6MAN
Internet-Draft
Intended status: Standards Track
Expires: June 13, 2011

P. Thubert, Ed.
Cisco Systems

December 10, 2010

Reverse Routing Header
draft-thubert-6man-reverse-routing-header-01

Abstract

For new classes of devices such as highly constrained nodes, forward and return Record Route capabilities are required to enable basic forwarding operations. This memo defines a such a technique for IPv6.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on June 13, 2011.

Copyright Notice

Table of Contents

1. Introduction

This document assumes that the reader is familiar with the IPv6 RH0 operation as specified in [RFC2460] and the RH2 as previously defined in [RFC3775] and the RH4 defined for RPL [I-D.ietf-roll-rpl] in [I-D.ietf-6man-rpl-routing-header]. It is more specifically targetted to address the needs of the RPL extensions defined in Reactive Discovery of Point-to-Point Routes [I-D.ietf-roll-p2p-rpl].

This specification defines a Forward Record Routing Header (FRRH), that is a controlled variant of the Loose Source and Record Route (LSRR) defined for IPv4 in [RFC0791] and hereby adapted for IPv6. FRRH records the path of a packet within a closed Source Routing Domain (SRD) such as a RPL network.

This specification also introduces a new Routing Header, called the Reverse Routing Header (RRH), to perform source routing within the RPL network along the way back. As opposed to the FRRH that records a forward path, RRH stacks the route bottom up and can be trivially converted into a RH4 to force packets to follow an identical reverse path within the same RPL network.

The FRRH and the RRH are designed to be trivially converted into a RH4 to force further packets to follow an identical path within the same RPL network, so the rules that govern the construction of a Routing Header type 4 in [I-D.ietf-6man-rpl-routing-header] also apply similarily to FRRH and RRH.

1.1. Motivations

A Low Power Lossy Network (LLN) often forms a dynamic NBMA Subnetwork of devices that might be so constrained in memory that they cannot hold all the states that would be required to route within their own Subnetwork. In some instances, default routes to some border routers can be maintained, but the way back to specific destination cannot. In other instances, even the route to the border router will be lost

rapidly.

RPL [I-D.ietf-roll-rpl] is a Subnetwork Gateway Protocol (SGP), that is a routing protocol that can build and maintain a routing topology within a subnet as well as distribute some subnet information. RPL is optimized for Point to Multipoint (P2MP) from a root and Multipoint to Point (MP2P) to a root of the LLN, but allows a stretch for any to any communication. [I-D.ietf-roll-p2p-rpl] extends RPL to establish on-demand an arbitrary Point to Point (P2P) path with lesser stretch and lower set up and repair latency than the base protocol. This specification allows to locate the additional states at the end-points so as to avoid extra in the intermediate nodes.

With strict source routing, the intermediate nodes find the next hop for a given packet in a routing header that is set by the source as specified for instance in [I-D.ietf-6man-rpl-routing-header] that defines the RH type 4 for IPv6. With this specification, the path information in the RH4 is maintained with consistent snapshots of the full path across the Source Route Domain that is recorded in-band with selected packets.

2. Terminology

This document assumes that the reader is familiar with ROLL terminology defined in [I-D.ietf-roll-terminology].

Additional terms are defined hereafter:

DAG Directed Acyclic Graph.

SRN Source Routing Node. A host or a router with the capability to support this specification and make use of RRH within its NBMA Subnetwork .

SRD Source Routing Domain. A domain in which the Source Route operation is accepted. All intermediate addresses within the same RH must belong to the same SRD. A domain can be:

* A node or a contiguous set of nodes such as a dominating set

* A Subnetwork such as a RPL Network

* A network serving the same Unique Local Aggregation.

SRA: Source Routable Address. An address that can be inserted in a RH/RRH. An SRA is an Ipv6 address that belongs to the SR domain with a scope that is valid across the SR domain.

TA: Target Address. The last Address in the Routing Header identifying the target. There is no constraint on that address.

CRH: Constrained Routing Header. A constrained routing header is a routing header that can be forwarded only within an SRD. It is formed of a list of SRA, followed by at most one TA.

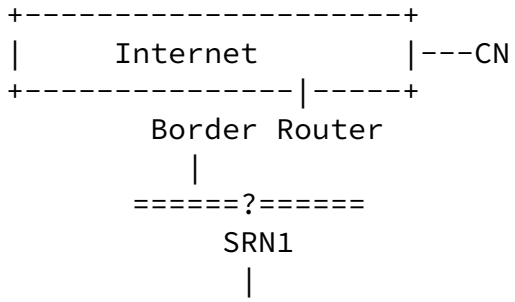
FRRH: Forward Record Routing Header, defined in this specification; a variable size record route header used to learn a path hop-by-hop. It is preferably formed of addresses that are located on the ingress interface of the packets.

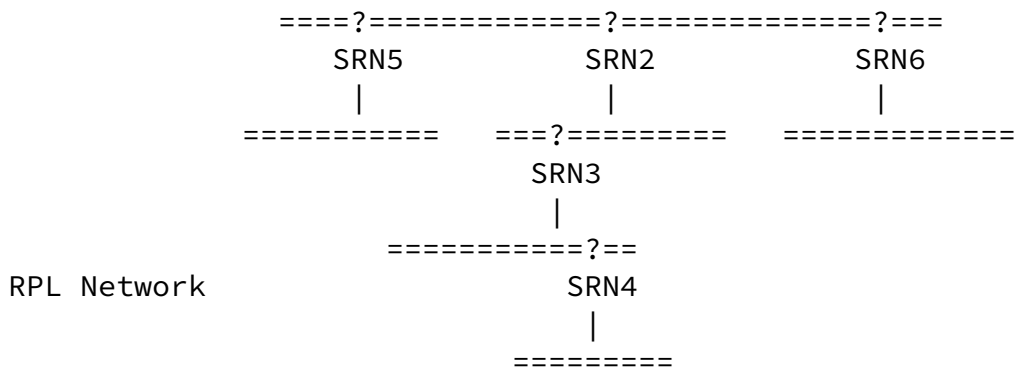
RRH: Reverse Routing Header, defined in this specification; a variable size reverse route header used to learn a path back hop-by-hop. It is formed of addresses that are located on the egress interface of the packets.

NULL RH: An FRRH or an RRH with a zero "Segments Used".

3. Examples

For the sake of the example, the RPL Network in the following figure assumes the logical shape of a tree towards a border router. This abstraction is chosen because it is simpler to represent than the actual Directed Acyclic Graph shape that most RPL Networks will form.





A tree shaped RPL network

This example focuses on a SRD node at depth 3 identified as Source Routing Node 3 (SRN3). The path to the border router and then the Internet is

SRN3 -> SRN2 -> SRN1 -> Border Router ->Internet

In one example, the Border Routers first initiates a multicast flooding to build a Reverse Routing Header that records the source route path from each node towards itself. In another example, a node that wishes to be reachable starts a record route towards the border router.

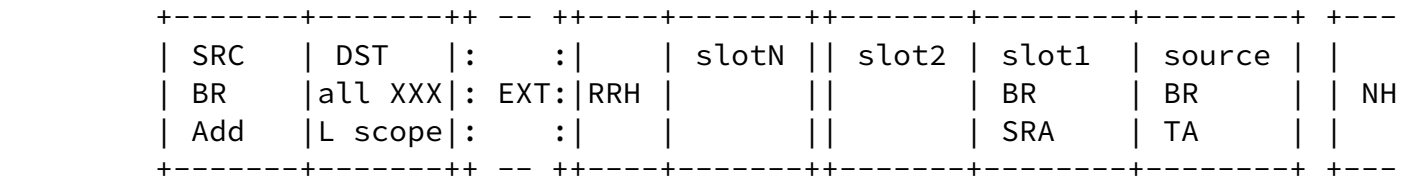
A node that wishes to be reachable inserts a reverse routing header with a number of N pre-allocated slots that derive from its estimation of its depth.

3.1. Flooding downwards

In this example, no preexisting routing structure exists and the routing header is being assembled by a flooding mechanism from the Border Router (BR) downwards.

The packet has source an IPv6 address of the Border Router Address, BR_Add, and destination a multicast link scope address that is used

for the flooding:



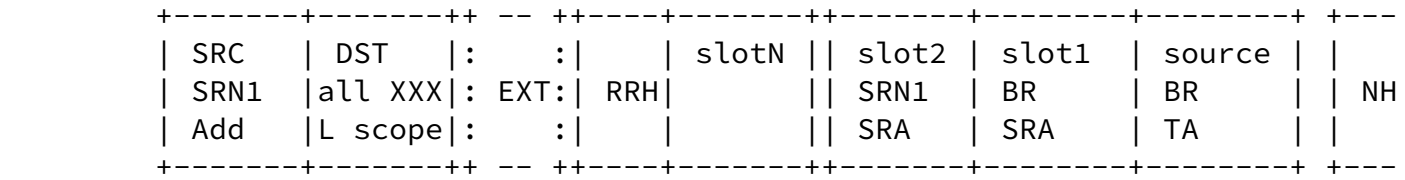
The BR-SRA acts as locator and it is possible that this address has a limited reach such as ULA. The BR-TA is a global identifier for the BR. It might be omitted when the source of the RRH is only used as an intermediate router and not as a destination.

It must be noted that BR-SRA is preferably an address on the BR interface towards SRN1, that is directly visible from SRN1, in case there is no routing between BR and SRN1.

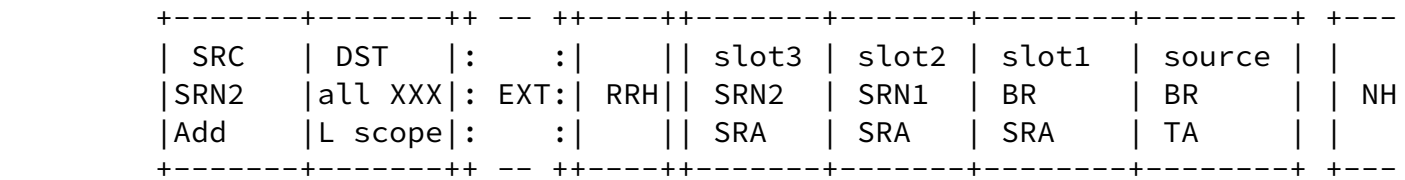
BR-SRA is also preferably an address that has a scope as large as the Source Route Domain, enabling the hop-by-hop recording process to possibly omit tracing some intermediate hops and thus form a loose source route header.

The routers one hop away figure the best message they receive and propagate it, including the augmented RRH.

For SRN1, this gives:



When SRN3 gets the packet, it receives:

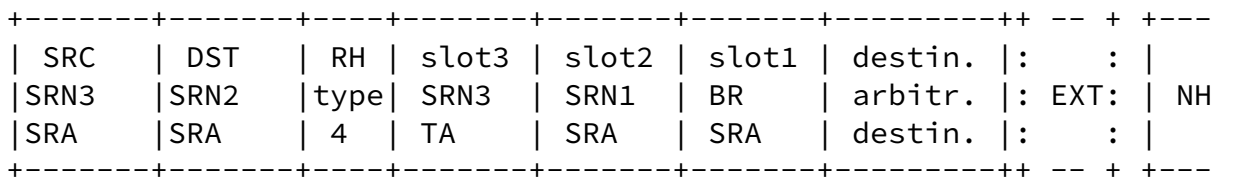


The RH4 is trivially built by picking the tail of the incoming RRH,

to be inserted when sending a packet to the border router. Additionally, the node might create a tunnel interface towards the border and install a default route there.

So an arbitrary destination in the Internet can replace the BR TA and will cause a packet flow like this:

transport mode (1 slot consumed if SRN3 TA is included)):

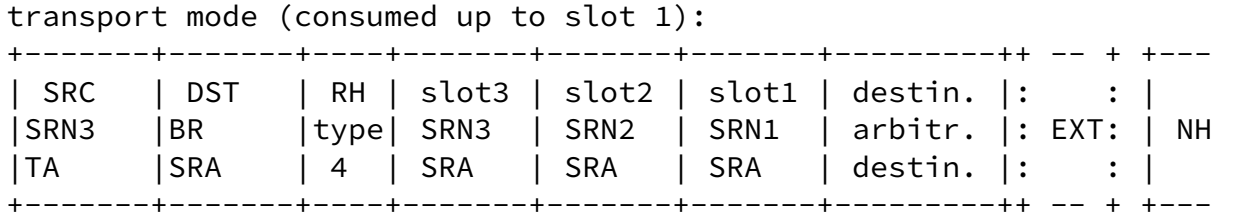


tunnel mode (nothing consumed):

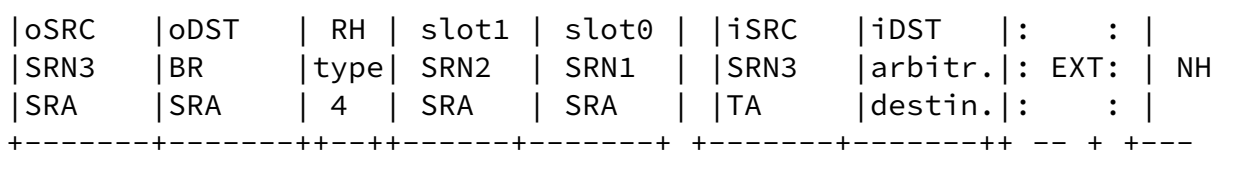
Message going out of SRN3 to the BR

That reaches the Border Router as this:

transport mode (consumed up to slot 1):



tunnel mode (all consumed):



Message going out of BR:

Message coming in the border router from SRN3

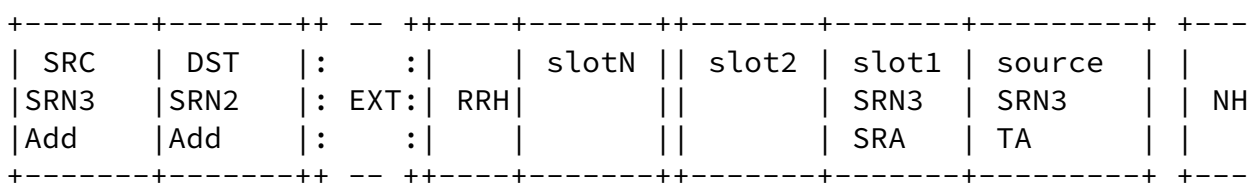
Upon decapsulation, it is up to the border router to decide by policy whether it should route the packet or not. In particular in Transport mode, security reasons might dictate to drop the packet.

3.2. Following an implicit upwards path

In this example, a preexisting routing structure exists that leads to a well-known border router. The RRH is assembled along that path.

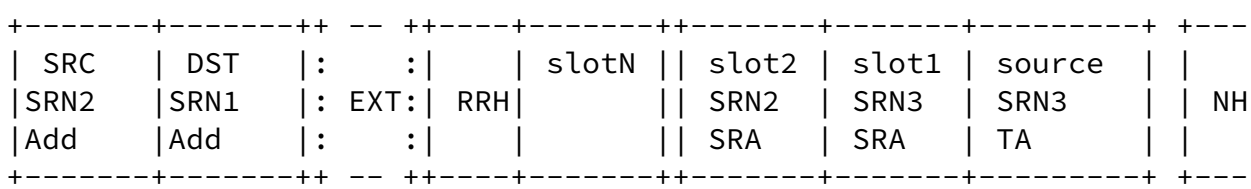
The last (bottom) slot contains a global identifier for the SRN, SRN3 TA. there is no constraint with regard to the type of IPv6 address used there. It might be omitted if SRN3 uses its SRA to terminate its connections. Then SRN3 inserts its SRA in the slot directly above.

The IPv6 header in the packet has source SRN3's Address, SRN3_Add, and destination SRN3's next hop Add, SRN2_Add, on the link between SRN2 and SRN3:

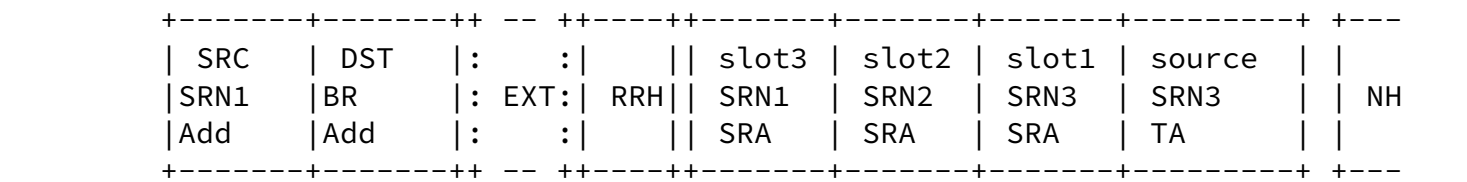


The second router on the path, SRN2, receives that the packet. If it is not the border router, then it might wish to propagate the protocol payload towards the border router that is the implicit termination of the propagation as dictated by the protocol operation.

The outer packet now has source SRN2 Add and destination SRN1 Add; the RRH from top to bottom is: empty_slots | SRN2_SRA | SRN3_SRA | SRN3_TA:



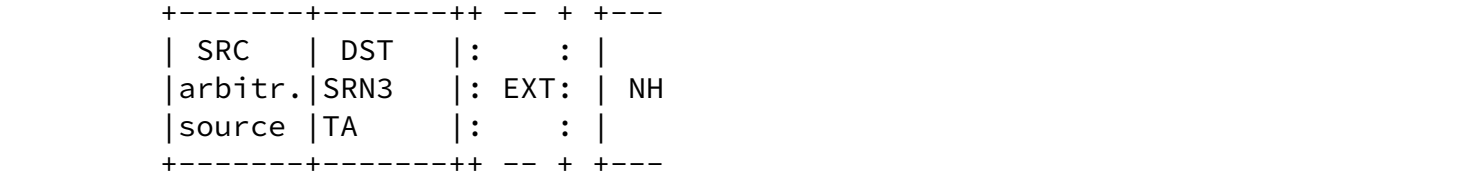
In general the process followed by the second router is repeated by all the routers on the path, till the border router that receives and absorbs:



When the border router, receives the packet, it MAY store the information in RRH to build an RH4 back to SRN3 TA (or SRN3 SRA if the TA is omitted)

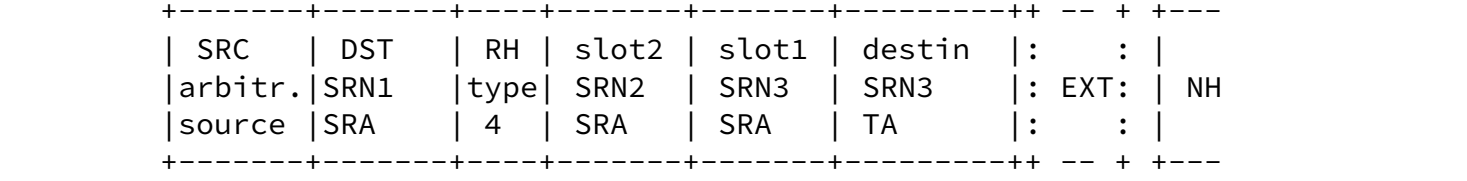
Again, the RH is trivially built by picking the trail of the previous RRH, to be inserted by the border router into any packet flowing down to SRN3:

Message coming in the border router from the infrastructure behind:

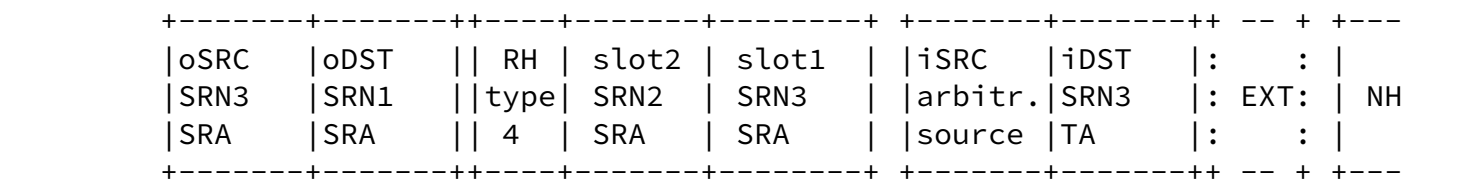


Message going out the border router:

transport mode:



Tunnel mode:



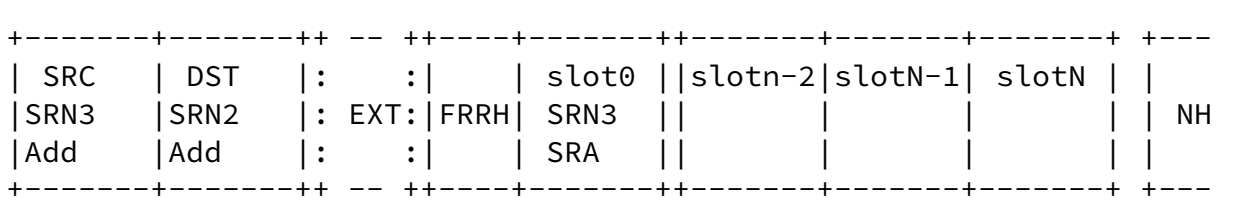
The RH type 4 is consumed along the source route path to SRN3 as a deprecated [RFC2460] RH type 0 would, and the last hop (SRN3 SRA to SRN3 TA) is consumed internally in SRN3, if it was present in the first place, like a RH type 2 would be in the case of Mobile IPv6 [RFC3775].

3.3. Recording a forward path

In this example, a Forward Record RH is filled as the protocol information is propagated along the same upwards path.

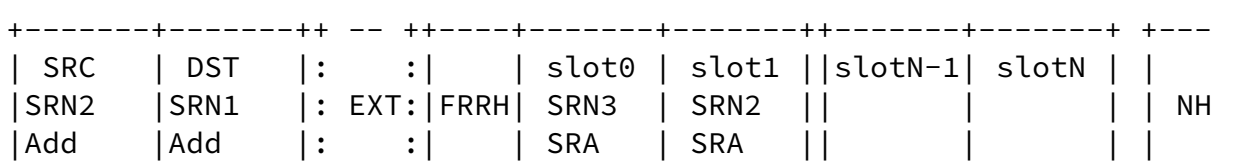
The FRRH is initially empty. The source SRN3 might virtually start from SRN3-TA in which case that address is added to the FRRH. Then, as any node along the path, SRN3 adds it SRA and passes the packet long.

The IPv6 header in the packet has source SRN3's Address, SRN3_Add, and destination SRN3's next hop Add, SRN2_Add, on the link between SRN2 and SRN3:



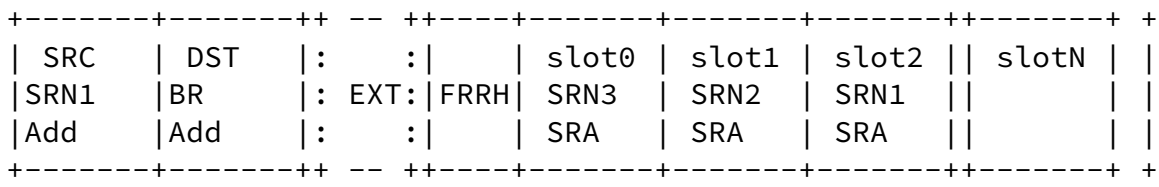
The second router on the path, SRN2, receives that the packet. Again, it might wish to propagate protocol payload towards the border router that is the implicit termination of the propagation.

The outer packet now has source SRN2 Add and destination SRN1 Add; the FRRH from top to bottom is SRN3_SRA | SRN2_SRA | empty_slots :

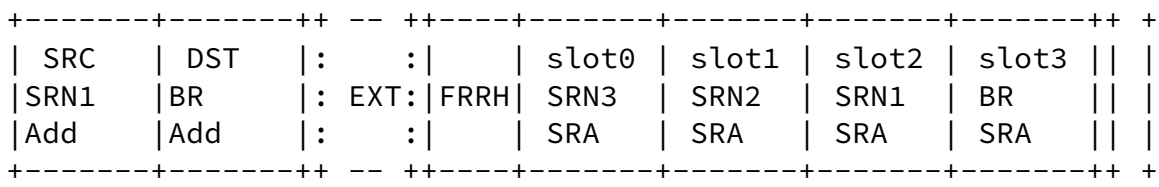


It must be noted that SRN2-SRA is preferably an address on the SRN2 ingress interface from SRN3, that is directly visible from SRN3, in case there is no routing between SRN3 and SRN.

In general the process followed by the second router is repeated by all the routers on the path, till the border router that receives:



The BR also adds its own information for the internal hop to BR_TA:



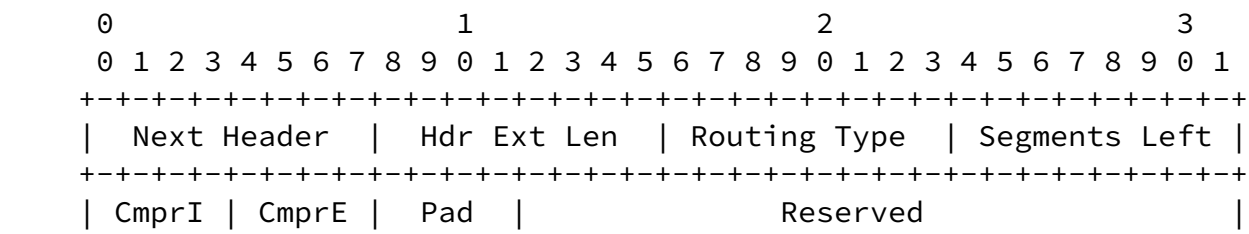
At this point, the BR possesses a source route path that is usable from any address along that path back to the BR. It may trivially transform the FRRH into a completed RRH and pass it back to SRN3. SRN3 may then transform the RRH into a RH type 4 and send further packets along the same path.

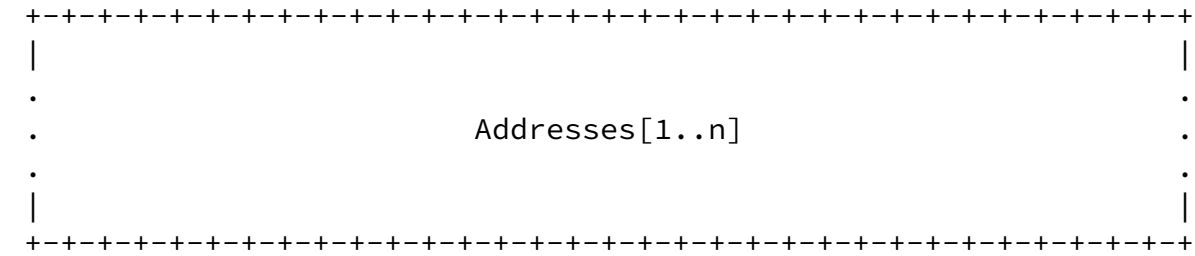
4. New Routing Headers

This draft introduces new loose source and record Constrained Route Headers for IPv6. The headers have the same format decribed below and only differ from the Routing type.

4.1. FRRH and RRH formats

The FRRH and the RRH share the same overall format as the RH4 as defined in [I-D.ietf-6man-rpl-routing-header], with the same contraints:





Next Header

8-bit selector. Identifies the type of header immediately following the Routing header. Uses the same values as the IPv4 Protocol field [RFC3232].

Hdr Ext Len

8-bit unsigned integer. Length of the Routing header in 8-octet units, not including the first 8 octets. Hdr Ext Len MUST NOT exceed RH4_MAX_SIZE / 8. Note that when Addresses[1..n] are compressed (i.e. value of CmprI or CmprE is not 0), Hdr Ext Len does not equal twice the number of Addresses.

Routing Type

8-bit unsigned integer. Set (tentatively) to 3 for FRRH and 5 for RRH.

Segments Left

8-bit unsigned integer. Number of route segments remaining.

CmprI

4-bit unsigned integer. Number of prefix octets from each segment, except than the last segment, that are elided. For example, a (F)RRH header carrying full IPv6 addresses in Addresses[1..n-1] sets CmprI to 0.

CmprE

4-bit unsigned integer. Number of prefix octets from the segment that are elided. For example, (F)RRH carrying a full IPv6 address in Addresses[n] sets CmprE to 0.

Pad

4-bit unsigned integer. Number of octets that are used to for padding after Address[n] and the end of the (F)RRH.

Reserved

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

Address slot []

Vector of 128-bit addresses, numbered 0 to N in LRRH and N to 0 in RRH.

4.2. Optimum number of slots

A SRN always initializes the number of slots in the F/RRH to the maximum of DEF_RRH_SLOTS and its estimation of its depth, if the latter is known from a reliable hint such as a routing protocol. The message may have a number of unused (NULL) slots, when it is received by the Border Router. The receiver end point crops out the extra entries in order to generates a RH.

From a RRH, the receiver generates a RH type 4 that it can use for a response back.

From a FRRH, the receiver generates a RRH that is fully consumed, and send that back to the sender which in turn will generate a RH type 4. None of those operations need to change the order of the slots in the header and are mostly plain copies.

The RH type 4 contains the number of required slots that the SRN now uses until it gets a hint that the topology changes or until the next route recording.

When a node adds its address either to an FRRH or an RRH, it MUST ensure that it owns none of the addresses that are already present in the packet. If it does, then the packet is following a loop. The node drop the packet, or alternatively it may strip the loop from the RH and keep forwarding via an alternate next hop. In that case, it will decrement the Hop limit as usual, to ensure that a loop is ultimately terminated.

The number of slots in the RRH MUST NOT be larger than MAX_RRH_SLOTS. If a SRN is deeper than MAX_RRH_SLOTS, it is expected that the rest of the way is already known ot the endpoint.

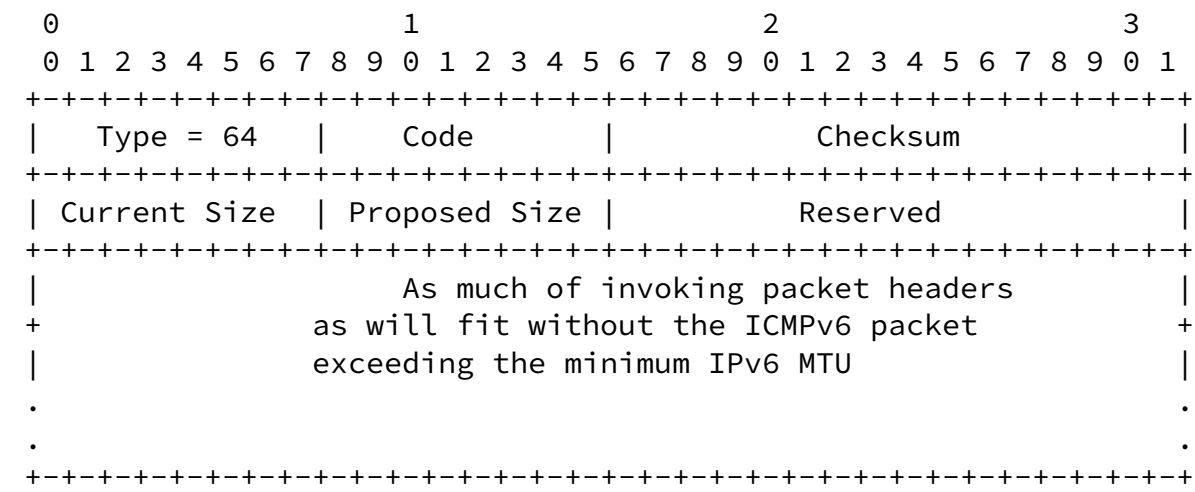
In runtime, it may happen that the RRH has fewer slots than required for the number of SRNs in the path because either the NBMA Subnetwork topology is changing too quickly, or the SRN that inserted the RRH had a wrong representation of the topology.

To solve this problem a new ICMP message is introduced, "RRH Warning", type (proposed) 64. A SRN on the upwards path that gets a packet without a free slot in the F/RRH MAY send that ICMP "RRH warning" back to the SRN that inserted the RRH in the first place.

This message allows a SRN on the path to propose a larger number of slots to the SRN that creates the RRH. The Proposed Size MUST NOT be larger than MAX_RRH_SLOTS. The originating SRN must rate-limit the ICMP messages to avoid excessive ICMP traffic in the case of the source failing to operate as requested.

The originating SRN must insert an RH type 4 based on the F/RRH in the associated IP header, in order to route the ICMP message back to the source of the reverse tunnel. A SRN that receives this ICMP message is the actual destination and it MUST NOT forward it to the source of the packet if the tunnel mode is being used.

The "RRH Warning" ICMP has the following format:



Type

64 [To Be Assigned]

Code 1: RRH too small; 2: Loop detected.

The originating SRN requires the source to set the RRH size to a larger value. The packet that triggered the ICMP will still be forwarded by the SRN, but the path cannot be totally optimized (see Section 5.3).

Checksum

The ICMP checksum [RFC2463].

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.

5. Source Routing Node Operation

5.1. Processing of ICMP "RRH Warning"

The New ICMP message "RRH Warning" is presented in Section 4.2. This message is addressed to the SRN which performs the tunnel encapsulation and generates the RRH.

Hence, a SRN that receives the ICMP "RRH Warning" MUST NOT propagate it to the originating SRN 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 4.2, then the SRN MAY adapt the number of slots to the Proposed Size.

5.2. Processing of ICMP error

When the SRN 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 SRN should transform it in a RH type 4 to forward the ICMP to the original source of the packet. If the included packet has an FRRH, then the SRN may reverse it into a RH type 4 to forward the ICMP to the original source of the packet.

5.3. Processing of RRH Packets

A router that receives a RRH is a link scoped protocol packet may save that RRH and associate it with the propagation of the protocol information. the router performs ULP checksum validation and security header checks including the RRH as received

When the router sends the propagated protocol information over an interface, the router adds one of its addresses from that interface at the head of the RRH, and then computes upper layer checksums and IPSec/AH signatures as required.

It is preferred that the address as a scope that is as large as the

Source Route Domain, in order to enable a loose operation. In particular, if the router has consistent states to route to the seconds most recent entry via the source address of the packet, then it can overwrite the most recent entry with its own.

The node at the end of the propagation and any node on the way may decide to keep a source route state towards the address located in slot 0 using a source route path that is directly inferred from the

RRH.

5.4. Processing of FRRH Packets

A router that receives a FRRH is a link scoped protocol packet may save that RRH and associate it with the propagation of the protocol information. the router performs ULP checksum validation and security header checks including the FRRH as received.

Then the router adds an address from the ingress interface at the end of the FRRH, which is now ready to be associated to the propagation of the protocol. When the router sends the propagated protocol information over an interface, it adds the FRRH as and computes upper layer checksums and IPSec/AH signatures as required.

The node at the end of the propagation and any node on the way may decide to reverse the FRRH into a RRH and send it back to the source located in slot 0 for the FRRH, which in turn can reverse it again, this time into a RH type 4.

6. Security Considerations

The FRRH and the RRH are propagated as part of a higher hop-by-hop protocol operation, so it is not mutable. Each hop adds its info, then computes the checksum and IPSec headers and then it transmits with a link scope to the next node(s) on the way of the upper layer protocol operation.

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

7. IANA considerations

This document requires IANA to define 2 new IPv6 Routing Header types for Forward Record Routing Header and Reverse Routing Header. The allocation is governed by [I-D.ietf-6man-iana-routing-header] The desired values would be 3 for FRRH and 5 for RRH.

This document also requires the allocation of a new ICMP error type "RRH Warning" with a proposed value of 64.

8. Protocol Constants

DEF_RRH_SLOTS: 7

MAX_RRH_SLOTS: 10

9. Acknowledgements

The author wishes to thank Mukul Goyal and Emmanuel Baccelli for their contributions and reviews. Also Jonathan Hui, JP Vasseur, Dave Culler and Vishwas Manral for their work on RH 4 from which this

works inherits.

10. References

10.1. informative reference

[I-D.ietf-6man-iana-routing-header]
Arkko, J. and S. Bradner, "IANA Allocation Guidelines for the IPv6 Routing Header", draft-ietf-6man-iana-routing-header-00 (work in progress), October 2009.

[I-D.ietf-6man-rpl-routing-header]
Hui, J., Vasseur, J., Culler, D., and V. Manral, "An IPv6 Routing Header for Source Routes with RPL", draft-ietf-6man-rpl-routing-header-01 (work in progress), October 2010.

[I-D.ietf-roll-p2p-rpl]
Goyal, M. and E. Baccelli, "Reactive Discovery of Point-to-Point Routes in Low Power and Lossy Networks", draft-ietf-roll-p2p-rpl-01 (work in progress), October 2010.

[I-D.ietf-roll-rpl]
Winter, T., Thubert, P., Brandt, A., Clausen, T., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., and J. Vasseur, "RPL: IPv6 Routing Protocol for Low power and Lossy Networks", draft-ietf-roll-rpl-16 (work in progress), December 2010.

[I-D.ietf-roll-terminology]
Vasseur, J., "Terminology in Low power And Lossy Networks", draft-ietf-roll-terminology-04 (work in progress), September 2010.

10.2. normative reference

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

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

[RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[RFC2402] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[RFC2406] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, November 1998.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2463] Conta, A. and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 2463, December 1998.

[RFC3232] Reynolds, J., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, January 2002.

[RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

Author's Address

Pascal Thubert (editor)
Cisco Systems
ViAddge d'Entreprises Green Side
400, Avenue de Roumanille
Batiment T3
Biot - Sophia Antipolis 06410
FRANCE

Phone: +33 497 23 26 34
Email: pthubert@cisco.com