

ROLL Working Group
Internet-Draft
Intended status: Informational
Expires: April 21, 2016

M. Robles
Ericsson
M. Richardson
SSW
P. Thubert
Cisco
October 19, 2015

**When to use [RFC 6553](#), 6554 and IPv6-in-IPv6
draft-robles-roll-useofrplinfo-02**

Abstract

This document states different cases where [RFC 6553](#), [RFC 6554](#) and IPv6-in-IPv6 encapsulation is required to set the bases to help defining the compression of RPL routing information in LLN environments.

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 April 21, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4](#).e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Terminology and Requirements Language	3
3.	Sample/reference topology	3
4.	Use cases	6
5.	Storing mode	8
5.1.	Example of Flow from RPL-aware-leaf to root	8
5.2.	Example of Flow from root to RPL-aware-leaf	9
5.3.	Example of Flow from root to not-RPL-aware-leaf	10
5.4.	Example of Flow from not-RPL-aware-leaf to root	10
5.5.	Example of Flow from RPL-aware-leaf to Internet	11
5.6.	Example of Flow from Internet to RPL-aware-leaf	11
5.7.	Example of Flow from not-RPL-aware-leaf to Internet	12
5.8.	Example of Flow from Internet to not-RPL-aware-leaf	13
5.9.	Example of Flow from RPL-aware-leaf to RPL-aware-leaf	14
5.10.	Example of Flow from RPL-aware-leaf to non-RPL-aware-leaf	15
5.11.	Example of Flow from not-RPL-aware-leaf to RPL-aware-leaf	16
5.12.	Example of Flow from not-RPL-aware-leaf to not-RPL-aware-leaf	17
6.	Non Storing mode	18
6.1.	Example of Flow from RPL-aware-leaf to root	18
6.2.	Example of Flow from root to RPL-aware-leaf	19
6.3.	Example of Flow from root to not-RPL-aware-leaf	19
6.4.	Example of Flow from not-RPL-aware-leaf to root	20
6.5.	Example of Flow from RPL-aware-leaf to Internet	20
6.6.	Example of Flow from Internet to RPL-aware-leaf	21
6.7.	Example of Flow from not-RPL-aware-leaf to Internet	22
6.8.	Example of Flow from Internet to not-RPL-aware-leaf	23
6.9.	Example of Flow from RPL-aware-leaf to RPL-aware-leaf	24
6.10.	Example of Flow from RPL-aware-leaf to not-RPL-aware-leaf	25
6.11.	Example of Flow from not-RPL-aware-leaf to RPL-aware-leaf	26
6.12.	Example of Flow from not-RPL-aware-leaf to not-RPL-aware-leaf	26
7.	Future RPL work	27
8.	IANA Considerations	27
9.	Security Considerations	27
10.	Acknowledgments	27
11.	References	28
11.1.	Normative References	28
11.2.	Informative References	28
	Authors' Addresses	29

1. Introduction

RPL [[RFC6550](#)] is a routing protocol for constrained networks. [RFC 6553](#) [[RFC6553](#)] defines the "RPL option", carried within the IPv6 Hop-by-Hop header to quickly identify inconsistencies in the routing topology. [RFC 6554](#) [[RFC6554](#)] defines the "RPL Source Route Header", an IPv6 Extension Header to deliver datagrams within a RPL routing domain.

Several discussions in the ROLL/6lo/6TiSCH Mailing Lists took place focusing in the definition of how to compress RPL Information in constrained environment. ROLL Virtual Interim Meeting (02-2015) concluded that there is a need to define how to use [[RFC6553](#)], [[RFC6554](#)] and IPv6-in-IPv6 encapsulation to be able to set the correct environment for compression. A Routing Header Dispatch for 6LoWPAN (6LoRH) [[I-D.thubert-6lo-routing-dispatch](#)] defines a method to compress RPL Option information and Routing Header type 3 ([RFC6554](#)) and an efficient IP-in-IP technique.

This document is going to be focused in data plane messages and how can be transmitted within the above mentioned RFCs.

2. Terminology and Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

Terminology defined in [[RFC7102](#)]

3. Sample/reference topology

A RPL network is composed of a 6LBR (6LoWPAN Border Router), Backbone Router (6BBR), 6LR (6LoWPAN Router) and 6LN (6LoWPAN Node) as leaf logically organized in a DODAG structure (Destination Oriented Directed Acyclic Graph).

RPL defines the RPL Control messages (control plane), a new ICMPv6 message with Type 155. DIS, DIO and DAO messages are all RPL Control messages but with different Code values.

RPL supports two modes of Downward traffic: in storing mode, it is fully stateful or an in non-storing, it is fully source routed. A RPL Instance is either fully storing or fully non-storing, i.e. a RPL Instance with a combination of storing and non-storing nodes is not supported with the current specifications.

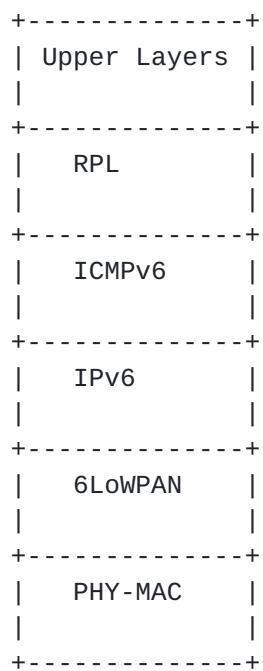


Figure 1: RPL Stack.

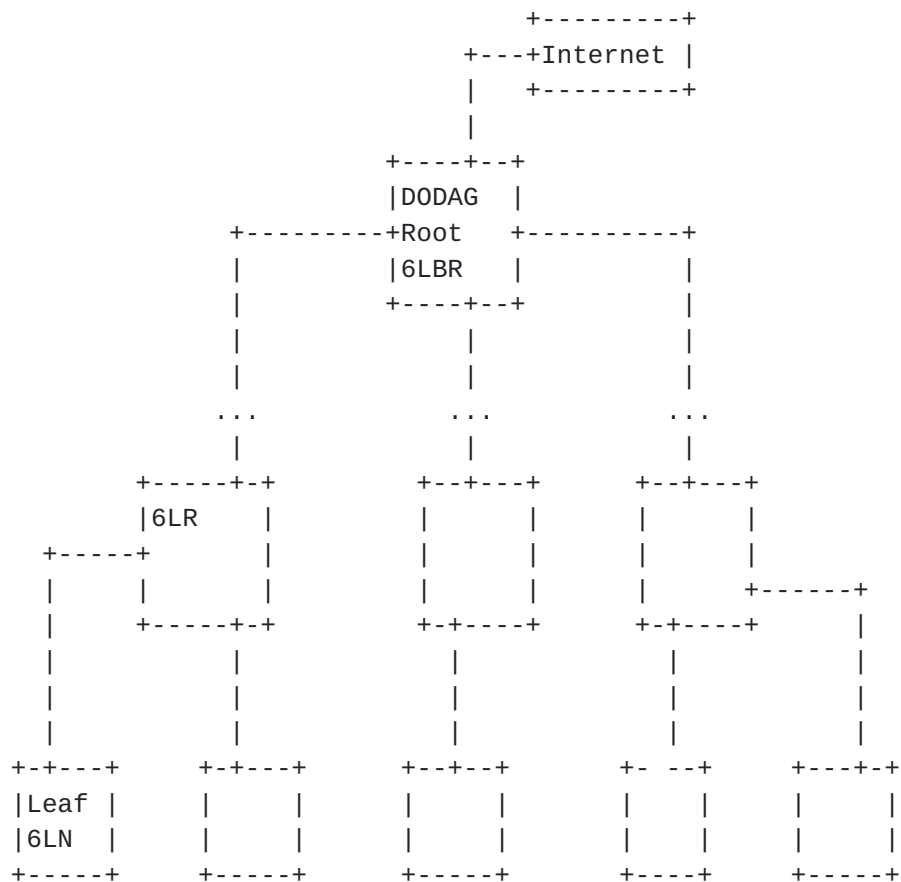


Figure 2: A reference RPL Topology.

This document is in part motivated by the work that is ongoing at the 6TiSCH working group. The 6TiSCH architecture [\[I-D.ietf-6tisch-architecture\]](#) draft explains the network architecture of a 6TiSCH network. This architecture is used for the remainder of this document.

The scope of the 6TiSCH Architecture is a Backbone Link that federates multiple LLNs (mesh) as a single IPv6 Multi-Link Subnet. Each LLN in the subnet is anchored at a Backbone Router (6BBR). The Backbone Routers interconnect the LLNs over the Backbone Link and emulate that the LLN nodes are present on the Backbone thus creating a so-called: Multi-Link Subnet. An LLN node can move freely from an LLN anchored at a Backbone Router to another LLN anchored at the same or a different Backbone Router inside the Multi-Link Subnet and conserve its addresses.

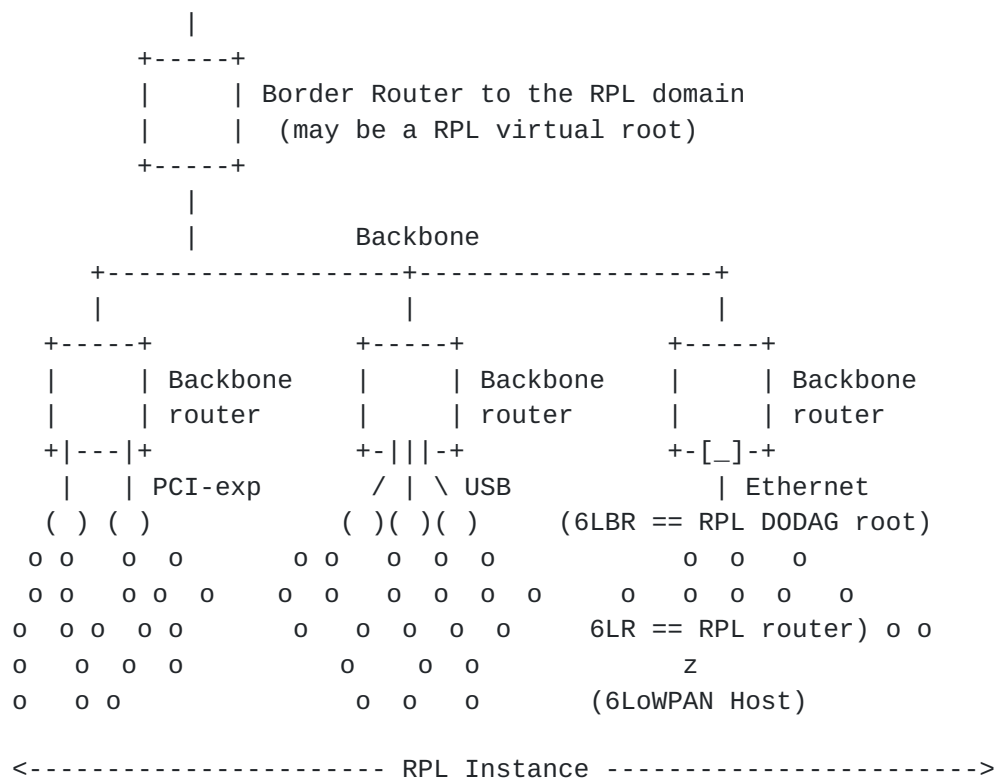


Figure 3: RPL domain architecture

4. Use cases

In data plane context a combination of [RFC6553](#), [RFC6554](#) and IPv6-in-IPv6 encapsulation is going to be analyzed for the following traffic flows:

- Flow from RPL-aware-leaf to root
- Flow from root to RPL-aware-leaf
- Flow from not-RPL-aware-leaf to root
- Flow from root to not-RPL-aware-leaf
- Flow from RPL-aware-leaf to Internet
- Flow from Internet to RPL-aware-leaf
- Flow from not-RPL-aware-leaf to Internet
- Flow from Internet to not-RPL-aware-leaf

- Flow from RPL-aware-leaf to RPL-aware-leaf
- Flow from RPL-aware-leaf to not-RPL-aware-leaf
- Flow from not-RPL-aware-leaf to RPL-aware-leaf
- Flow from not-RPL-aware-leaf to not-RPL-aware-leaf

This document assumes a rule that a Header cannot be inserted or removed on the fly inside an IPv6 packet that is being routed.

- This means that an intermediate router that needs to add a header must encapsulate the packet in an outer IP header where the new header can be placed.

- This also means that a Header can only be removed by an intermediate router if it is placed in an encapsulating IPv6 Header, and in that case, the whole encapsulating header must be removed - a replacement may be added.

This document recognizes that some headers such as a Routing Header or a Hop-by-Hop header may be modified by routers on the path of the packet without the need to add to remove an encapsulating header.

The RPL RH and the RPL option are mutable but recoverable .

RPI should be present in every single RPL data packet. There is an exception in non-storing mode, when a packet is going down from the route: the entire route is written, so there are no loops of confusion about which table to use (purpose of instanceID).

The applicability for storing (RPL-SN) and non-Storing (RPL-NSN) modes for the previous cases is showed as follows:

Use Case	RPL-SN RPI (RFC 6553)	RPL-SN RH3 (RFC 6554)	RPL-SN IP-in-IP	RPL-NSN RPI	RPL-NSN RH3	RPL-NSN IP-in-IP
RPL-aware-leaf to root	Yes	No	No	Yes	No	No
root to RPL-aware-leaf	Yes	No	No	Yes	Yes	No
not-RPL-aware-leaf to	Yes	No	Yes	Yes	No	Yes

root							
root to not-	Yes	No	Yes	Yes	Yes	Yes	
RPL-aware-							
leaf							
RPL-aware-	Yes	No	Yes	Yes	No	Yes	
leaf to							
Internet							
Internet to	Yes	No	Yes	Yes	Yes	Yes	
RPL-aware-							
leaf							
not-RPL-	Yes	No	Yes	Yes	No	Yes	
aware-leaf to							
Internet							
Internet to	Yes	No	Yes	Yes	Yes	Yes	
not-RPL-							
aware-leaf							
RPL-aware-	Yes	No	No	Yes	Yes	Yes	
leaf to RPL-							
aware-leaf							
RPL-aware-	Yes	No	Yes	Yes	Yes	Yes	
leaf to not-							
RPL-aware-							
leaf							
not-RPL-	Yes	No	Yes	Yes	Yes	Yes	
aware-leaf to							
RPL-aware-							
leaf							
not-RPL-	Yes	No	Yes	Yes	Yes	Yes	
aware-leaf to							
not-RPL-							
aware-leaf							
+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+

Table 1: Possibility to transmit in Storing or Non-Storing mode: RPI, RH3, IP-in-IP encapsulation

5. Storing mode

5.1. Example of Flow from RPL-aware-leaf to root

As states in [Section 16.2 of \[RFC6550\]](#) a RPL-aware-leaf node does not generally issue DIO messages, a leaf node accepts DIO messages (In inconsistency a leaf node generates DIO with infinite rank, to fix it). It may issue DAO and DIS messages though it generally ignores DAO and DIS messages.

In storing mode is suitable the use of [RFC 6553](#) to send RPL Information through HBH field checking the routing table to find out where to send the message.

In this case the flow comprises:

RPL-aware-leaf (6LN) --> 6LR --> 6LR,... --> root (6LBR) Note: In this document 6LRs, 6LBR are always full-fledge RPL routers

The 6LN inserts the RPI header, and send the packet to 6LR which decrement the rank in RPI and send the packet up. When the packet arrives to 6LBR, the RPI is removed and the packet is processed.

Header	6LN	6LR	6LBR
Inserted headers	RPI	--	--
Removed headers	--	--	RPI
Re-added headers	--	--	--
Modified headers	--	RPI	--
Untouched headers	--	--	--

Storing: Summary of the use of headers from RPL-aware-leaf to root

5.2. Example of Flow from root to RPL-aware-leaf

In this case the flow comprises:

root (6LBR)--> 6LR --> RPL-aware-leaf (6LN)

In this case the 6LBR insert RPI header and send the packet down, the 6LR is going to increment the rank in RPI (examines instanceID for multiple tables), the packet is processed in 6LN and RPI removed.

Header	6LBR	6LR	6LN
Inserted headers	RPI	--	--
Removed headers	--	--	RPI
Re-added headers	--	--	--
Modified headers	--	RPI	--
Untouched headers	--	--	--

Storing: Summary of the use of headers from root to RPL-aware-leaf

5.3. Example of Flow from root to not-RPL-aware-leaf

In this case the flow comprises:

root (6LBR)--> 6LR --> not-RPL-aware-leaf (6LN)

It includes IPv6-in-IPv6 encapsulation to transmit information not related with the RPL domain. In the 6LBR the RPI header is inserted into an IPv6-in-IPv6 header addressed to the last 6LR, which removes the header before pass the packet to the IPv6 node.

Header	6LBR	6LR	IPv6
Inserted headers	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--
Modified headers	--	--	--
Untouched headers	--	--	--

Storing: Summary of the use of headers from root to not-RPL-aware-leaf

5.4. Example of Flow from not-RPL-aware-leaf to root

In this case the flow comprises:

not-RPL-aware-leaf (6LN) --> 6LR --> root (6LBR)

When the packet arrives from IPv6 node to 6LR. This router insert the RPI encapsuladed in a IPv6-in-IPv6 header addressed to the root. The root removes the header and process the packet

Header	IPv6	6LR	6LBR
Inserted headers	--	IPv6-in-IPv6(RPI)	--
Removed headers	--	--	IPv6-in-IPv6(RPI)
Re-added headers	--	--	--
Modified headers	--	--	--
Untouched headers	--	--	--

Storing: Summary of the use of headers from not-RPL-aware-leaf to root

5.5. Example of Flow from RPL-aware-leaf to Internet

RPL information from [RFC 6553](#) should not go out to Internet. The router should take this information out before send the packet to Internet. The HBH Option is going to be analyzed in each node to the root.

In this case the flow comprises:

RPL-aware-leaf (6LN) --> 6LR --> root (6LBR) --> Internet

6LN insert RPI in a IPv6-in-IPv6 in a outer header, and send the packet to 6LR, which modified the rank in the RPI. When the packet arrives to 6LBR, the RPI is removed.

Header	6LN	6LR	6LBR	Internet
Inserted headers	IPv6-in-IPv6(RPI)	--	--	--
Removed headers	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	RPI	--	--
Untouched headers	--	--	--	--

Storing: Summary of the use of headers from RPL-aware-leaf to Internet

5.6. Example of Flow from Internet to RPL-aware-leaf

In this case the flow comprises:

Internet --> root (6LBR) --> 6LR --> RPL-aware-leaf (6LN)

When the packet arrives from Internet to 6LBR the RPI header is added in a outer IPv6-in-IPv6 header and send to 6LR, which modifies the rank in the RPI. When the packet arrives 6LN the RPI header is removed and the packet processed.

Header	Internet	6LBR	6LR	6LN
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	--	--	IPv6-in-IPv6(RPI)
Re-added headers	--	--	--	--
Modified headers	--	--	RPI	--
Untouched headers	--	--	--	--

Storing: Summary of the use of headers from Internet to RPL-aware-leaf

[5.7.](#) Example of Flow from not-RPL-aware-leaf to Internet

In this case the flow comprises:

not-RPL-aware-leaf (6LN) = IPv6 node --> 6LR --> root (6LBR) --> Internet

In the IPv6 node the flow label is assumed to be zero, the packet is transmitted to 6LR which encapsulate the RPI header in an outer IPv6-in-IPv6 header and send to 6LBR, which removes this header and send the packet to Internet and might set the flow label field.

Header	IPv6	6LR	6LBR	Internet
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Storing: Summary of the use of headers from not-RPL-aware-leaf to Internet

5.8. Example of Flow from Internet to not-RPL-aware-leaf

In this case the flow comprises:

Internet --> root (6LBR) --> 6LR --> not-RPL-aware-leaf (6LN)

6LBR get the packet from Internet and add a RPI header encapsulated in a IPv6-in-IPv6 header addressed to 6LR and send the packet down. The flow label is set to zero on inner IP. The last 6LR removes the RPI header. The IPv6 node might set the flow label since may arrive with zero value.

Header	Internet	6LBR	6LR	IPv6
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Storing: Summary of the use of headers from Internet to not-RPL-aware-leaf

5.9. Example of Flow from RPL-aware-leaf to RPL-aware-leaf

In [RFC6550] RPL allows a simple one-hop P2P optimization for both storing and non-storing networks. A node may send a P2P packet destined to a one-hop neighbor directly to that node. [Section 9 in \[RFC6550\]](#).

In this case the flow comprises:

6LN --> 6LR --> common parent (6LR) --> 6LR --> 6LN

This case is assumed in the same RPL Domain. In the common parent, the direction of RPI is changed (from increasing to decreasing the rank).

Header	6LN src	6LR	6LR (common parent)	6LR	6LN dst
Inserted headers	RPI	--	--	--	--
Removed headers	--	--	--	--	RPI
Re-added headers	--	--	--	--	--
Modified headers	--	RPI (decreasing rank)	RPI (increasing rank)	--	--
Untouched headers	--	--	--	--	--

Storing: Summary of the use of headers for RPL-aware-leaf to RPL-aware-leaf

5.10. Example of Flow from RPL-aware-leaf to non-RPL-aware-leaf

In this case the flow comprises:

6LN --> 6LR --> common parent (6LR) --> 6LR --> not-RPL-aware 6LN

Somehow, the sender has to know that the receiver is not RPL aware, and needs to know 6LR, and not even the root knows where the 6LR is (in storing mode). This case FAILS.

Header	6LN	6LR	6LR (common parent)	6LR	IPv6
Inserted header	IPv6-in-IPv6(RPI)	--	--	--	--
Removed headers	--	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--	--
Modified header	--	IPv6-in-IPv6(RPI)	IPv6-in-IPv6(RPI)	--	--
Untouched headers	--	--	--	--	--

Storing: Summary of the use of headers from RPL-aware-leaf to not-RPL-aware-leaf

[5.11.](#) Example of Flow from not-RPL-aware-leaf to RPL-aware-leaf

In this case the flow comprises:

not-RPL-aware 6LN --> 6LR --> common parent (6LR) --> 6LR --> 6LN

The 6LR that get the packet from IPv6 node, insert the RPI header encapsulated in IPv6-in-IPv6 header with destination to 6LN, the common parent change the direction of RPI and finally it is removed by 6LN.

Header	IP v6	6LR	common parent (6LR)	6LR	6LN
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--	--
Removed headers	--	--	--	--	IPv6-in-IPv6(RPI)
Re-added headers	--	--	--	--	--
Modified headers	--	--	IPv6-in-IPv6(RPI)	IPv6-in-IPv6(RPI)	--
Untouched headers	--	--	--	--	--

Storing: Summary of the use of headers from not-RPL-aware-leaf to RPL-aware-leaf

5.12. Example of Flow from not-RPL-aware-leaf to not-RPL-aware-leaf

In this case the flow comprises:

not-RPL-aware 6LN --> 6LR --> root (6LBR) --> 6LR --> not-RPL-aware 6LN

The problem to solve is how to indicate where to send the packet when get into LLN.

Header	IPv6	6LR	6LR	6LR	IPv6
	6		(common		dst
	src		parent)		
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--	--
Removed headers	--	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--	--
Modified headers	--	--	--	--	--
Untouched headers	--	--	--	--	--

Storing: Summary of the use of headers from not-RPL-aware-leaf to not-RPL-aware-leaf

6. Non Storing mode

6.1. Example of Flow from RPL-aware-leaf to root

In non-storing mode the leaf node uses Hop-By-Hop option ([RFC 6553](#)) to indicate the routing information to send messages to the DODAG root, this message is going to be analyzed in each node until arrive the DODAG root.

In this case not need to use IPv6-in-IPv6 because no header is not going to be removed, neither RH3, the flow comprises:

RPL-aware-leaf (6LN) --> 6LR --> root (6LBR)

This case is the same case as storing mode.

Header	6LN	6LR	6LBR
Inserted headers	RPI	--	--
Removed headers	--	--	RPI
Re-added headers	--	--	--
Modified headers	--	RPI	--
Untouched headers	--	--	--

Non Storing: Summary of the use of headers from RPL-aware-leaf to root

6.2. Example of Flow from root to RPL-aware-leaf

In this case the flow comprises:

root (6LBR)--> 6LR --> RPL-aware-leaf (6LN)

6LBR might instert RPI header, and the rute is indicated in RH3. 6LR updated RH3 and 6LN remove these headers.

Header	6LBR	6LR	6LN
Inserted headers	(optional: RPI), RH3	--	--
Removed headers	--	--	RH3,RPI
Re-added headers	--	--	--
Modified headers	--	RH3	--
Untouched headers	--	--	--

Non Storing: Summary of the use of headers from root to RPL-aware-leaf

6.3. Example of Flow from root to not-RPL-aware-leaf

In this case the flow comprises:

root (6LBR)--> 6LR --> not-RPL-aware-leaf (IPv6 node)

In 6LBR the RH3 is added, and modified in 6LR where is fully consumed, but left there. If the RPI is present, the IPv6 node which does not understand it will drop it. To avoid it the RPI should be removed before reach IPv6 node or it is recommended that RPI be omitted. An IPv6-in-IPv6 header should be necessary in this case. The DAO from 6LR about IPv6 could say if that the final IPv6 is not RPL (RPI) capable.

Header	6LBR	6LR	IPv6
Inserted headers	RH3	--	--
Removed headers	--	--	--
Re-added headers	--	--	--
Modified headers	--	RH3	--
Untouched headers	--	--	--

Non Storing: Summary of the use of headers from root to not-RPL-aware-leaf

6.4. Example of Flow from not-RPL-aware-leaf to root

In this case the flow comprises:

IPv6-node --> 6LR1 --> 6LR2 --> root (6LBR)

In this case the RPI is encapsulated in the first 6LR, and is not modified in the followings 6LRs.

Header	IPv6	6LR1	6LR2	6LBR
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	--	--	IPv6-in-IPv6(RPI)
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	IPv6-in-IPv6(RPI)	--	--

Non Storing: Summary of the use of headers from not-RPL-aware-leaf to root

6.5. Example of Flow from RPL-aware-leaf to Internet

In this case the flow comprises:

RPL-aware-leaf (6LN) --> 6LR --> root (6LBR) --> Internet

This case requires that the network is awareness of what is external to the LLN. Internet node never sees RPI or IPv6-in-IPv6 header. In

the 6LBR the flow label is computed if it is zero. RPI remains unmodified.

Header	6LN	6LR	6LBR	Internet
Inserted headers	IPV6-in-IPv6(RPI)	--	--	--
Removed headers	--	--	IPV6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	RPI	--	--

Non Storing: Summary of the use of headers from RPL-aware-leaf to Internet

[6.6.](#) Example of Flow from Internet to RPL-aware-leaf

In this case the flow comprises:

Internet --> root (6LBR) --> 6LR --> RPL-aware-leaf (6LN)

If the last RH3 entry is the 6LR, then the IPV6-in-IPv6 will be removed there, if the last entry is the 6LN, then the RH3 will go all the way to the leaf. In 6LBR the flow label should be set to zero.

Header	Internet	6LBR	6LR	6LN
Inserted headers	--	IPv6-in-IPv6(RH3, optional:RPI)	--	--
Removed headers	--	--	IPv6-in-IPv6 can be removed if RH3 consumed	--
Re-added headers	--	--	--	--
Modified headers	--	--	IPv6-in-IPv6(RH3)	--
Untouched headers	--	--	--	--

Non Storing: Summary of the use of headers from Internet to RPL-aware-leaf

[6.7.](#) Example of Flow from not-RPL-aware-leaf to Internet

In this case the flow comprises:

not-RPL-aware-leaf (6LN) --> 6LR --> root (6LBR) --> Internet

In this case the flow label is recommended to be zero in the IPv6 node. no RPL headers are added in the IPv6 node, since it is ignorant of RPL. Internet node does not see special headers. In 6LBR the flow label is computed if it is zero.

Header	IPv6	6LR	6LBR	Internet
Inserted headers	--	IPv6-in-IPv6(RPI)	--	--
Removed headers	--	--	IPv6-in-IPv6(RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Non Storing: Summary of the use of headers from not-RPL-aware-leaf to Internet

[6.8.](#) Example of Flow from Internet to not-RPL-aware-leaf

In this case the flow comprises:

Internet --> root (6LBR) --> 6LR --> not-RPL-aware-leaf (6LN)

In this case the flow label in 6LBR should be set zero in 6LBR, where RH3 is inserted and optionally RHI. RH3 must end at 6LR.

Header	Internet	6LBR	6LR	IPv6
Inserted headers	--	IPv6-in-IPv6(RH3, optional:RPI)	--	--
Removed headers	--	--	IPv6-in-IPv6(RH3, RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

NonStoring: Summary of the use of headers from Internet to not-RPL-aware-leaf

[6.9.](#) Example of Flow from RPL-aware-leaf to RPL-aware-leaf

In this case the flow comprises:

6LN --> 6LR --> root (6LBR) --> 6LR --> 6LN

This case comprises in the same RPL Domain. In the 6LN the RPI header is inserted. In the 6LBR the RH3 header is inserted in a IPv6-in-IPv6 header and removed at the 6LN destination.

Header	6LN src	6LBR	6LR	6LN dst
Inserted headers	RPI	IPv6-in-IPv6(RH3 to 6LN) {IP,RPI,payload}	--	--
Removed headers	--	--	--	IPv6-in-IPv6(RH3) {IP,RPI,payload}
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Non Storing: Summary of the use of headers for RPL-aware-leaf to RPL-aware-leaf

6.10. Example of Flow from RPL-aware-leaf to not-RPL-aware-leaf

In this case the flow comprises:

6LN --> 6LR --> root (6LBR) --> 6LR --> not-RPL-aware 6LN

The 6LN insert the RPI in a IPv6-in-IPv6 header, which is addressed to 6LBR. The 6LBR remove this RPI header and insert a RH3 header with an optional RPI. These headers are removed by 6LR before send the packet to the IPv6 node.

Header	6LN	6LBR	6LR	IPv6
Inserted headers	IPv6-in-IPv6(RPI)	IPIP(RH3, opt RPI)	--	--
Removed headers	--	IPIP(RPI)	IPIP(RH3, opt RPI)	--
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Non Storing: Summary of the use of headers from RPL-aware-leaf to not-RPL-aware-leaf

[6.11.](#) Example of Flow from not-RPL-aware-leaf to RPL-aware-leaf

In this case the flow comprises:

not-RPL-aware 6LN --> 6LR --> root (6LBR) --> 6LR --> 6LN

RPI is added in 6LR until the root and then removed, then RH3 is added and removed at destination.

Header	IPv6	6LR	6LBR	6LN
Inserted headers	--	IPIP(RPI)	IPIP(RH3)	--
Removed headers	--	IPIP(RPI)	--	IPIP(RH3)
Re-added headers	--	--	--	--
Modified headers	--	--	--	--
Untouched headers	--	--	--	--

Non Storing: Summary of the use of headers from not-RPL-aware-leaf to RPL-aware-leaf

[6.12.](#) Example of Flow from not-RPL-aware-leaf to not-RPL-aware-leaf

In this case the flow comprises:

not-RPL-aware 6LN --> 6LR --> root (6LBR) --> 6LR --> not-RPL-aware 6LN

RPI is added in 6LR until the root and then might be removed, then RH3 is added. These headers are removed at 6LR before go to destination.

Header	IPv6	6LR	6LBR	6LR	IPv6
Inserted headers	--	IPIP(RPI)	IPIP(RH3)	--	--
Removed headers	--	--	IPIP(RPI)	IPIP(RH3, opt RPI)	--
Re-added headers	--	--	--	--	--
Modified headers	--	--	--	--	--
Untouched headers	--	--	--	--	--

Non Storing: Summary of the use of headers from not-RPL-aware-leaf to not-RPL-aware-leaf

7. Future RPL work

There are cases from above that are not clear how to send the information. It requires further analysis on how to proceed to send the information from source to destination.

From the above cases, we have in storing mode:

- Flow from RPL-aware-leaf to non-RPL-aware-leaf: Somehow, the sender has to know that the receiver is not RPL aware, and needs to know 6LR, and not even the root knows where the 6LR is located.
- Flow from not-RPL-aware-leaf to not-RPL-aware-leaf: The problem to solve is how to indicate where to send the packet when get into LLN.

8. IANA Considerations

There are no IANA considerations related to this document.

9. Security Considerations

TODO.

10. Acknowledgments

This work is partially funded by the FP7 Marie Curie Initial Training Network (ITN) METRICS project (grant agreement No. 607728).

The authors would like to acknowledge the review, feedback, and comments of Thomas Watteyne, Xavier Vilajosana and Robert Cragie.

To be completed with additional Acknowledgments.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", [RFC 6550](#), DOI 10.17487/RFC6550, March 2012, <<http://www.rfc-editor.org/info/rfc6550>>.
- [RFC6553] Hui, J. and JP. Vasseur, "The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams", [RFC 6553](#), DOI 10.17487/RFC6553, March 2012, <<http://www.rfc-editor.org/info/rfc6553>>.
- [RFC6554] Hui, J., Vasseur, JP., Culler, D., and V. Manral, "An IPv6 Routing Header for Source Routes with the Routing Protocol for Low-Power and Lossy Networks (RPL)", [RFC 6554](#), DOI 10.17487/RFC6554, March 2012, <<http://www.rfc-editor.org/info/rfc6554>>.

11.2. Informative References

- [I-D.ietf-6tisch-architecture] Thubert, P., "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4", [draft-ietf-6tisch-architecture-08](#) (work in progress), May 2015.
- [I-D.thubert-6lo-routing-dispatch] Thubert, P., Bormann, C., Toutain, L., and R. Cragie, "A Routing Header Dispatch for 6LoWPAN", [draft-thubert-6lo-routing-dispatch-06](#) (work in progress), August 2015.
- [RFC7102] Vasseur, JP., "Terms Used in Routing for Low-Power and Lossy Networks", [RFC 7102](#), DOI 10.17487/RFC7102, January 2014, <<http://www.rfc-editor.org/info/rfc7102>>.

Authors' Addresses

Maria Ines Robles
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: maria.ines.robles@ericsson.com

Michael C. Richardson
Sandelman Software Works
470 Dawson Avenue
Ottawa, ON K1Z 5V7
CA

Email: mcr+ietf@sandelman.ca

URI: <http://www.sandelman.ca/>

Pascal Thubert
Cisco Systems, Inc
Village d'Entreprises Green Side 400, Avenue de Roumanille
Batiment T3, Biot - Sophia Antipolis 06410
France

Email: pthubert@cisco.com

