Workgroup: ROLL
Updates: 6550 (if approved)
Published: 17 October 2019
Intended Status: Standards Track
Expires: 19 April 2020
Authors: P. Thubert, Ed.   D. Barthel   R.A. Jadhav
         Cisco Systems   Orange Labs   Huawei Tech

Eliding and Querying RPL Information

Abstract

This document presents a method to safely elide a group of global RPL options by synchonizing the state associated with each of these options between parent and child using a new sequence counter in DIO messages. A child that missed a DIO message with an update of any of those protected options detects it by the change of sequence counter and queries the update with a DIS Message.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Classical Link State protocol synchronize their Link State Database (LSDB) by sequencing every change. Each interested node maintains the last sequence of the LSDB it is synchronizing with. If the last known sequence number is older than the current, the node needs to learn one by one all the state changes between the last known and the current state.

[RPL] does not operate that way. With RPL, the routing information is repeated over and over in DODAG Information Object (DIO) and Destination Advertisement Object (DAO) messages. There is no concept of synchronization. The most recent information overrides a previous one and a stale state eventually times out.

The RPL way was designed to enable routing from most nodes to most nodes most of the time in a Low-Power Lossy Network (LLN) where the quality of the links and the cost of communications does not permit to maintain a permanent synchronization.

This principle was applied to both the routing and non-routing information such as configuration settings, prefix information, and node capabilities.

This non-routing state is carried in RPL Messages as options. Some of the DIO options may be needed to decide whether a node can join a network as a leaf or as a router, and may affect the parent selection or the address selection. It is thus critical that each node maintains its state to the freshest and selects parents that are also synchronized to the freshest.

[RPL] allows a parent to elide options in the DIO messages that it sends repeatedly, to conserve battery and save bandwidth. When it does so, a newcomer child that missed DIOs that contained the configuration option may operate on default or partial information. If it is pessimistic, it may query all possible the information even when it is not needed. Conversely, a node that slept may have missed a DIO with a change in some critical information and may not be even aware of it, so it may fail to query for the update and operate on deprecated parameters.

This document uses a new sequence counter called RCSS to synchronize the state in a child node with that of its parent, and recursively with that of the whole network, to the latest setting from the Root.

The protected options are:

1. The Route Information Option (RIO) defined in section 6.7.5 of [RPL]

2. The DODAG Configuration Option (DCO) defined in section 6.7.6 of [RPL]

3. The Prefix Information Option (PIO) defined in section 6.7.10 of [RPL]

4. The Extended MOP Option (MOPex) defined in [MOPEX-CAP]

5. The Global Capabilities Option (GCO) defined in [MOPEX-CAP]

Any change in those options causes an increment of the RCSS and enables a network-wide synchronization to the new state. If the change impacts the routing substantially, the Root should decide to increment the Version Number at the same time to fully rebuild the DODAG with the new settings of the options. It must be noted that the operation of the Version Number in itself provides no guarantee that the non-routing state is fully resynchronized everywhere unless all the options are present in all the DIO messages.

2. Terminology

2.1. BCP 14

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119][RFC8174] when, and only when, they appear in all capitals, as shown here.

2.2. References

The Terminology used in this document is consistent with and incorporates that described in Terms Used in Routing for Low-Power and Lossy Networks [RFC7102].

Other terms in use in LLNs are found in Terminology for Constrained-Node Networks [RFC7228].

A glossary of classical RPL acronyms is given in Section 2.3.

The term "byte" is used in its now customary sense as a synonym for "octet".

"RPL", "RPL Packet Information" (RPI) and "RPL Instance", DIO, DAO and DIS messages are defined in the "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks" [RPL] specification.

This document uses the terms RPL-Unaware Leaf (RUL) and RPL Aware Leaf (RAL) consistently with [USE_OF_RPL_INFO].

The term RPL-Unaware Leaf (RUL) is used to refer to a node that uses a RPL router (without necessarily knowing it) as 6LR and depends on that router to obtain reachability for its addresses inside the RPL domain. On the contrary, the term RPL-Aware Node (RAN) is used to refer to a RAL or a RPL router that participates to RPL and advertises its addresses of prefixes by itself.

2.3. Glossary

This document often uses the following acronyms:

DODAG	Destination-Oriented Directed Acyclic Graph
LLN	Low-Power and Lossy Network
RPI	RPL Packet Information (an Option in the Hop-By_Hop Header)
RAL	RPL-Aware Leaf
RAN	RPL-Aware Node, a RPL router or a RPL-Aware Leaf
RS	Router Solicitation
RCSS	RPL Configuration State Sequence
RPL	IPv6 Routing Protocol for LLNs (pronounced ripple)
RUL	RPL-Unaware Leaf

3. Updating RFC 6550

This document adds a new field called RCSS to the DIO message. The RCSS is a sequence counter set by the Root and operated as specified in Section 7 of [RPL], more in Section 5.

This document also introduces a new RPL Control Message Option called the Abbreviated Option Option (AOO). The AOO is the compressed replacement of a protected option that indicates the RCSS of the last change of that option, but elides its content, more in Section 4.3.

This document modifies the DIS Base Objectto enable the individual query of the protected options by a node that missed a change, more in Section 4.2.

4. Message Formats

4.1. Updated DIO Base Object

The format of the DIO Base Object is defined in section 6.3.1 of [RPL]. This specification uses a 8th octet that was reserved in [RPL] to transport the RCSS.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RPLInstanceID |Version Number |             Rank              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|G|0| MOP | Prf |     DTSN      |     Flags     |      RCSS     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                            DODAGID                            +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Option(s)...
+-+-+-+-+-+-+-+-+
```

Figure 1: Updated DIO Base Object

Updated fields:

RCSS
One Byte, the RPL Configuration State Sequence

4.2. Updated DIS Base Object

The DIS Base Object is use by a child to query from a parent the most recent changes in protected options. This specification adds flags to indicate which options are requested and the freshest RCSS to which the querying node was synchronized.

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R|D|P[M|O| Flg | LastSync RCSS |   Option(s)...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Updated DIS Base Object

Updated fields:

R
One Bit, indicates that the RIO is requested

D
One Bit, indicates that the DCO is requested

P
One Bit, indicates that the PIO(s) is(are) requested

M
One Bit, indicates that the MOPex is requested

O
One Bit, indicates that the GCO is requested

Last Synchronized RCSS
One Byte, indicates the freshest RCSS to which the querier was synchronized

4.3. New Abbreviated Option Option

When a protected option is unchanged from the previous DIOs, the Root MAY replace it with its abbreviated version. The abbreviated version of an option is transported in a 4-bytes long Abbreviated Option Option (AOO). The AOO indicates the RCSS at which the protected option was last changed.

```
 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 2
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |  Option Type   | Option Length | Abbrev. opt.  | Last Mod RCSS |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Abbreviated Option Option Format

Option fields:

Option Type
One byte indicating "Abbreviated Option", see Table 2 in Section 8.2

Option Length
MUST be set to 2 indicating Option data of 2 bytes

Abbreviated Option
The Option Type of the option being abbreviated

Last Modification RCSS
The RCSS at which the option was last modified

5. RCSS Operation

Settings and updates to network-wide parameters are initiated by the Root and propagated down the DODAG in RPL Control Message Options in DIO messages. The DIO messages arrive asynchronously via different parents and may confuse a child. The RCSS allows the child to keep synchronized to the latest settings network-wide parameters that are propagated in protected options.

The RCSS is a sequence number that is operated as specified in section 7.2 of [RPL]. The scope of an RCSS is one DODAG within one RPL Instance. The RCSS applies to a DIO Message and a same value of the RCSS can be used in DIO messages that are sent consecutively with no change in the protected options.

The Root of the DODAG is autoritative to set and update the RCSS and the options that it protects. The RCSS and the protected options are propagated together down the DODAG without a change, more in Section 5.1.

The RCSS allows a child node to recognize the fresher DIO Message(s) as received from one or more advertising parents and to use only parents with a consistent state of network-wide parameters, more in Section 5.2.

By extension, the RCSS is also defined for each protected option. A child associates an option with the values of the RCSS indicated in DIO Messages in which the option is advertised and uses it to assess the relative freshness of different versions of an option, more in Section 5.3.

Unchanged options may be sent in full, elided, or in the abbreviated form specified in Section 4.3. Eliding an option is NOT RECOMMENDED as it may cause multiple children to resynchronize the option even if it was not changed.

If the link MTU does not permit to send a single DIO message with all the options packaged then the options may be spread over multiple consecutive DIO messages with the same RCSS that are sent in a rapid sequence.

5.1. Updating the RCSS

The RCSS is incremented by the Root using a lollipop technique as specified in section 7.2 of [RPL]. RCSS values are comparable if they are within a window of comparison of SEQUENCE_WINDOW increments or one indicates a reboot.

A reboot of the Root is detected when the RCSS moves from the circular to the straight part of the lollipop. In order to maximize the chances of detection, the straight part should be kept very short with a RECOMMENDED initialization at 252 or above.

During the straight part of the lollipop, a second reboot of the Root might not be recognized and a same value of the RCSS may reappear with different settings in the protected options. For that reason the protected options MUST be provided in full with each increment on the RCSS during the straight part of the lollipop.

When a field is modified in one of the protected options, the Root MUST send a DIO with an incremented RCSS and the modified protected option(s) in full. The Root MAY also update the Version Number to form a new DODAG altogether.

The Root SHOULD jump rapidly away from the straight part once the network has sufficiently settled by resetting the RCSS to 0, which places the RCSS in the circular region of the lollipop, where the protected options MAY be elided or abbreviated.

5.2. RCSS Freshness and Parent selection

A child node maintains the freshest RCSS received from its parents in each of the RPL Instances that it participates to, and uses that RCSS for its own DIO messages.

A child and a candidate parent are out-of-sync when the RCSS values that they maintain for a RPL Instance are not comparable. A child MUST NOT use a parent that is out-of-sync unless no other parent is available, in which case it MAY align its RCSS and resynchronize to that parent.

When a child receives from a candidate parent a DIO with an RCSS that is fresher than the one it is using, the child MUST synchronize the state relative to the protected options with that parent. The child node MUST refrain from using that parent and the new state including the RCSS, until it has synchronized all of the protected options to that RCSS. When it is fully synchronized, the child may then use that parent and the new RCSS.

Using a back-level parent may cause packets to be dropped, misunderstood or misrouted. The child SHOULD refrain from using a parent that exposes an older RCSS if the change causes an incompatibility issue.

5.3. RCSS of an Option

By extension, the RCSS of an option is maintained by a child as the freshest RCSS indicated by a DIO message from a candidate parent in which the option was present in the abbreviated form or in full. A child maintains a state for the freshest RCSS received for each of the protected options and synchronizes its state for each option to the freshest RCSS of that option.

When a child receives a DIO from a candidate parent, for each option:

If the Option is advertised in the abbreviated form then the RCSS that the DIO advertises for the option is the Last Modification RCSS of the AOO, else

If the Option is advertised in full then the RCSS that the DIO advertises for the option is the RCSS of the DIO, else

If the Option is elided then the RCSS is unspecified but it is at most as fresh as the RCSS of the DIO, and the RCSS of the DIO is assumed for the comparison

This means that if an Option is advertised in both the abbreviated form and in full in a same DIO message then the RCSS in the AOO has precedence.

To keep the RCSS comparable for each option, the RCSS of an option must lazily progress along with the global RCSS even if there was no change in the options. Each parent including the Root MUST advertise a new RCSS for each of the protected options at least once within a sliding window of SEQUENCE_WINDOW increments.

When an option was not changed for a new RCSS, one parent may advertise it in the abbreviated form while another sends the option in full only, e.g., in response to a DIS message. A fresher RCSS indicates that the option is either the same or carries a more recent update than the one with an older RCSS.

The RCSS of an option may be obtained from a DIO message that carries the option in full even if the RCSS of the DIO is not the freshest across parents, as long as the RCSS of the DIO is fresher than the current one for that option.

If current value of the maintained RCSS for an given option is not as fresh or fresher than that advertised in a DIO message, then the child MUST update its state for that option as specified in Section 6.

6. Synchronizing Options

A child can resynchronize any of the protected options to the latest RCSS by sending a DIS Message to a candidate parent that advertises that RCSS in DIO messages. The child MUST set the desired combination of 'R', 'D', 'P', 'M' and 'O' flags to indicate the option(s) that it needs updated. The child MUST signal in the Last Synchronized RCSS field of the DIS the freshest value of RCSS for which it was fully synchronized, or a conventional value of OUT-OF-SYNC-RCSS of 129 if it was never synchronized or is out-of-sync with the parent.

The DIO message that is sent in response MUST contain in full all the options that are requested and that were updated since the Last Synchronized RCSS in the DIS Message. This means all of the protected options of the child was never synchronized or is out-of-sync with the parent. The other options MUST be added in the abbreviated form. The options MAY be spread over more than one DIO message sent in a quick sequence and the child SHOULD wait a reasonable technology-dependent time before it retries the request.

7. Security Considerations

TBD

8. IANA Considerations

8.1. New DODAG Information Object Flags

5 new bits are allocated in the Registry for the DODAG Information Object (DIO) Flags defined for [RPL].

Bit Number	Capability description	Defining RFC
0	'R' bit "RIO requested"	THIS RFC
1	'D' bit "DCO requested"	THIS RFC
2	'P' bit "PIO(s) requested"	THIS RFC
3	'M' bit "MOPex requested"	THIS RFC
4	'O' bit "GCO irequested"	THIS RFC

Table 1: New DIO Flags

8.2. New RPL Control Message Option

A new entry is required for the new option of type "Abbreviated Option", from the "RPL Control Message Options" space defined for [RPL].

Code	Description	Defining RFC
TBD IANA	Abbreviated Option	THIS RFC

Table 2: New Option Type

9. Acknowledgments

10. Normative References

[MOPEX-CAP] Jadhav, R. and P. Thubert, "Mode of Operation extension and Capabilities", Work in Progress, Internet-Draft, draft-ietf-roll-mopex-cap-00, 9 August 2019, <https://tools.ietf.org/html/draft-ietf-roll-mopex-cap-00>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC7102] Vasseur, JP., "Terms Used in Routing for Low-Power and Lossy Networks", RFC 7102, DOI 10.17487/RFC7102, January 2014, <https://www.rfc-editor.org/info/rfc7102>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <https://www.rfc-editor.org/info/rfc7228>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RPL] 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, <https://www.rfc-editor.org/info/rfc6550>.

[USE_OF_RPL_INFO] Robles, I., Richardson, M., and P. Thubert, "Using RPL Option Type, Routing Header for Source Routes and IPv6-in-IPv6 encapsulation in the RPL Data Plane", Work in Progress, Internet-Draft, draft-ietf-roll-useofrplinfo-31, 7 August 2019, <https://tools.ietf.org/html/draft-ietf-roll-useofrplinfo-31>.

11. Informative References

Authors' Addresses

Pascal Thubert (editor)
Cisco Systems, Inc
Building D, 45 Allee des Ormes - BP1200
06254 Mougins - Sophia Antipolis
France

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

Dominique Barthel
Orange Labs
28 chemin du Vieux Chêne
38243 Meylan
France

Email: dominique.barthel@orange.com

Rahul Arvind Jadhav
Huawei Tech
Kundalahalli Village, Whitefield,
Bangalore 560037
Karnataka
India

Phone: +91-080-49160700
Email: rahul.ietf@gmail.com