one primary CoA, only one uplink interface can be used at a given point of time. Since the MR attaches to a single access router, if due care is applied to avoid loops, then the resulting topology is a tree.

3. An Example

The nested NEMO in the following figure has a tree topology, according to the assumptions in Section 2.2. In the tree each node is a simple NEMO, represented by its MR.

```
                   +----------------------+
                   |     Internet         |---CN
                   +----------------|-----+
                     /         Access Router
               MR3_HA               |
                          ======?======
                               MR1
                                |
```

```
                         ====?=============?==============?===
                            MR5           MR2            MR6
                             |             |              |
                       ===========   ===?=========   =============
                                        MR3
                                         |
                                  ==|=========?==   <-- NEMO3
                                   LFN1      MR4
                                              |
                                          =========
```

An example nested NEMO

This example focuses on a NEMO node at depth 3 (NEMO3) inside the tree, represented by its mobile router MR3. The path to the Top Level Mobile Router (TLMR) MR1 and then the Internet is

MR3 -> MR2 -> MR1 -> Internet

Consider the case where a LFN belonging to NEMO3 sends a packet to a CN in the Internet, and the CN replies back.

With the tunnel within tunnel approach described by [3], we would have three bi-directional nested tunnels:

```
                                 -----------.
                       --------/          /-----------.
              -------/        |          |          /-----------
    CN ------( -  - | -  -  - |  -  -  - | -  -  -  |  -  -  -  (-------- LFN
       MR3_HA -------\        |          |          \----------- MR3
                MR2_HA --------\          \----------- MR2
                          MR1_HA ----------- MR1
```


Depending on the relative location of MR1_HA, MR2_HA and MR3_HA, this may lead to a very inefficient "pinball" routing.

On the other hand, with the RRH approach we would have only one bi-directional tunnel:

```
          --------------------------------- MR1 ---- MR2 ----
CN ------(  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  (-------- LFN
   MR3_HA --------------------------------- MR1 ---- MR2 ---- MR3
```

The first mobile router on the path, MR3, in addition to tunneling the packet to its HA, adds a reverse routing header with N = 3 pre-allocated slots. Choosing the right value for N is discussed in Section 6.2. The bottom slot is equivalent to the MIPv6 Home Address option. MR3 inserts its home address MR3_HAddr into slot 0.

The outer packet has source MR3's CareOf Address (CoA), MR3_CoA and destination MR3's Home Agent, MR3_HA:

```
<--------------- outer IPv6 header --------------------->
+-------+-------++ -- ++----+-------+-------+---------+ +-------
|oSRC   |oDST   |:    :|oRRH| slot2 | slot1 | slot0   | |
|MR3_CoA|MR3_HA |:oEXT:|type|       |       |MR3_HAddr| |iPACKET
|       |       |:    :| 4  |       |       |         | |
+-------+-------++ -- ++----+-------+-------+---------+ +-------
```

The second router on the path, MR2, notices that the packet already contains an RRH, and so it overwrites the source address of the packet with its own address, MR2_CoA, putting the old source address, MR3_CoA, in the first free slot of the RRH.

The outer packet now has source MR2_CoA and destination MR3_HA; the RRH from top to bottom is MR3_CoA | MR3_HAddr:

```
<--------------- outer IPv6 header --------------------->
+-------+-------++ -- ++----+-------+-------+---------+ +-------
|oSRC   |oDST   |:    :|oRRH| slot2 | slot1 | slot0   | |
|MR2_CoA|MR3_HA |:oEXT:|type|       |MR3_CoA|MR3_HAddr| |iPACKET
|       |       |:    :| 4  |       |       |         | |
+-------+-------++ -- ++----+-------+-------+---------+ +-------
```

In general the process followed by the second router is repeated by all the routers on the path, including the TLMR (in this example MR1). When the packet leaves MR1 the source address is MR1_CoA and the RRH is MR2_CoA | MR3_CoA | MR3_HAddr:

```
<-------------- outer IPv6 header -------------------->
+-------+-------++ -- ++----+-------+-------+---------+ +--------
|oSRC   |oDST   |:    :|oRRH| slot2 | slot1 | slot0   | |
|MR1_CoA|MR3_HA |:oEXT:|type|MR2_CoA|MR3_CoA|MR3_HAddr| |iPACKET
|       |       |:    :| 4  |       |       |         | |
+-------+-------++ -- ++----+-------+-------+---------+ +--------
```

In a colloquial way we may say that while the packet travels from MR3 to MR3_HA, the NEMO tunnel end point "telescopes" from MR3 to MR2 to MR1.

When the home agent MR3_HA receives the packet it notices that it contains a RRH and it looks at the bottom entry, MR3_HAddr. This entry is used as if it were a MIPv6 Home Address destination option, i.e. as an index into the Binding Cache. When decapsulating the inner packet the home agent performs the checks described in Section 8, and if successful it forwards the inner packet to CN.

MR3_HA stores two items in the Bind Cache Entry associated with MR3: the address entries from RRH, to be used to build the RH, and the packet source address MR1_CoA, to be used as the first hop.

Further packets from the CN to the LFN are plain fixed IPv6 packets. Destination is LFN, and so the packet reaches MR3's home network.

MR3_HA intercepts it, does a Bind Cache prefix lookup and obtains as match the MR3 entry, containing the first hop and the information required to build the RH. It then puts the packet in the tunnel MR3_HA -- MR3 as follows: source address MR3_HA and destination address the first hop, MR1_CoA. The RH is trivially built out of the previous RRH: MR2_CoA | MR3_CoA | MR3_HAddr:

```
<-------------- outer IPv6 header -------------------->
+-------+-------++ -- ++----+-------+-------+---------+ +--------
|oSRC   |oDST   |:    :|oRH |       |       |         | |
|MR3_HA |MR1_CoA|:oEXT:|type|MR2_CoA|MR3_CoA|MR3_HAddr| |iPACKET
|       |       |:    :| 2  |       |       |         | |
+-------+-------++ -- ++----+-------+-------+---------+ +--------
```

The packet is routed with plain IP routing up to the first destination MR1_CoA.

The RH of the outer packet is type 2 as in [1], but has additional semantics inherited from type 0: it contains the path information to traverse the nested NEMO from the TLMR to the tunnel endpoint MR. Each intermediate destination forwards the packet to the following destination in the routing header. The security aspects of this are treated in Section 11.2.

MR1, which is the initial destination in the IP header, looks at the RH and processes it according to Section 9, updating the RH and the destination and sending it to MR2_CoA. MR2 does the same and so on until the packet reaches the tunnel endpoint, MR3.

When the packet reaches MR3, the source address in the IP header is MR3_HA, the destination is MR3_CoA and in the RH there is one segment left, MR3_HAddr. As a consequence the packet belongs to the MR3_HA -- MR3 tunnel. MR3 decapsulates the inner packet, applying the rules described in Section 9 and sends it to LFN. The packet that reaches LFN is the plain IPv6 packet that was sent by CN.

4. New Routing Headers

This draft modifies the MIPv6 Routing Header type 2 and introduces two new Routing Headers, type 3 and 4. Type 3 will be discussed in Appendix A.3.1. The draft presents their operation in the context of Mobile Routers although the formats are not tied to MIP and could be used in other situations.

4.1 Routing Header Type 2 (MIPv6 RH with extended semantics)

Mobile IPv6 uses a Routing header to carry the Home Address for packets sent from a Correspondent Node to a Mobile Node. In [1], this Routing header (Type 2) is restricted to carry only one IPv6 address. The format proposed here extends the Routing Header type 2 to be multi-hop.

The processing of the multi-hop RH type 2 inherits from the RH type 0 described in [10]. Specifically: the restriction on multicast addresses is the same; a RH type 2 is not examined or processed until it reaches the node identified in the Destination Address field of the IPv6 header; in that node, the RH type 0 algorithm applies, with

added security checks.

The construction of the multi-hop RH type 2 by the HA is described in Section 8; the processing by the MRs is described in Section 9.5; and the security aspects are treated in Section 11.2.

The multi-hop Routing Header type 2, as extended by this draft, has the following format:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  Next Header  |  Hdr Ext Len  | Routing Type=2| Segments Left |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                            Reserved                           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +                                                               +
 |                                                               |
 +                           Address[1]                          +
 |                                                               |
 +                                                               +
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +                                                               +
 |                                                               |
 +                           Address[2]                          +
 |                                                               |
 +                                                               +
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 .                               .                               .
 .                               .                               .
 .                               .                               .
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +                                                               +
 |                                                               |
 +                           Address[n]                          +
 |                                                               |
 +                                                               +
```

```
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Next Header

8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [13].

Hdr Ext Len

8-bit unsigned integer. Length of the Routing header in 8-octet

units, not including the first 8 octets. For the Type 2 Routing header, Hdr Ext Len is equal to two times the number of addresses in the header.

Routing Type

8-bit unsigned integer. Set to 2.

Segments Left

8-bit unsigned integer. Number of route segments remaining, i.e., number of explicitly listed intermediate nodes still to be visited before reaching the final destination.

Reserved

32-bit reserved field. Initialized to zero for transmission; ignored on reception.

Address[1..n]

Vector of 128-bit addresses, numbered 1 to n.

The destination node of a packet containing a RH type 2 can be a MR or some other kind of node. If it is a MR it will perform the algorithm described in Section 9.5, otherwise it will operate as prescribed by IPv6 [10] when the routing type is unrecognized.

4.2 Routing Header Type 4 (The Reverse Routing Header)

The Routing Header type 4, or Reverse Routing Header (RRH), is a variant of IPv4 loose source and record route (LSRR) [6] adapted for IPv6.

Addresses are added from bottom to top (0 to n-1 in the picture). The RRH is designed to help the destination build an RH for the return path.

When a RRH is present in a packet, the rule for upper-layer checksum computing is that the source address used in the pseudo-header is that of the original source, located in the slot 0 of the RRH, unless the RRH slot 0 is empty, in which case the source in the IP header of the packet is used.

As the 'segment left' field of the generic RH is reassigned to the number of segments used, a IPv6 Host that does not support RRH will discard the packet, unless the RRH is empty.

The Type 4 Routing Header has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Next Header  |  Hdr Ext Len  | Routing Type=4| Segments Used |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                            Slot[n-1]                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                               .                               .
.                               .                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
```

```
   |                                                               |
   +                    Slot[1] (1st MR CoA)                       +
   |                                                               |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   |                                                               |
   +                  Slot[0] (Home address)                       +
   |                                                               |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Next Header

8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [13].

Hdr Ext Len

8-bit unsigned integer. Length of the Routing header in 8-octet units, not including the first 8 octets. For the Type 4 Routing header, Hdr Ext Len is equal to two times the number of addresses

in the header.

Routing Type

8-bit unsigned integer. Set to 4.

Segments Used

8-bit unsigned integer. Number of slots used. Initially set to 1 by the MR when only the Home Address is there. Incremented by the MRs on the way as they add the packets source addresses to the RRH.

Sequence Number

32-bit unsigned integer. The Sequence Number starts at 0, and is incremented by the source upon each individual packet. Using the lollipop algorithm, the high order byte is never reset to zero but passes from 255 to 1.

The sequence number is used to check the freshness of the RRH; anti-replay protection is left to IPsec AH.

Slot[n-1..0]

Vector of 128-bit addresses, numbered n-1 to 0.

When applied to the NEMO problem, the RRH can be used to update the HA on the actual location of the MR. Only MRs forwarding packets on an egress interface while not at home update it on the fly.

A RRH is inserted by the first MR on the Nemo outbound path, as part of the reverse tunnel encapsulation; it is removed by the associated HA when the tunneled packet is decapsulated. The RRH contains n pre-allocated address slots, to be filled by each MR in the path.

4.3 Extension Header order

The RH type 2 is to be placed as any RH as described in [10] section 4.1. If a RH type 0 is present in the packet, then the RH type 2 is placed immediately after the RH type 0, and the RH type 0 MUST be consumed before the RH type 2.

RH type 3 and 4 are mutually exclusive. They are to be placed right after the Hop-by-Hop Options header if any, or else right after the IPv6 header.

As a result, the order prescribed in section 4.1 of RFC 2460 becomes:

IPv6 header

Hop-by-Hop Options header

Routing header type 3 or 4

Destination Options header (note 1)

Routing header type 0

Routing header type 2

Fragment header

Authentication header (note 2)

Encapsulating Security Payload header (note 2)

Destination Options header (note 3)

upper-layer header

5. ICMP

The RRH could have fewer slots than the number of MRs in the path because either the nested NEMO topology is changing too quickly or the MR that inserted the RRH could have a wrong representation of the topology.

To solve this problem a new ICMP message is introduced, "RRH Warning", type 64. Note that this ICMP message creates a new class of warning messages besides the error messages and the control messages of ICMP.

This message allows a MR on the path to propose a larger number of slots to the MR that creates the RRH. The Proposed Size MUST be larger than the current size and MUST NOT be larger than 8.

The originating MR must rate-limit the ICMP messages to avoid excessive ICMP traffic in the case of the source failing to operate as requested.

The originating MR must insert an RH type 2 based on the RRH in the associated IP header, in order to route the ICMP message back to the source of the reverse tunnel. A MR that receives this ICMP message is the actual destination and it MUST NOT forward it to the (LFN) source of the tunneled packet.

The type 64 ICMP has the following format:

```
   0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   Type = 64   |     Code = 0  |            Checksum           |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  | Current Size  | Proposed Size |           Reserved            |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |                   As much of invoking packet                  |
  +              as will fit without the ICMPv6 packet            +
  |              exceeding the minimum IPv6 MTU                   |
  .                                                               .
  .                                                               .
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

64 [To Be Assigned]

Code 0: RRH too small

The originating MR requires the source to set the RRH size to a larger value. The packet that triggered the ICMP will still be forwarded by the MR, but the path cannot be totally optimized (see Section 9.3).

Checksum

The ICMP checksum [12].

Current Size

RRH size of the invoking packet, as a reference.

Proposed Size

The new value, expressed as a number of IPv6 addresses that can fit in the RRH.

Reserved

16-bit reserved field. Initialized to zero for transmission; ignored on reception.

6. Modifications to IPv6 Neighbor Discovery

6.1 Modified Router Advertisement Message Format

Mobile IPv6 [1] modifies the format of the Router Advertisement message [11] by the addition of a single flag bit (H) to indicate that the router sending the Advertisement message is serving as a home agent on this link.

This draft adds another single flag bit (R) to indicate that the router sending the advertisement message is a MR. This means that the link on which the message is sent is a NEMO, which may or may not be at home.

The Router Advertisement message has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Cur Hop Limit |M|O|H|N|Reservd|       Router Lifetime         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Reachable Time                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Retrans Timer                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Options ...
+-+-+-+-+-+-+-+-+-+-+-+-+-
```

This format represents the following changes over that originally specified for Neighbor Discovery [11]:

Home Agent (H)

The Home Agent (H) bit is set in a Router Advertisement to indicate that the router sending this Router Advertisement is also functioning as a Mobile IP home agent on this link.

NEMO Capable (N)

The NEMO Capable (N) bit is set in a Router Advertisement to indicate that the router sending this Router Advertisement is also functioning as a Mobile Router on this link, so that the link is a NEMO, possibly away from home.

Reserved

Reduced from a 6-bit field to a 4-bit field to account for the addition of the Home Agent (H) and the Nemo Capable (N) bits.

6.2 New Tree Information Option Format

This draft defines a new Tree Information option, used in Router Advertisement messages.

The Tree Information option has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |  Length = 5   |  Preference   |   TreeDepth   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F|H|        Reserved           |           DelayTime           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                    Tree TLMR Identifier                       +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                         Tree Group                            +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type

Set to 7.

Length

8-bit unsigned integer. The length of the option (including the type and length fields) in units of 8 octets. The value of this field MUST be 5.

Preference

8-bit signed integer. The preference of a tree as configured on

its TLMR. Range from 0 = lowest to 255 = maximum.

TreeDepth

8-bit signed integer. Set to 0 by the TLMR and incremented by 1 by each MR down the tree.

Fixed (F)

1-bit flag. Set to indicate that the TLMR is either attached to a fixed network or at home. This field is set by the TLMR and left unchanged by the other MRs.

Home (H)

1-bit flag. Set to indicate that the TLMR is also functioning as a HA, for re-homing purposes. Set by the TLMR and left unchanged by the other MRs.

Reserved

14-bit reserved field. Initialized to zero for transmission; ignored on reception.

DelayTime

Tree-wide time constant in milliseconds, set by the TLMR and left unchanged by other MRs.

Tree TLMR Identifier

Set by the TLMR and left unchanged by other MRs. Identifier of

the tree. It is a unique IPv6 address.

Tree Group

Group Identifier. Set by the TLMR and left unchanged by other MRs. A MR may use the Tree Group in its tree selection algorithm.

The TLMR MUST include this option in its Router Advertisements.

A MR receiving this option from its Access Router MUST update the TreeDepth field and MUST forward it on its ingress interface(s), as described in Section 9.4.

The alignment requirement of the Tree Information Option is 8n.

7. Binding Cache Management

7.1 Binding Updates

Binding Updates are still used as described in MIPv6 [1] for Home Registration and de-registration, but only when the MR registers for the first time with its HA.

Since the BU doesn't contain the full NEMO path to the MR, it cannot be used in this design of nested NEMOs.

7.2 RRH Heartbeat

Subsequent updates (or just refreshes) to the CoA binding are obtained as one of the results of processing the RRH by the HA.

When the MR becomes aware of a topology change in the tree (for examples it changes point of attachment, it obtains a new CoA, it receives a Tree Information message) or in the absence of traffic (detected by a timeout) to the HA, it must send an RRH Heartbeat (IP packet with the RRH and empty payload).

Security issues are discussed in Section 11.2.

8. Home Agent Operation

This section inherits from chapter 10 [1], which is kept unmodified except for parts 10.5 and 10.6 which are extended. This draft mostly adds the opportunity for an MN to update the Binding Cache of its Home Agent using RRH, though it does not change the fact that MNs still need to select a home agent, register and deregister to it, using the MIP Bind Update.

This draft extends [1] section 10.6 as follows:

- o The entry point of the tunnel is now checked against the TLMR as opposed to the primary CoA.
- o The Binding Cache can be updated based on RRH with proper AH authentication.

As further explained in Section 7.1, this specification modifies MIP so that HA can rely on the RH type 4 (RRH) to update its the Bind Cache Entry (BCE), when the Mobile Node moves. The conceptual content of the BCE is extended to contain a sequence counter, and the sequence of hops within the -potentially nested- Mobile Network to a given Mobile Node. The sequence counter is initially set to 0.

When the HA gets a packet destined to itself, it checks for the presence of a Routing Header of type 3 or 4. Both contain as least the entry for the home address of the MN in slot 0; this replaces the MIP Home Address Option and allows the HA to determine the actual source of the packet, to access the corresponding security association.

As explained in Section 11.2, the HA MUST verify the authenticity of the packet using IPSEC AH and drop packets that were not issued by the proper Mobile Node. An RRH is considered only if the packet is authenticated and if its sequence number is higher than the one saved in the BCE. Also, an RRH is considered only if an initial Bind Update exchange has been successfully completed between the Mobile Node and its Home Agent for Home Registration. If the RRH is valid, then the Bind Cache Entry is revalidated for a lifetime as configured from the initial Bind Update.

The BCE abstract data is updated as follows:

The first hop for the return path is the last hop on the path of the incoming packet, that is between the HA and the Top Level Mobile Router (TLMR) of the Mobile Network. The HA saves the IP address of the TLMR from the source field in the IP header.

The rest of the path to the MN is found in the RRH.

The sequence counter semantics is changed as described in Section 4.2

This draft extends [1] section 10.5 as follows:

A Home Agent advertises the prefixes of its registered Mobile Routers, during the registration period, on the local Interior Gateway Protocol (IGP).

The Routing Header type 2 is extended to be multi-hop.

The Home Agent is extended to support routes to prefixes that are owned by Mobile Routers. This can be configured statically, or can be exchanged using a routing protocol as in [3], which is out of the scope of this document. As a consequence of this process, the Home Agent which is selected by a Mobile Router advertises reachability of the MR prefixes for the duration of the registration over the local IGP.

When a HA gets a packet for which the destination is a node behind a Mobile Router, it places the packet in the tunnel to the associated

MR. This ends up with a packet which destination address in the IP Header is the TLMR, and with a Routing Header of type 2 for the rest of the way to the Mobile Router, which may be multi-hop.

To build the RH type 2 from the RRH, the HA sets the type to 2, and clears the bits 32-63 (byte 4 to 7).

9. Mobile Router Operation

This section inherits from chapter 11 of [1], which is extended to support Mobile Networks and Mobile Routers as a specific case of

Mobile Node.

This draft extends section 11.2.1 of [1] as follows:

- o When not at home, an MR uses a reverse tunnel with its HA for all the traffic that is sourced in its mobile network(s); traffic originated further down a nested network is not tunneled twice but for exception cases.

- o The full path to and within the Mobile Network is piggy-backed with the traffic on a per-packet basis to cope with rapid movement. This makes the packet construction different from MIPv6.

The MR when not at home sets up a bi-directional tunnel with its HA. The reverse direction MR -> HA is needed to assure transparent topological correctness to LFNs, as in [3]. But, as opposed to that solution, nested tunnels are generally avoided.

9.1 Processing of ICMP "RRH too small"

The New ICMP message "RRH too Small" is presented in Section 5. This message is addressed to the MR which performs the tunnel encapsulation and generates the RRH.

Hence, a MR that receives the ICMP "RRH too small" MUST NOT propagate it to the originating LFN or inner tunnel source, but MUST process it for itself.

If the Current Size in the ICMP messages matches the actual current number of slots in RRH, and if the ICMP passes some safety checks as described in Section 5, then the MR MAY adapt the number of slots to the Proposed Size.

9.2 Processing of ICMP error

ICMP back {

```
      if RRH is present {
         compute RH type 2 based on RRH
         get packet source from IP header
         send ICMP error to source including RH type 2.
      }
      else {
         get packet source from IP header
         send ICMP error to source with no RH.
      }
   }
```

When the MR receives an ICMP error message, it checks whether it is the final destination of the packet by looking at the included packet. If the included packet has an RRH, then the MR will use the RRH to forward the ICMP to the original source of the packet.

9.3 Processing of RHH for Outbound Packets

```
if no RRH in outer header               /* First Mobile Router specific */
   or RRH present but saturated {       /* Need a nested encapsulation */

   if RRH is saturated {
      do ICMP back (RRH too small)
   }

   /* put packet in sliding reverse tunnel */
   insert new IP header plus RRH
   set source address to the MR Home Address
   set destination address to the MR Home Agent Address
   add an RRH with all slots zeroed out
   compute IPsec AH on the resulting packet
}

/* All MRs including first */
if packet size <= MTU {
   select first free slot in RRH bottom up
   set it to source address from IP header
   overwrite source address in IP header with MR CareOf
   transmit packet
} else {
   do ICMP back (Packet too Big)
}
```

If the packet already contains an RRH in the outer header, and has a spare slot, the MR adds the source address from the packet IP header to the RRH and overwrites the source address in the IP header with its CoA. As a result, the packets are always topologically correct.

Else, if the RRH is present but is saturated, and therefore the source IP can not be added, the MR sends a ICMP 'RRH too small' to the tunnel endpoint which originated the outer packet, using the RRH info to route it back. The ICMP message is a warning, and the packet is not discarded. Rather, the MR does a nested encapsulation of the packet in its own reverse tunnel home with an additional RRH.

Else, if the packet does not have an RRH, the MR puts it in its reverse tunnel, sourced at the CoA, with an RRH indicating in slot 0 the Home Address of the MR, and with proper IPsec AH as described further in Section 11.1.

9.4 Processing of Tree Information Option

The Tree Information Option in Router Advertisement messages allows

the Mobile Router to select a tree and learn about its capabilities.

The treeDepth can be used to compute the optimum number of slots in the RRH.

The Option contains an entry for the home address in slot 0, and one for every CareOf on the way but that of the last Mobile Router (TLMR). As the TLMR sets the treeDepth to 0 and each MR increments it on the way down the tree, the optimum number of slots is normally (treeDepth+1), where treeDepth is the depth advertised by the MR over its Mobile Networks.

9.5 Processing of the extended Routing Header Type 2

```
if Segments Left = 0 {

   /* new check: packet must be looped back internally */
   if packet doesn't come from a loopback interface {
       discard the packet
       return
   }

   proceed to process the next header in the packet, whose type is
   identified by the Next Header field in the Routing header
}
else if Hdr Ext Len is odd {
      send an ICMP Parameter Problem, Code 0, message to the Source
      Address, pointing to the Hdr Ext Len field, and discard the
      packet
}
else {
   compute n, the number of addresses in the Routing header, by
   dividing Hdr Ext Len by 2

   if Segments Left is greater than n {
      send an ICMP Parameter Problem, Code 0, message to the Source
      Address, pointing to the Segments Left field, and discard the
      packet
   }
   else {
      decrement Segments Left by 1;
```

```
        compute i, the index of the next address to be visited in
        the address vector, by subtracting Segments Left from n

        if Address [i] or the IPv6 Destination Address is multicast {
           discard the packet
        }
        else {
           /* new security check */
```



```
           if Address [i] doesn't belong to one of the NEMO prefixes {
               discard the packet
               return
           }

           /* new check: keep MIPv6 behavior: prevent packets from being
            * forwarded outside the node.
            */
           if Segments Left equals 0 and Address[i] isn't the node's own
           home address {
               discard the packet
               return
           }
           swap the IPv6 Destination Address and Address[i]

           if the IPv6 Hop Limit is less than or equal to 1 {
              send an ICMP Time Exceeded -- Hop Limit Exceeded in
              Transit message to the Source Address and discard the
              packet
           }
           else {
              decrement the Hop Limit by 1

              resubmit the packet to the IPv6 module for transmission
              to the new destination;
           }
        }
     }
  }
```

9.6 Decapsulation

A MR when decapsulating a packet from its HA must perform the following checks

1. Destination address

   The destination address of the inner packet must belong to one of the NEMO prefixes.

10. Mobile Host Operation

When it is at Home, a Mobile Host issues packets with source set to its home address and with destination set to its CN, in a plain IPv6 format.

When a MH is not at home but is attached to a foreign link in the Fixed Infrastructure, it SHOULD use MIPv6 as opposed to this draft to manage its mobility.

When a MH is visiting a foreign Mobile Network, it forwards its outbound packets over the reverse tunnel (including RRH) to its HA. One can view that operation as a first MR process applied on a plain IPv6 packet issued by an LFN.

As a result, the encapsulating header include:

   with source set to the MH COA and destination set to the MH HA

   with slot 0 set to the MH Home Address

The inner packet is the plain IPv6 packet from the MH Home Address to the CN.

11. Security Considerations

This section is not complete; further work is needed to analyse and solve the security problems of record and source route.

Compared to MIPv6, the main security problem seems to be the fact that the RRH can be modified in transit by an in-axis attacker. It has to be noted that an in-axis attacker (for example any MR in the

NEMO) can perform more effective attacks than modifying the RRH.

Selecting the tree to attach to is a security critical operation outside of the scope of this draft.

11.1 IPsec Processing

The IPsec [7] AH [8] and ESP [9] can be used in tunnel mode to provide different security services to the tunnel between a MR and its HA. ESP tunnel mode SHOULD be used to provide confidentiality and authentication to the inner packet. AH tunnel mode MUST be used to provide authentication of the outer IP header fields, especially the RH.

The Routing Header Type 2 is treated as Type 0, namely as mutable but predictable [8], and so will be included in the Authentication Data calculation. As per IPsec, the sender must order the field so that it appears as it will at the receiver, prior to performing the Integrity Check Value (ICV) computation.

The Routing Header Type 4 is "partially mutable", and as such can be included in the Authentication Data calculation. Given the way Type

4 is processed, the sender cannot order the field so that it appears as it will at the receiver; this means the receiver will have to shuffle the fields.

The sender (the MR) will zero out all the slots and the Segment Used field of the RRH, and will put as source address of the outer packet its Home Address, and then will perform the ICV computation.

The receiver (the HA) will put the entry in slot 0 (the MR Home Address) in the source address and will zero out all the slots and the Segment Used field of the RRH, and then will perform the ICV verification.

11.2 New Threats

The RH type 4 is used to construct a MIPv6 RH type 2 with additional semantics, as described in Section 4.1. Since RH type 2 becomes a multi hop option like RH type 0, care must be applied to avoid the spoofing attack that can be performed with the IPv4 source route

option. This is why IPv6 [10] takes special care in responding to packets carrying Routing Headers.

AH authenticates the MR Home Address identity and the RRH sequence number. The RRH sequence number is to be used to check the freshness of the RRH; anti-replay protection can be obtained if the receiver enables the anti-replay service of AH [8].

As a consequence, the only kind of successful attack seems to require to be able to modify the packet in flight.

If one of the RRH entry is faked either to an address outside the tree or to an address that doesn't match the tree topology (not belonging to one of the NEMO prefixes at that level) then the reply packet containing a RH type 2 built out of the previous RRH will be dropped by the first MR that processes that entry, as described in Section 9.

It is still an issue how to validate that the source of the outer packet is the actual TLMR as opposed to a forged IP address put by an on-axis attacker outside the NEMO.

12. Acknowledgements

The authors wish to thank Steve Deering, Fred Baker, Thomas Fossati, Dave Auerbach, Kent Leung, Francois Le Faucheur, Dana Blair, Dan Shell, Dave Forster and Massimo Lucchina for their constructive comments on the ideas that sustain this document.

References

[1] Perkins, C., Johnson, D. and J. Arkko, "Mobility Support in IPv6", draft-ietf-mobileip-ipv6-17 (work in progress), May 2002.

[2] Ernst, T., "Network Mobility Support Terminology", draft-ernst-monet-terminology-00 (work in progress), February 2002.

[3] Kniveton, T., "Mobile Router Support with Mobile IP", draft-kniveton-mobrtr-01 (work in progress), March 2002.

[4] Ernst, T., Castelluccia, C., Bellier, L., Lach, H. and A. Olivereau, "Mobile Networks Support in Mobile IPv6 (Prefix Scope Binding Updates)", draft-ernst-mobileip-v6-network-03 (work in progress), March 2002.

[5] Deering, S. and B. Zill, "Redundant Address Deletion when Encapsulating IPv6 in IPv6", draft-deering-ipv6-encap-addr-deletion-00 (work in progress), November 2001.

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

[7] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[8] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[9] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, November 1998.

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

[11] Narten, T., Nordmark, E. and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[12] Conta, A. and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 2463, December 1998.

[13] Reynolds, J., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, January 2002.

Authors' Addresses

Pascal Thubert
Cisco Systems Technology Center
Village d'Entreprises Green Side

400, Avenue Roumanille
Biot - Sophia Antipolis 06410
FRANCE

EMail: pthubert@cisco.com

Marco Molteni
Cisco Systems Technology Center
Village d'Entreprises Green Side
400, Avenue Roumanille
Biot - Sophia Antipolis 06410
FRANCE

EMail: mmolteni@cisco.com

Appendix A. Optimizations

A.1 Prefix Scope Binding Updates

[4] suggests modifications to MIPv6 to enable support for LFNs in non-nested NEMOs, leaving for later investigation more complex scenarios like MNs behind the MR or nested NEMOs.

The solution described there has bi-directional route optimization as in MIPv6: the CN to MR direction uses the RH type 2, while the MR to CN direction uses the home address destination option. Route optimization is obtained by introducing a new kind of binding update, the Prefix Scope BU (PSBU) and by modifying the CN and MR operations in order to exploit it.

The MR has to keep track of all the pending communications between hosts in his NEMO and their CNs, in order to send to the CNs a PSBU each time the MR changes its point of attachment.

If we extend [4] in such a way that each MR in a nested NEMO sends a full set of PSBUs each time it changes its point of attachment, then each CN by receiving all the PSBUs and processing them can infer a partial topology of the nested NEMO, sufficient to build a multi-hop routing header for packets sent to nodes inside the nested NEMO.

However, this extension seems to come at a too high price:

1. PSBU storm

   when one MR changes its point of attachment, it needs to send a PSBU to all the CNs of each node behind him. When the NEMO is nested, the number of nodes and relative CNs can be huge. In order to send the PSBUs, the MR has to keep track of all the traffic it forwards to maintain his list of CNs.

2. CN operation

   The computation burden of the CN becomes heavy, because it has to analyze each PSBU in a recursive fashion in order to deduct nested NEMO topology required to build a multi hop routing header.

3. Missing PSBU

   If a CN doesn't receive the full set of PSBU sent by the MR, it will not be able to infer the full path to a node inside the nested NEMO. The RH will be incomplete and the packet may or may not be delivered. If PSBU are sent asynchronously by each MR, then, when the relative position of MRs and/or the TLMR point of attachment change rapidly, the image of Mobile Network that the CN maintains is highly unstable.

A conclusion is that the path information must be somehow aggregated to provide the CN with consistent snapshots of the full path across the Mobile Network. If this is achieved by a series of stacked Home Address Options, then the problem turns into a format war and about the opportunity to insert headers in a packet as opposed to tunneling. Either way is a route record, which is why defining a real V6 version of LSRR is relevant in the first place.

A.2 Path Optimization with RRH

The body of the draft presents RRH as a header that circulates in the reverse tunnel exclusively. The RRH format by itself has no such limitation. This section illustrates a potential optimization for end-to-end traffic between a Mobile Network Node and its Correspondent Node.

The MNN determines that it is part of a Nemo by screening the Tree Information option in the RA messages from its Access Router. In particular, the MNN knows the TreeDepth as advertised by the AR. An initial test phase could be derived from MIPv6 to decide whether optimization with a given CN is possible.

When an MNN performs end-to-end optimization with a CN, the MNN

inserts an empty RRH inside its packets, as opposed to tunneling them home, which is the default behavior of a Mobile Host as described in Section 10. The number of slots in the RRH is initially the AR treeDepth plus 1, but all slots are clear as opposed to the MR process as described in Section 9. The source address in the header is the MNN address, and the destination is the CN.

The AR of the MNN is by definition an MR. Since an RRH is already present in the packet, the MR does not put the packets from the MNN on its reverse tunnel, but acts as an intermediate MR; it adds the source address of the packet (the MNN's address) in the RRH (in slot 0) and stamps its careOf instead in the IP header source address field. Recursively, all the MRs on a nested network trace in path in the RRH and take over the source IP.

The support required on the CN side extends MIPv6 in a way similar to the extension that this draft proposes for the HA side. The CN is required to parse the RRH when it is valid, refresh its BCE accordingly, and include an RH type 2 with the full path to its packets to the MNN.

Note that there is no Bind Update between the MNN and the CN. The RRH must be secured based on tokens exchanged in the test phase. For the sake of security, it may be necessary to add fields to the RRH or to add a separate option in the Mobility Header.

A.3 Packet Size Optimization

RRH allows to update the Correspondent BCE on a per packet basis, which is the highest resolution that we can achieve. While this may cope with highly mobile and nested configurations, it can also be an overkill in some situations.

The RRH comes at a cost: it requires processing in all intermediate Mobile Routers and in the Correspondent Node. Also, a RRH increases the packet size by more than the size of an IP address per hop in the Mobile Network.

This is why an additional Routing Header is proposed (type 3). The semantics of type 3 are very close to type 4 but:

o Type 3 has only one slot, for the Home Address of the source.

o When it can not add the source to the RH type 3 of an outbound packet, an intermediate MR:

   * MR MUST NOT send ICMP (RRH too small)

   * MUST NOT put the packet in a reverse tunnel

   Rather, it simply overwrites the source and forwards the packet up the tree as if the RRH had been properly updated.

```
/* MR processing on outbound packet with RH type 3 support */
{

   if no RH type 3 or 4 in outer header    /* First Mobile Router specific *
      or RH type 4 present but saturated { /* Need a nested encapsulation */

      if RRH is saturated {
         do ICMP back (RRH too small)
      }

      /* put packet in sliding reverse tunnel */
      insert new IP header plus RRH
      set source address to the MR Home Address
      set destination address to the MR Home Agent Address
      add an RRH with all slots zeroed out
      compute IPsec AH on the resulting packet
   }

   /* All MRs including first */
   if packet size > MTU {
      do ICMP back (Packet too Big)
   } else if RRH {
      select first free slot in RRH bottom up
      set it to source address from IP header
      overwrite source address in IP header with MR CareOf
      transmit packet
   } else if RH type 3 {
      if slot 0 is still free {
         /* this is end-to-end optimization */
```

```
            set it to source address from IP header
         }
         overwrite source address in IP header with MR CareOf
         transmit packet
      }
   }
```

o Since the path information is not available, the correspondent MUST NOT update its BCE based on the RH type 3. The CN (or HA) identifies the source from the entry in slot 0 and may reconstruct the initial packet using the CareOf in slot 1 as source for AH purposes.

A.3.1 Routing Header Type 3 (HAddr option replacement)

This is an RH-based alternative to the Home Address destination option. Its usage is described in Appendix A.3.

The Type 3 Routing Header has the following format:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Next Header  |  Hdr Ext Len  | Routing Type=3| Segments Used |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Reserved                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   |                                                               |
   +                        Home Address                           +
   |                                                               |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Next Header

8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [13].

Hdr Ext Len

8-bit unsigned integer. Length of the Routing header in 8-octet units, not including the first 8 octets. For the Type 3 Routing header, Hdr Ext Len is always 2.

Routing Type

8-bit unsigned integer. Set to 3.

Segment Used

8-bit unsigned integer. Number of slots used. Either 0 or 1. When the field is zero, then there is no MR on the path and it is valid for a CN that does not support RRH to ignore this header.

Reserved

32-bit reserved field. Initialized to zero for transmission;

ignored on reception.

Home Address

128-bit home address of the source of the packet.

The decision to sent RH type 3 or type 4 is up to the source of the RRH. Several algorithms may apply, one out of N being the simplest.

IPsec HA processing is done as described in Section 11.1 for Type 4.

Appendix B. Multi Homing

B.1 Multi-Homed Mobile Network

Consider difference between situation A and B in this diagram:

```
              ===?==          ==?===
                MR1             MR2
```

```
                   |              |
              ==?=====?==   ==?======   situation A
               MR3   MR4     MR5
                |     |       |
               ===   ===     ===


                ===?==          ==?===
                  MR1            MR2
                   |              |
              ==?=====?=======?======   situation B
               MR3   MR4     MR5
                |     |       |
               ===   ===     ===
```

Going from A to B, MR5 may now choose between MR1 and MR2 for its Access (default) Router. In terms of Tree Information, MR5, as well as MR3 and MR4, now sees the MR1's tree and MR2's tree. Once MR5 selects its AR, MR2, say, MR5 belongs to the associated tree and whether MR1 can be reached or not makes no difference.

As long as each MR has a single default router for all its outbound traffic, 2 different logical trees can be mapped over the physical configurations in both situations, and once the trees are established, both cases are equivalent for the processing of RRH.

Note that MR5 MUST use a CareOf based on a prefix owned by its AR as source of the reverse tunnel, even if other prefixes are present on the Nemo, to ensure that a RH type 2 can be securely routed back.

B.2 Multihomed Mobile Router

Consider the difference between situation B and C in this diagram:

```
                ===?==          ==?===
                  MR1            MR2
                   |              |
              ==?=====?=======?======   situation B
```

```
              MR3   MR4     MR5
               |     |       |
              ===   ===     ===


               ==? ?==
                 MR1
                  |
             ==?=====?=======?======   situation C
              MR3   MR4     MR5
               |     |       |
              ===   ===     ===
```

In situation C, MR2's egress interface and its properties are migrated to MR1. MR1 has now 2 different Home Addresses, 2 Home Agents, and 2 active interfaces.

If MR1 uses both CareOf addresses at a given point of time, and if they belong to different prefixes to be used via different access routers, then MR1 actually belongs to 2 trees. It must perform some routing logic to decide whether to forward packets on either egress interface. Also, it MUST advertise both tree information sets in its RA messages.

The difference between situations C and B is that when an attached router (MR5, say) selects a tree and forwards egress packets via MR1, it can not be sure that MR1 will actually forward the packets over that tree. If MR5 has selected a given tree for a specific reason, then a new source route header is needed to enforce that path on MR1.

The other way around, MR5 may leave the decision up to MR1. If MR1 uses the same access router for a given flow or at least a given

destination, then the destination receives consistent RRHs. Otherwise, the BCE cache will flap, but as both paths are valid, the traffic still makes it through.

The RRH seems compatible with the various cases of multi-homing exposed here, though in some cases, some additional work is needed.

Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.