Internet Engineering Task Force P. Thubert
Internet-Draft Cisco
Intended status: Standards Track R. Wakikawa
Expires: August 30, 2007 Keio University
C. Bernardos
UC3M
R. Baldessari
NEC Europe
J. Lorchat
Keio University
February 26, 2007

Network In Node Advertisement
draft-thubert-nina-00.txt

Status of this Memo

Copyright Notice

Abstract

The Internet is evolving to become a more ubiquitous network, driven by the low prices of wireless routers and access points and by the users' requirements of connectivity anytime and anywhere. For that reason, a cloud of nodes connected by wireless technology is being created at the edge of the Internet. This cloud is called a MANEMO Fringe Stub (MFS). It is expected that networking in the MFS will be highly unmanaged and ad-hoc, but at the same time will need to offer excellent service availability. The NEMO Basic Support protocol could be used to provide global reachability for a mobile access network within the MFS and the Tree-Discovery mechanism could be used to avoid the formation of loops in this highly unmanaged structure. Since Internet connectivity in mobile scenarios can be costly, limited or unavailable, there is a need to enable local routing between the Mobile Routers within a portion of the MFS. This form of local routing is useful for Route Optimization (RO) between Mobile Routers that are communicating directly in a portion of the MFS.

NINA is the second of a 2-passes routing protocol; a first pass, Tree Discovery, builds a loop-less structure -a tree-, and the second pass, NINA, exposes the Mobile Network Prefixes (MNPs) up the tree. The protocol operates as a multi-hop extension of Neighbor Discovery (ND), to populate TD-based trees with prefixes, and establish routes towards the MNPs down the tree, from the root-MR towards the MR that owns the prefix, whereas the default route is oriented towards the root-MR.

The NINA protocol introduces a new option in the ND Neighbor Advertisement (NA), the Network In Node Option (NINO). An NA with NINO(s) is called a NINA (Network In Node Advertisement). NINA is designed for a hierarchical model where an embedded network is abstracted as a Host for the upper level of network abstraction. With NINA, a Mobile Router presents its sub-tree to its parent as an embedded network and hides the inner topology and movements.

Table of Contents

1. Introduction

Mobile IP [3] allows transparent routing of IPv4 datagrams to mobile nodes in the Internet. Mobile IPv6 (MIPv6) [4] extends this facility for IPv6, and NEMO [5] enables it for mobile prefixes. In any case, a mobile node is always identified by its Home Address (HoA), regardless of its current point of attachment to the Internet. In turn, MANET [11] allows a set of unrelated nodes and routers to discover their peers and establish communication.

Mobile Routers (MRs) may attach to other MRs and form a Care-of Address (CoA) from a Mobile Network Prefix (MNP). As a result, MRs are really MARs, Mobile Access Routers, because they can accept connections from other MRs on their ingress interfaces. When Mobile Routers attach to other Mobile Routers with a single Care-of Address in a loop-less manner, they end up building trees. This process is described in Tree Discovery (TD) [6].

This draft provides a minimum extension to IPv6 Neighbor Discovery (ND) Neighbors Advertisements (NA) - called NINA (Network In Node Advertisement) - extending RFC 4191 [9] to add the capability to include a prefix option - called NINO (Network In Node Option) - in the NAs. This enables an MR to learn the prefixes of all other MRs down its sub-tree. Note that NINO is pronounced NEE-GNO and NINA is pronounced NEE-GNA.

A NEMO Mobile Router has a double behavior. On its egress interfaces, which are used to backhaul the traffic to the Home Network and the rest of the Internet, it is seen as a Mobile Node

(MN), performing the IPv6 and MIPv6 host-required features such as neighbor and router discovery [2]. On the (ingress) interfaces to the Mobile Networks, the Mobile Router behaves as an IPv6 router with support of the MIPv6 requirements on routers. This is why TD [6] extends ND RA over the ingress interface of a Mobile Router whereas NINA extends ND NAs to advertise over the egress interface the prefixes that are reachable via the MR.

2. Terminology

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 [1].

Readers are expected to be familiar with all the terms defined in the RFC 3753 [7], the NEMO Terminology draft [8] and the MANEMO Problem Statement draft [17].

NINO (Network In Node Option): a new Neighbor Discovery (ND) option that adds the capability to include a prefix option in Neighbor Advertisements (NAs) and Solicitations (NSs).

NINA (Network In Node Advertisement): a Neighbor Discovery (ND) Neighbor Advertisement (NA) carrying a NINO. NINA is also used to refer to the protocol itself (defined in this document).

3. Motivations

The Internet is evolving to become a more ubiquitous network, driven by the low prices of wireless routers and access points and by the users' requirements of connectivity anytime and anywhere. For that reason, a cloud of nodes connected by wireless technology is being created at the edge of the Internet. This cloud is called a MANEMO Fringe Stub (MFS) in [17]. Examples of wireless technologies used within a MFS are wireless metropolitan and local area network protocols (WiMAX, WLAN, 802.20, etc), short distance wireless technology (bluetooth, IrDA, UWB), and radio mesh networks (e.g., 802.11s). It is expected that networking in the MFS will be highly unmanaged and ad-hoc, but at the same time will need to offer excellent service availability.

The NEMO Basic Support protocol [5] could be used to provide global reachability for a mobile access network within the MFS. Analogously, the Tree-Discovery mechanism [6] could be used to avoid the formation of loops in this highly unmanaged structure. However, even with these two technologies in place, packet delivery within the MFS can still be highly inefficient. Since Internet connectivity in mobile scenarios can be costly, limited or unavailable, there is a need to enable local routing between the Mobile Routers within a portion of the MFS. NINA can provide this form of local routing; it is an example of Route Optimization (RO) between Mobile Routers that are communicating directly in a portion of the MFS.

4. Rationale for the proposed solution

4.1. Why ND based

NINA extends the Neighbor Discovery protocol to address the MANEMO requirements listed in [17], although MANET protocols [12], [14], [15] provides similar features such as local routing and Internet access over multihop.

One of the drawbacks of MANET protocols is the question of which protocol should be used. AODV, DSR, DYMO, OLSR, etc. are standardized in IETF and each has distinct features, like proactive and reactive. In MANEMO scenarios, Mobile Routers, mobile hosts, and fixed access routers are involved, and therefore, it is highly important to deploy a consistent protocol in the network. On the other hand, ND is a core component of IPv6 and is supported by all IPv6 nodes. All IPv6 nodes can process a NINO(s) in ND messages if desired.

MANEMO does not require full link states of a network as OLSR does, it only requires path to and from the exit router (tree root) in the tree fashion. Flooding the entire network with route information is a redundant process and its overhead is not negligible. ND simply carries prefix information to setup the path from the tree root to each mobile router/node.

4.2. Why NA based

Since MR appears as a host on the egress interface side, it is legitimate to use NA in the visited network. There are two reasons for that:

- o If an MR advertises itself as a router in the visited network using RA, it might get used as a default router by LFNs attached to the visited network and cause trouble.

- o By using NINA, the whole part of the fringe behind the MR has the footprint of a single host from the visited network standpoint (and moves as a single host).

By using NINA on top of a TD established tree, MANEMO can be made to reproduce the NEMO behavior for a whole subtree by reducing to a single host footprint, and retain NEMO compatibility by avoiding spurious RAs. Thus, a whole subtree can move within the fringe as a single host.

4.3. Relationship with TD

NINA exploits the loop-less cluster established by Tree Discovery, so it does not need to provide loop avoidance.

With TD, MRs setup a default route up the tree via the parent Access Router, and all the packets are directed by default towards the clusterhead (Top Level Mobile Router or root-MR in NEMO terms). To provide complete reachability, it is enough for NINA to expose the prefixes down the tree from any given MR, while propagating prefixes information up the tree.

This allows an extreme conciseness of the routing information, with no topological knowledge past the first hop. That conciseness enables a high degree of movement within the nested structure; in particular, a movement within a subtree is not seen outside of that subtree, so most of the connectivity is maintained at all times while there might never be such a thing as a convergence.

4.4. Relationship with NEMO

The Reverse Routing Header (RRH) described in [16] operates in the nested NEMO as a layer 3 Source Route Bridging (SRB) technique for nested NEMO Route Optimization. It allows a quick reaction to inner movements with the resolution of the packet; but the cost, an IPv6 address per packet per hop, might be deemed excessive.

Also, the Home Agent needs to cache the RRH in its binding cache, and again, the overhead might be significant for a large deployment.

On the other hand, NINA establishes states in the intermediate nodes, in a fashion similar to Transparent Bridging (TB), but at layer 3. The integration of these 2 approaches allows switching between SRB to TB models dynamically as the NINA states are populated or become obsolete. To obtain this capability, the operation of an intermediate MR described in [16] is altered in the following manner:

- o If the MR has a (NINA) route to the upper entry in the RRH via the source of the packet, it still updates the source of the packet with its own Care-of Address, but does not save the previous source as a new entry in the RRH.

- o At best, if NINA has established states all along in a given branch of the tree, the RRH for that branch has always 2 entries, the first MR's Home Address, and its Care-of Address, regardless of the depth of the first MR in the nested NEMO.

o When some MRs in the tree support NINA and some do not, the resulting RRH will be only partly compressed. Also, if the NINA route does not match the RRH, then the route is obsolete and the source address is added to the RRH as described in [16], in order to ensure a correct routing on the way back. When NINA catches up, the entry will be saved again.

The integration of NINA and RRH can offer the best of 2 worlds: a quick (per packet) resolution to the network changes, and the transparent (stateful) operation when the NINA routing protocol establishes the states in the nested NEMO.

5. Overview

This section provides an overview of the operation of NINA to set-up MNP route state in two different scenarios: a non-nested NEMO-to-NEMO communication and a nested-NEMO one.

5.1. Basic kiss (MR to MR over egress)

A basic scenario where NINA can be used occurs when two or more MRs are attached with their egress interfaces on the same link but do not have connectivity to the infrastructure network (Figure 1). This scenario includes, for example, the case of MRs equipped with short-range communication technology (e.g. WLAN 802.11) that find themselves in their communication range but are isolated from the Internet. The NEMO Basic Support protocol [5] does not allow LFNs to communicate, as it requires that the Home Agent is reachable and that data packets go through it. By exchanging NINA messages, the MRs expose their Mobile Network Prefixes to each other. Consequently, MRs can add in their routing table a temporary route for MNPs exposed by other MRs, which allows data packets to be exchanged between LFNs.

```
  ---+-------------------+-------------+------
     |                   |             |
  +--+--+             +--+--+       +--+--+
  | MR1 |             | MR2 |       | MR3 |
  +---+-+             +-+---+       +--+--+
      |                 |              |
  -+--+---          -+--+--         ---+--+--
   |                 |                    |
+--+---+          +--+---+            +---+--+
| LFN1 |          | LFN2 |            | LFN3 |
+------+          +------+            +------+
```

Figure 1: Basic Kiss scenario

A new multicast group is created (all MANEMO Mobile Routers, see Section 6.1). An MR that needs to know all MNPs on link will multicast a NINA NS to that group with a link scope. Reversely, an MR that wishes to advertise its MNPs to all its siblings on link will

multicast a NINA containing information about all the MNPs it manages to that group with a link scope. A NINA sent to siblings can contain information regarding its own MNPs and only its own MNPs (i.e. it MUST NOT contain information about MNPs managed by others than the MR sending the NINA). This information MUST NOT be propagated by the siblings. For that purpose, the flag in the NINO ('P' bit, see Section 6.2) MUST be set to 0.

MRs that want to advertise their MNPs to all of their siblings, in

addition to answering to NINA NS messages sent to the all MANEMO Mobile Routers multicast address, SHOULD also send periodically unsolicited NINAs to this link-scoped multicast group. This enables information about sibling MNPs to expire (i.e. time out) when MRs disappear from the network.

5.2. Nested NEMO

NINA requires the Tree Discovery protocol to build and maintain a tree topology. It relies on TD to discover that a change occurs in a sub-tree of the topology, and that change triggers a flow of route updates for that sub-tree in the topology.

```
                     +---------------------+
                     |     Internet        |---CN
                     +---------------|-----+
                      /         Access Router
                 MR3_HA              |
                            ======?======
                                MR1
                                 |
                    ====?=============?==============?===
                       MR5           MR2            MR6
                        |             |              |
                  ===========    ===?=========   =============
                                  MR3
                                   |
                             ==|=========?==
                              LFN1      MR4
                                         |
                                     =========
```

Figure 2: Nested NEMO scenario

Each tree that TD self-forms is considered a separate routing topology. If a Mobile Router belongs to multiple of such topologies, then it is expected that both the NINA signaling and the data packets are flagged to follow the topology for which the packet was introduced in the network.

NINA expects a Mobile Router to own one or more Mobile Network Prefix(es) that move with the MR. With that model, it is assumed that there is a single source for the advertisement of a given prefix within a topology. If multiple MRs share a given MNP, some protocol must take place between those MRs to make sure that one and only one MR advertises a given prefix in a given tree.

Tree Discovery formats the nested NEMO into a loop-less logical

graph, thus providing loop avoidance for the NINA protocol. Each time a movement occurs, TD restores the loop-less structure before NINA can operate again and repaint the graph with prefixes.

The root-MR of a nested NEMO is selected for a number of properties, the primary one being an access to the wired infrastructure. It is the default sink for every node in the tree.

More generally, the default gateway for a Mobile Router is its parent up in the tree; the more specific routes, towards the Mobile Network Prefixes, are always oriented down the tree, and NINA advertisements flow up the tree towards the root-MR.

Each NINO contains a prefix and a sequence counter. The Mobile Router that owns the prefix generates the NINO for that prefix, including the sequence counter associated to that prefix and that is incremented each time it generates a new NINO.

Due to a movement, a sub-tree can be temporarily out of sequence and a NINO can be received from a sub-tree where the MR was but is no more, until the parents realize it is gone. But by construction of the tree, there can be a single route to a given prefix, so older information is always invalid.

A parent-MR maintains a state for each prefix it learns from NINA.

In particular, the last sequence number is kept. An out-of-sequence NINO must be disregarded. If the NINO appears valid, it is forwarded to the parent's parent in the next burst, carried by a NINA, together with the parent's own prefixes.

6. Message Formats

6.1. All MANEMO Mobile Routers multicast address

RFC 4291 [10] defines the IPv6 Addressing Model and in particular Multicast Addresses.

Multicast addresses have the following format:

```
|   8    |  4 |  4 |                  112 bits                   |
+------ -+----+----+---------------------------------------------+
|11111111|flgs|scop|                  group ID                   |
+--------+----+----+---------------------------------------------+
```

MANEMO requires a permanently-assigned multicast address: the MANEMO Mobile Routers Group (IANA TBD 199?). As a result:

o FF02:0:0:0:0:0:0:(199?) means all MANEMO Mobile Routers on the

same link as the sender.

6.2. NINO NA option

NINO is a new option of ND Neighbors Advertisements. This specification extends RFC 4191 [9] and adds the capability to include a prefix option in the Neighbor Advertisements (NAs). The NA is a necessary exchange that allows the AR to map the IPv6 address of a node with its L2 address. The prefix option is normally present in Router Advertisements (RAs) only. The meaning of such an option in a NA is the concept of 'network in node', so we refer to the option as NINO (Network In Node Option) and we name the resulting message NINA (Network In Node Advertisement).

When Tree Discovery is used to build a tree, there can be a single route to a given prefix along that tree, so the freshest information is always the best for unicast routes. In order to track that, the NINO includes a sequence counter to the prefix advertisement.

The sequence counter is incremented by the source of the NINO, that is the Mobile Router that owns the MNP, each time it issues a NINA, and then forwarded as is up the tree. A depth is also added for tracking purposes; the depth is incremented at each hop as the NINO is propagated up the tree.

On an egress interface, if NINA is configured, the MR:

- o selects an Access Router (AR) as its point of attachment to the network

- o auto-configures a Care-of Address (CoA)

- o acts as a host as opposed to a router. In particular, it refrains from sending RAs

- o sends NINAs, as unicast, to its AR only, with the propagate ('P') bit set in the NINOs indicating that the NINO can be propagated up the tree

- o sends NINAs as solicited or unsolicited unicast or multicast to siblings, with the propagate ('P') bit reset in the NINOs

indicating that the NINO cannot be propagated up the tree

- o accepts unicast NINAs from any node BUT its AR
- o optionally solicits multicast NINAs from all MANEMO MRs
- o optionally accepts multicast NINAs (note, multicast NINAs advertise only CONNECTED routes)

On an ingress interface, if NINA is configured, the MR:

- o acts as a router, may accept visitors
- o sends RAs with the Tree Information Option (RA-TIO)
- o accepts NINAs from any node

Every NA to the AR contains a NINO. In particular, receiving a Tree Discovery RA-TIO from the AR stimulates the sending of a delayed NINA back, with the collection of all known prefixes (that is the prefixes learned from NINO and the connected prefixes). A NINA is also sent to the AR once it has been selected as new AR after a movement, or when the list of advertised prefixes has changed.

NINA may advertise positive (prefix is present) or negative (removed) NINOs. A no-NINO is stimulated by the disappearance of a prefix below. This is discovered by timing out after a request (a RA-TIO) or by receiving a no-NINO. A no-NINO is a NINO with a Route lifetime of 0.


```
 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     |    IPv6 PL    | IPv4 PL       |
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Route Lifetime                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   NINO Depth  |6|4|P|L|       |  NINO Sequence                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        IPv4 Prefix                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                                 |
+                                                                 +
|                                                                 |
+                        IPv6 Prefix                              +
|                                                                 |
+                                                                 +
|                                                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type:

NINO NA (number to be assigned by IANA)

Length:

8-bit unsigned integer. The length of the option (including the Type and Length fields) in units of 8 octets.

IPv6 PL:

Number of valid leading bits in the IPv6 Prefix.

IPv4 PL:

Number of valid leading bits in the IPv4 Prefix.

Route Lifetime:

32-bit unsigned integer. The length of time in seconds (relative to the time the packet is sent) that the prefix is valid for route determination. A value of all one bits (0xFFFFFFFF) represents infinity. A value of all zero bits (0x00000000) indicates a loss of reachability.

NINO Depth:

Set to 0 by the MR that owns the MNP and issues the NINO. Incremented by all MRs that propagate the NINO.

'6' bit:

Indicates that the IPv6 prefix is valid.

'4' bit:

Indicates that the IPv4 prefix is valid.

'P' bit:

Set only in a parent-child relationship, this flag indicates that the NINO can be propagated up the tree.

'L' bit:

Indicates that the prefix or address is on-link as opposed to another interface of the MR. This is useful for a child MR to expose its IPv4 address on its egress interface. In that case, the parent can set up forwarding to all the IPv4 prefixes in the NINA via that address on this link.

NINO Sequence:

Incremented by the MR that owns the MNP for each new NINO for that prefix. Left unchanged by all MRs that propagate the NINO. A lollipop mechanism is used to wrap from 0xFFFF directly to 10.

IPv4 Prefix:

4-byte field containing an IPv4 address or a prefix of an IP address. The IPv4 PL field contains the number of valid leading bits in the prefix. The bits in the prefix after the prefix length (if any) are reserved and MUST be initialized to zero by the sender and ignored by the receiver.

IPv6 Prefix:

Variable-length field containing an IPv6 address or a prefix of an IPv6 address. The IPv6 PL field contains the number of valid leading bits in the prefix. The bits in the prefix after the prefix length (if any) are reserved and MUST be initialized to zero by the sender and ignored by the receiver.

6.3. NINO NS option

NINA messages MAY be solicited or unsolicited. In particular, for privacy reasons, MRs may disable sending of unsolicited NINAs and send only NINA to other nodes or MRs that requested it. This allows the solicited MRs to decide whether or not to reveal its MNP. Definition of policies for revealing MNP is out of scope of this document.

As opposed to standard Neighbor Discovery [2], NINO Neighbor Solicitations aim at resolving an MNP into an IPv6 address (MR link local address) and not a known IPv6 address into a link-layer address. Therefore, a NINA NS is sent to the All MANEMO MRs multicast address and optionally contains the target MNP in a NINO NS option.

An MR receiving a NINA NS compares its MNP with the one specified in the NINO NS option. If the MNP matches the requested one and privacy policies allow that, the MR sends a unicast NINA message to the sender.

When there is no NINA NS option in the NS, the MR adds NINOs to its NAs based on its policy. The NINA NS message MAY contain also a NINO to inform other MRs about its MNP.

```
 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     | Prefix Length |           |4| |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Requested IPv4 prefix                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                     Requested IPv6 Prefix                     +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type:

NINO NS (number to be assigned by IANA).

Length:

8-bit unsigned integer. The length of the option (including the Type and Length fields) in units of 8 octets. This is set to 1 for an IPv4 query and to 3 for an IPv6 query.

Prefix Length:

Number of valid leading bits in the IPv4/IPv6 Requested Prefix.

'4' bit:

If set to 1, it indicates that the target (requested) prefix is an IPv4 prefix. If set to 0, it indicates that the target (requested) prefix is an IPv6 prefix.

Requested IPv4 Prefix:

The IPv4 MNP that the sender wants to resolve into an IPv4 address.

Requested IPv6 Prefix:

The IPv6 MNP that the sender wants to resolve into an IPv6 address (MR link local address).

7. Mobile Router Operation

The Mobile Router operation is autonomous, based on the information provided by the potential Access Routers in sight. Each MR selects an AR (a MAR) in a loop-less and case-optimized fashion, and installs a default route up the tree via the selected AR. The resulting tree (the cluster) may never be globally stable enough to be mapped in a global graph. So the adaptation to local movements must be rapid and localized.

For NEMO flows, the Reverse Routing Header allows the update to the path on a per packet basis. Hopefully, the root of the tree (the clusterhead) is connected to the infrastructure where Home can be reached, and can be used as a gateway to discover Home. When the NEMO tunnel is established, it becomes the default route for the MR.

If the tree is not connected to the infrastructure or in any case if Home can not be reached, MRs need an ad-hoc protocol to establish local connectivity. This specification takes advantage of the TD cluster and allows an MR to discover the prefixes below itself.

NINA information can be redistributed in a routing protocol, MANET or IGP. But the MANET or the IGP SHOULD NOT be redistributed into NINA. This creates a hierarchy of routing protocols where NINA routes stand somewhere between connected and IGP routes.

NINA also allows a compression of the Reverse Routing header when the routes match the topology as traced by RRH on a per packet basis. In particular, if a NINA route exists to the first entry in the RRH via the source of the packet, then the MR can override the source of the packet with its own CoA without adding the original source to the RRH. At that point, the RRH operation becomes loose, in other words an hybrid between transparent (stateful) and source routing.

As a result:

- o Tree Discovery establishes a tree using extended Neighbor Discovery RS/RA flows.
- o The NEMO Basic Support protocol exploits the tree to get optimally out of a nested set of MRs and register Home.
- o RRH extends the NEMO Basic Support to provide Route Optimization and faster path reestablishment.
- o NINA also extends Neighbor Discovery in order to establish quickly the routes down the cluster.

NINA maintains abstract lists of known prefixes. A prefix entry contains the following abstract information:

- o The state of the entry: ELAPSED, PENDING, or CONFIRMED.
- o A reference to the adjacency that was created for that prefix.
- o A reference to the ND entry that was created for the advertiser Neighbor.
- o The IPv6 address of the advertiser Neighbor.
- o The logical equivalent of the full NINA information.
- o A reference to the interface of the advertiser Neighbor.
- o A 'reported' Boolean to keep track whether this prefix was reported already to the parent AR.

o A counter of retries to count how many RA-TIOs were sent on the interface to the neighbor without reachability confirmation for the prefix.

NINA stores the prefix entries in either one of 3 abstract lists; the Connected, the Reachable and the Unreachable lists.

The Connected list corresponds to the MNP of the Mobile Router.

As long as an MR keeps receiving NINOs for a prefix timely, its prefix entry is listed in the Reachable list.

Once scheduled to be destroyed, a prefix entry is moved to the Unreachable list if the MR has a parent to which it sends NINOs, otherwise the entry is cleaned up right away. The entry is removed from the Unreachable list when the parent changes or when a no-NINO is sent to the parent indicating the loss of the prefix.

NINA requires 2 timers; the DelayNA timer and the Destroy Timer.

o The DelayNA timer is armed upon a stimulation to send a NINA (such as a TIO from the AR). When the timer is armed, all entries in the Reachable list as well as all entries for Connected list are set to not reported yet.

o The DelayNA timer has a duration that is DEF_NA_LATENCY divided by 2 with the tree depth.

o The Destroy timer is armed when at least one entry has exhausted its retries, which means that a number of RA-TIO were sent over the ingress interface but that the entry was not confirmed with a NINO. When the destroy timer elapses, for all exhausted entries, the associated route is removed, and the entry is scheduled to be destroyed.

o The Destroy timer has a duration of min (MAX_DESTROY_INTERVAL, RA_INTERVAL).

7.1. NINA messages between siblings

When sending NINA to siblings, an MR includes the NINOs corresponding to the prefix entries in the Connected list, with the 'P' bit set to 0.

Depending on its policy, the receiving MR MAY install a route to the prefix in the NINO via the link local address of the source MR but it SHOULD NOT propagate the information, either as a NINO or by means of redistribution into a routing protocol, since the 'P' bit is not set.

7.2. Multicast TD RA messages from parent

When ND sends a NA to the AR, NINA extends the message with prefix options for:

- o All the prefixes that are not 'DELETED' for all the ingress interfaces.

- o All the prefixes in the removed list as no-NINO.

- o All the prefixes in the advertised list that are not reported yet. The entries are set to reported.

When ND receives a NA from a visitor over an ingress interface, NINOs are processed in a loop. For known prefixes, the sequence counter in the NINO is checked against the last received and the update is used only if the sequence is newer. This filters out obsolete advertisements when a prefix has moved between 2 subtrees attached to a same node.

If a prefix is advertised as a no-NINO, the associated route is removed, and the entry is transferred to the removed list. Otherwise, the route table is looked up:

- o If a preferred route to that prefix from another protocol already exists, the prefix is ignored.

- o If a new route can be created, a new prefix entry is allocated to track it, as CONFIRMED, but not reported.

- o If a NINA route existed already via the same Neighbor, it is CONFIRMED.

o If a NINA route existed via a different Neighbor, this is equivalent to a no-NINO for the previous entry followed by a new NINO for the new entry. So the old entry is scheduled to be destroyed, whereas the new one is installed.

7.3. Unicast NINA messages from child to parent

When sending NINA to its parent, an MR includes the NINOs about not already reported prefix entries in the Reachable and Connected lists, as well as no-NINOs for all the entries in the Unreachable list. The 'P' bit in the NINOs is set, indicating that the parent can propagate the NINOs up the tree. Depending on its policy, the receiving MR SHOULD install a route to the prefix in the NINO via the link local address of the source MR and it SHOULD propagate the information, either as a NINO or by means of redistribution into a routing protocol.

The RA-TIO from the root-MR is used to synchronize the whole tree. Its period is expected to range from 500ms to hours, depending on the stability of the configuration and the bandwidth available.

When an MR receives a RA-TIO over an egress interface from the current parent AR, the DelayNA is armed to force a full update. As described in [6] the MR also issues a propagated RA-TIO over all its ingress interfaces, after a small jitter that aims at minimizing collisions of RA-TIO messages over the radio as it is propagated down the tree.

The design choice behind this is NOT TO synchronize the parent and children databases, but instead to update them regularly to cover from the loss of packets. The rationale for that choice is movement. If the topology can be expected to change frequently, synchronization might be an excessive goal in terms of exchanges and protocol complexity. This results in a simple protocol with no real peering.

When the MR send a RA-TIO over an ingress interface, for all entries on that interface:

o If the entry is CONFIRMED, it goes PENDING with the retry count set to 0.

o If the entry is PENDING, the retry count is incremented. If it reaches a maximum threshold, the entry goes ELAPSED If at least one entry is ELAPSED at the end of the process: if the Destroy timer is not running then it is armed with a jitter.

Since the DelayNA has a duration that decreases with the depth, it is expected to receive all NINOs from all children before the timer elapses and the full update is sent to the parent.

7.4. Other events

Finally, NINA listens to a series of events, such as:

o MR stopped or unable to run: NINA routes are cleaned up. NINA is inactive.

o NINA operation stopped: All entries in the abstract lists are freed. All the NINA routes are destroyed.

o Interface going down: for all entries in the Reachable list on that interface, the associated route is removed, and the entry is scheduled to be destroyed.

o Neighbor being removed from the ND list: if the entry is in the Reachable list the associated route is removed, and the entry is scheduled to be destroyed.

o Roaming: All entries in the Reachable list are set to not 'reported' and DelayNA is armed.

7.5. Default value

DEF_NA_LATENCY = 150 ms

MAX_DESTROY_INTERVAL = 200ms

8. Acknowledgments

We would like to thank all the people who have provided comments on this draft, specially to Ben McCarthy for his very helpful review of this document.

9. IANA Considerations

This document requires IANA to define a permanently-assigned multicast address: the MANEMO Mobile Routers Group (Section 6.1). Additionally, 2 new ND option types should be defined for NINO NA and NINO NS messages.

10. Security Considerations

The MANEMO Basic kiss (MR to MR over egress) scenario can be secured by SeND [13]. An MR can prove ownership of its prefix by the same certificate on egress as it could already on ingress.

The nested NEMO scenario has the same security concerns that ND/RFC 4191 without SeND. In order to secure this scenario, L2 trusts and policies are required.

11. References

11.1. Normative References

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

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

[3] Perkins, C., "IP Mobility Support for IPv4", RFC 3344, August 2002.

[4] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.

[5] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, January 2005.

[6] Thubert, P., "Nested Nemo Tree Discovery",

draft-thubert-tree-discovery-04 (work in progress), November 2006.

[7] Manner, J. and M. Kojo, "Mobility Related Terminology", RFC 3753, June 2004.

[8] Ernst, T. and H. Lach, "Network Mobility Support Terminology", draft-ietf-nemo-terminology-06 (work in progress), November 2006.

[9] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, November 2005.

[10] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

11.2. Informative References

[11] Corson, M. and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501, January 1999.

[12] Johnson, D., Hu, Y., and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4", RFC 4728, February 2007.

[13] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure

Neighbor Discovery (SEND)", RFC 3971, March 2005.

[14] Clausen, T., "The Optimized Link-State Routing Protocol version 2", draft-ietf-manet-olsrv2-02 (work in progress), June 2006.

[15] Clausen, T., "MANET Neighborhood Discovery Protocol (NHDP)", draft-ietf-manet-nhdp-01 (work in progress), February 2007.

[16] Thubert, P. and M. Molteni, "IPv6 Reverse Routing Header and its application to Mobile Networks", draft-thubert-nemo-reverse-routing-header-07 (work in progress), February 2007.

[17] Wakikawa, R., "MANEMO Problem Statement",

draft-wakikawa-manemo-problem-statement-00 (work in progress),
February 2007.

Authors' Addresses

Pascal Thubert
Cisco Systems
Village d'Entreprises Green Side
400, Avenue de Roumanille
Batiment T3

Biot - Sophia Antipolis 06410
FRANCE

Phone: +33 4 97 23 26 34
Email: pthubert@cisco.com

Ryuji Wakikawa
Keio University and WIDE
5322 Endo Fujisawa Kanagawa
252-8520
JAPAN

Email: ryuji@sfc.wide.ad.jp

Carlos J. Bernardos
Universidad Carlos III de Madrid
Av. Universidad, 30
Leganes, Madrid 28911
Spain

Phone: +34 91624 6236
Email: cjbc@it.uc3m.es

Roberto Baldessari
NEC Europe Network Laboratories
Kurfuersten-anlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342167
Email: roberto.baldessari@netlab.nec.de

Jean Lorchat
Keio University and WIDE
5322 Endo Fujisawa Kanagawa
252-8520
JAPAN

Email: lorchat@sfc.wide.ad.jp

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF
Administrative Support Activity (IASA).