Abstract

This document extends RFC 6550, RFC 6553 and RFC 8138 and enable to install a limited amount of centrally-computed routes in a RPL graph, enabling loose source routing down a non-storing mode DODAG, or transversal routes inside the DODAG. In constrast with classical routes in RPL that are injected by the end devices, this draft enables the root of the DODAG to projects the routes that are needed on the nodes where they should be installed.

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 https://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 November 25, 2019.

Copyright Notice

Copyright (c) 2019 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
1. **Introduction**

The "Routing Protocol for Low Power and Lossy Networks" [RFC6550] (LLN)(RPL) is a generic Distance Vector protocol that is well suited low energy Internet of Things (IoT) networks. RPL forms Destination
Oriented Directed Acyclic Graphs (DODAGs) in which the root often acts as the Border Router to connect the RPL domain to the Internet. The root is responsible to select the RPL Instance that is used to forward a packet coming from the Internet into the RPL domain and set the related RPL information in the packets.

The 6TiSCH architecture [I-D.ietf-6tisch-architecture] leverages RPL for its routing operation and considers the Deterministic Networking Architecture [I-D.ietf-detnet-architecture] as one possible model whereby the device resources and capabilities are exposed to an external controller which installs routing states into the network based on some objective functions that reside in that external entity.

Based on heuristics of usage, path length, and knowledge of device capacity and available resources such as battery levels and reservable buffers, a Path Computation Element ([PCE]) with a global visibility on the system could install additional P2P routes that are more optimized for the current needs as expressed by the objective function.

This draft enables a RPL root to install and maintain projected routes (P-Routes) within its DODAG, along a selected set of nodes that may or may not include self, for a chosen duration. This potentially enables routes that are more optimized than those obtained with the distributed operation of RPL, either in terms of the size of a source-route header or in terms of path length, which impacts both the latency and the packet delivery ratio. P-routes may be installed in either Storing and Non-Storing Modes Instances of the classical RPL operation, resulting in potentially hybrid situations where the mode of some P-routes is different from that of the other routes in the RPL Instance.

P-Routes must be used with the parsimony to limit the amount of state that is installed in each device to fit within its resources, and to limit the amount of rerouted traffic to fit within the capabilities of the transmission links. The algorithm used to compute the paths and the protocol used to learn the topology of the network and the resources that are available in devices and in the network are out of scope for this document. Possibly with the assistance of a Path Computation Element ([PCE]) that could have a better visibility on the larger system, the root computes which segment could be optimized and uses this draft to install the corresponding P-Routes.
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. New Terms

P-Route: A route that is installed remotely by a RPL root.

2.3. References

In this document, readers will encounter terms and concepts that are discussed in the following documents:

- "Routing Protocol for Low Power and Lossy Networks" [RFC6550], and
- "Terminology in Low power And Lossy Networks" [RFC7102].

3. Extending RFC 6550

Section 6.7 of RPL [RFC6550] specifies Control Message Options (CMO) to be placed in RPL messages such as the Destination Advertisement Object (DAO) message. The RPL Target Option and the Transit Information Option (TIO) are such options. In Non-Storing Mode, the TIO option is used in the DAO message to indicate the immediate parent of a given path. The TIO applies to the Target options that immediately precede it. Options may be factorized; multiple TIOs may be present to indicate multiple routes to the one or more contiguous addressed indicated in the Target Options that immediately precede the TIOs in the RPL message.

This specification introduces two new Control Message Options referred to as Route Projection Options (RPO). One RPO is the Information option (VIO) and the other is the Source-Routed VIO (SRVIO). The VIO installs a route on each hop along a P-Route (in a fashion analogous to RPL Storing Mode) whereas the SRVIO installs a source-routing state at the ingress node, which uses it to insert a routing header in a fashion similar to Non-Storing Mode.

Like the TIO, the RPOs MUST be preceded by one or more RPL Target Options to which they apply, and they can be factorized: multiple contiguous RPOs indicate alternate paths to the target(s).
3.1. RPL Instances

It must be noted that RPL has a concept of instance but does not have a concept of an administrative distance, which exists in certain proprietary implementations to sort out conflicts between multiple sources of routing information. This draft conforms the instance model as follows:

- If the PCE needs to influence a particular instance to add better routes in conformance with the routing objectives in that instance, it may do so. When the PCE modifies an existing instance then the added routes must not create a loop in that instance. This is achieved by always preferring a route obtained from the PCE over a route that is learned via RPL.

- If the PCE installs a more specific (say, Traffic Engineered) route between a particular pair of nodes then it SHOULD use a Local Instance from the ingress node of that path. A packet associated with that instance will be routed along that path and MUST NOT be placed over a Global Instance again. A packet that is placed on a Global Instance may be injected in the Local Instance based on node policy and the Local Instance parameters.

In all cases, the path is indicated by a new Via Information option, and the flow is similar to the flow used to obtain loose source routing.

3.2. New RPL Control Message Options

The format of RPOs is as follows:
Option Type: 0x0A for VIO, 0x0B for SRVIO (to be confirmed by IANA)

Option Length: In bytes; variable, depending on the number of Via Addresses.

Path Sequence: 8-bit unsigned integer. When a RPL Target option is issued by the root of the DODAG (i.e. in a DAO message), that root sets the Path Sequence and increments the Path Sequence each time it issues a RPL Target option with updated information. The indicated sequence deprecates any state for a given Target that was learned from a previous sequence and adds to any state that was learned for that sequence.

Path Lifetime: 8-bit unsigned integer. The length of time in Lifetime Units (obtained from the Configuration option) that the prefix is valid for route determination. The period starts when a new Path Sequence is seen. A value of 255 (0xFF) represents infinity. A value of zero (0x00) indicates a loss of reachability. A DAO message that contains a Via Information
option with a Path Lifetime of zero for a Target is referred as a No-Path (for that Target) in this document.

Via Address: 16 bytes. IPv6 Address of the next hop towards the destination(s) indicated in the target option that immediately precede the RPO. Via Addresses are indicated in the order of the data path from the ingress to the egress nodes.

An RPO MUST contain at least one Via Address, and a Via Address MUST NOT be present more than once, otherwise the RPO MUST be ignored.

3.3. RPI for Projected Routes

RPL [RFC6550], Section 11.2, specifies the RPL Packet Information (RPI) as a set of fields that are placed by RPL routers in IP packets to identify the RPL Instance, detect anomalies and trigger corrective actions.

In particular, the SenderRank, which is the scalar metric computed by a specialized Objective Function such as described in [RFC6552], indicates the Rank of the sender and is modified at each hop. The SenderRank field is used to validate that the packet progresses in the expected direction, either upwards or downwards, along the DODAG.

RPL defines the "RPL Option for Carrying RPL Information in Data-Plane Datagrams" [RFC6553] to transport the RPI, which is carried in an IPv6 Hop-by-Hop Options Header [RFC8200], typically consuming eight bytes per packet.

This specification updates [RFC6550] as follows. When using projected routes, the Rank is useless and SHOULD be set to 0 in the non-compressed form, and can be elided in the compressed form (see Section 4.1). In a same fashion, the O, R, and F flags that are defined in Section 11.2 of [RFC6550] are not used for packets that follow a projected route and they MUST be reset. A new flag is added, the P flag that indicates that the packet is injected along a projected route.

3.4. Projected DAO

This draft adds a capability to RPL whereby the root of a DODAG projects a route by sending an extended DAO message called a Projected-DAO (P-DAO) to an arbitrary router in the DODAG, indicating one or more sequence(s) of routers inside the DODAG via which the target(s) indicated in the Target Information Option(s) (TIO) can be reached.
A P-DAO is sent from a global address of the root to a global address of the recipient, and MUST be confirmed by a DAO-ACK, which is sent back to a global address of the root.

A P-DAO message MUST contain at least one TIO and at least one RPO following it. There can be at most one such sequence of TIos and then RPOS.

Like a classical DAO message, a P-DAO is processed only if it is "new" per section 9.2.2. "Generation of DAO Messages" of the RPL specification [RFC6550]; this is determined using the Path Sequence information from the RPO as opposed to a TIO. Also, a Path Lifetime of 0 in an RPO indicates that a route is to be removed.

There are two kinds of operation for the P-Routes, the Storing Mode and the Non-Storing Mode.

- The Non-Storing Mode is discussed in Section 3.4.1. It uses an SRVIO that carries a list of Via Addresses to be used as a source-routed path to the target. The recipient of the P-DAO is the ingress router of the source-routed path. Upon a Non-Storing Mode P-DAO, the ingress router installs a source-routed state to the target and replies to the root directly with a DAO-ACK message.

- The Storing Mode is discussed in Section 3.4.2. It uses a VIO with one Via Address per consecutive hop, from the ingress to the egress of the path, including the list of all intermediate routers in the data path order. The Via Addresses indicate the routers in which the routing state to the target have to be installed via the next Via Address in the VIO. In normal operations, the P-DAO is propagated along the chain of Via Routers from the egress router of the path till the ingress one, which confirms the installation to the root with a DAO-ACK message. Note that the root may be the ingress and it may be the egress of the path, that it can also be neither but it cannot be both.

In case of a forwarding error along a P-Route, an ICMP error is sent to the root with a new Code "Error in Projected Route" (See Section 7.3). The root can then modify or remove the P-Route. The "Error in Projected Route" message has the same format as the "Destination Unreachable Message", as specified in RFC 4443 [RFC4443]. The portion of the invoking packet that is sent back in the ICMP message SHOULD record at least up to the routing header if one is present, and the routing header SHOULD be consumed by this node so that the destination in the IPv6 header is the next hop that this node could not reach. if a 6LoWPAN Routing Header (6LoRH) [RFC8138] is used to carry the IPv6 routing information in the outer header then that whole 6LoRH information SHOULD be present in the
ICMP message. The sender and exact operation depend on the Mode and is described in Section 3.4.1 and Section 3.4.2 respectively.

3.4.1. Non-Storing Mode P-Route

As illustrated in Figure 2, a P-DAO that carries an SRVIO enables the root to install a source-routed path towards a target in any particular router; with this path information the router can add a source routed header reflecting the P-route to any packet for which the current destination either is the said target or can be reached via the target.

---

Figure 2: Projecting a Non-Storing Route

A route indicated by an SRVIO may be loose, meaning that the node that owns the next listed Via Address is not necessarily a neighbor. Without proper loop avoidance mechanisms, the interaction of loose source routing and other mechanisms may effectively cause loops. In order to avoid those loops, if the router that installs a P-route does not have a connected route (a direct adjacency) to the next source routed hop and fails to locate it as a neighbor or a neighbor of a neighbor, then it MUST ensure that it has another P-Route to the next loose hop under the control of the same route computation system, otherwise the P-DAO is rejected.

When forwarding a packet to a destination for which the router determines that routing happens via the target, the router inserts the source routing header in the packet to reach the target. In the
case of a loose source-routed path, there MUST be either a neighbor that is adjacent to the loose next hop, on which case the packet is forwarded to that neighbor, or a source-routed path to the loose next hop; in the latter case, another encapsulation takes place and the process possibly recurses; otherwise the packet is dropped.

In order to add a source-routing header, the router encapsulates the packet with an IP-in-IP header and a non-storing mode source routing header (SRH) [RFC6554]. In the uncompressed form the source of the packet would be self, the destination would be the first Via Address in the SRVIO, and the SRH would contain the list of the remaining Via Addresses and then the target.

In practice, the router will normally use the "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch" [RFC8025] to compress the RPL artifacts as indicated in the "6LoWPAN Routing Header" [RFC8138] specification. In that case, the router indicates self as encapsulator in an IP-in-IP 6LoRH Header, and places the list of Via Addresses in the order of the VIO and then the target in the SRH 6LoRH Header.

```
+--+-+-+- ... -+-+-+- ...     ...      -+-+-+- ... -+-+ ...     
|11110001|SRH-6LoRH| ERPI- | IP-in-IP Encap | NH=1      |11110CPP|
|Page 1  |Type1 S=2| 6LoRH | 6LoRH Encap    |LOWPAN_IPHC| UDP    |
+--+-+-+- ... -+-+-+- ...     ...      -+-+-+- ... -+-+ ...     
<--RFC8138--><-This--><----RFC 8138----><-----RFC 6282------->
RFC      5 to 19 bytes No RPL artifact
```

Figure 3: Example Compressed Packet with SRH.

In case of a forwarding error along a Source Route path, the node that fails to forward SHOULD send an ICMP error with a code "Error in Source Routing Header" back to the source of the packet, as described in section 11.2.2.3. of [RFC6550]. Upon this message, the encapsulating node SHOULD stop using the source route path for a period of time and it SHOULD send an ICMP message with a Code "Error in Projected Route" to the root. Failure to follow these steps may result in packet loss and wasted resources along the source route path that is broken.

3.4.2. Storing-Mode P-Route

As illustrated in Figure 4, the Storing Mode projected iq used by the root to install a routing state towards a target in the routers along a segment between an ingress and an egress router; this enables the routers to forward along that segment any packet for which the next loose hop is the said target, for instance a loose source routed packet for which the next loose hop is the target, or a packet for
which the router has a routing state to the final destination via the target.

Figure 4: Projecting a route

In order to install the relevant routing state along the segment between an ingress and an egress routers, the root sends a unicast P-DAO message to the egress router of the routing segment that must be installed. The P-DAO message contains the ordered list of hops along the segment as a direct sequence of Via Information options that are preceded by one or more RPL Target options to which they relate. Each Via Information option contains a Path Lifetime for which the state is to be maintained.

The root sends the P-DAO directly to the egress node of the segment. In that P-DAO, the destination IP address matches the Via Address in the last VIO. This is how the egress recognizes its role. In a similar fashion, the ingress node recognizes its role as it matches Via Address in the first VIO.

The egress node of the segment is the only node in the path that does not install a route in response to the P-DAO; it is expected to be already able to route to the target(s) on its own. It may either be the target, or may have some existing information to reach the target(s), such as a connected route or an already installed P-Route. If one of the targets cannot be located, the node MUST answer to the root with a negative DAO-ACK listing the target(s) that could not be located (suggested status 10 to be confirmed by IANA).
If the egress node can reach all the targets, then it forwards the P-DAO with unchanged content to its loose predecessor in the segment as indicated in the list of Via Information options, and recursively the message is propagated unchanged along the sequence of routers indicated in the P-DAO, but in the reverse order, from egress to ingress.

The address of the predecessor to be used as destination of the propagated DAO message is found in the Via Information option the precedes the one that contain the address of the propagating node, which is used as source of the packet.

Upon receiving a propagated DAO, an intermediate router as well as the ingress router install a route towards the DAO target(s) via its successor in the P-DAO; the router locates the VIO that contains its address, and uses as next hop the address found in the Via Address field in the following VIO. The router MAY install additional routes towards the addresses that are located in VIOs that are after the next one, if any, but in case of a conflict or a lack of resource, a route to a target installed by the root has precedence.

The process recurses till the P-DAO is propagated to ingress router of the segment, which answers with a DAO-ACK to the root.

Also, the path indicated in a P-DAO may be loose, in which case the reachability to the next hop has to be asserted. Each router along the path indicated in a P-DAO is expected to be able to reach its successor, either with a connected route (direct neighbor), or by routing, for instance following a route installed previously by a DAO or a P-DAO message. If that route is not connected then a recursive lookup may take place at packet forwarding time to find the next hop to reach the target(s). If it does not and cannot reach the next router in the P-DAO, the router MUST answer to the root with a negative DAO-ACK indicating the successor that is unreachable (suggested status 11 to be confirmed by IANA).

A Path Lifetime of 0 in a Via Information option is used to clean up the state. The P-DAO is forwarded as described above, but the DAO is interpreted as a No-Path DAO and results in cleaning up existing state as opposed to refreshing an existing one or installing a new one.

In case of a forwarding error along a Storing Mode P-Route, the node that fails to forward SHOULD send an ICMP error with a code "Error in Projected Route" to the root. Failure to do so may result in packet loss and wasted resources along the P-Route that is broken.
4. Extending RFC 8138

4.1. Elective RPI 6LoRH

[RFC8138] defines a Critical 6LoRH to compress the RPL RPI found in normal packets inside a RPL domain, the RPI-6LoRH.

This specification introduces the ERPI-6LoRH header that MUST be used to compress the RPI in packets that follow a projected route. As discussed in Section 3.3, the Rank and the O, R, and F flags are always set to 0 and can be elided. The new P flag is always set and can also be elided. It results that in general only the RPL InstanceID is necessary in the compressed form.

This specification adds an optimization whereby the local RPLInstanceID 0 for the source of the packet (the encapsulator when using IP in IP) can be elided. This is the case where the RPLInstanceID is encoded as binary b10000000, decimal 128, in the non-compressed form.

The ERPI-6LoRH header is Elective since it does not contain information that is critical to the routing and it can be ignored when not understood. The resulting format is illustrated in Figure 5 below:

```
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1      |0      | Length  | 6LoRH Type 5 | RPLInstanceID |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: A ERPI-6LoRH carrying a RPLInstanceID

The ERPI-6LoRH header is identifies by a 6LoRH Type of 5 (to be confirmed by IANA), which is the same value as the RPI-6LoRH but in the Elective namespace. If the RPLInstanceID is a local RPLInstanceID 0 for the source of the packet then it MUST be elided and the length MUST be set to 0. Else the length MUST be set to 1 to indicate that the ERPI-6LoRH carries a RPLInstanceID.

5. Extending RFC 6553

5.1. Uncompressed RPL Option

[RFC6553] defines a format for the RPI that is suitable for transporting in the IPv6 Hop-by-Hop Header [RFC8200]. This...
specification introduces a new flag in the RPI that must be encoded in any format including uncompressed.

The updated format for the RPL Option is presented in Figure 6.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Option Type | Opt Data Len |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|O|R|F|P|0|0|0|0| RPLInstanceID |          SenderRank           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         (sub-TLVs)                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: RPL Option

New fields:

P: 1-bit flag; indicates that the packet is routed along a projected route.

6. Security Considerations

This draft uses messages that are already present in RPL [RFC6550] with optional secured versions. The same secured versions may be used with this draft, and whatever security is deployed for a given network also applies to the flows in this draft.

TODO: should probably consider how P-DAO messages could be abused by a) rogue nodes b) via replay of messages c) if use of P-DAO messages could in fact deal with any threats?

7. IANA Considerations

7.1. New Elective 6LoWPAN Routing Header Type

This specification assigns a new value (to be confirmed by IANA) in the Elective 6LoWPAN Routing Header Type Registry created for RFC 8138 as below:
### 7.2. New RPL Control Codes

This document extends the IANA registry created by RFC 6550 for RPL Control Codes as follows:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0A</td>
<td>Via</td>
<td>This document</td>
</tr>
<tr>
<td>0x0B</td>
<td>Source-Routed Via</td>
<td>This document</td>
</tr>
</tbody>
</table>

**RPL Control Codes**

This document is updating the registry created by RFC 6550 for the RPL 3-bit Mode of Operation (MOP) as follows:

<table>
<thead>
<tr>
<th>MOP value</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Non-Storing mode of operation with P-Routes</td>
<td>This document</td>
</tr>
<tr>
<td>6</td>
<td>Storing mode of operation with P-Routes</td>
<td>This document</td>
</tr>
</tbody>
</table>

**DIO Mode of operation**

### 7.3. Error in Projected Route ICMPv6 Code

In some cases RPL will return an ICMPv6 error message when a message cannot be forwarded along a P-Route. This ICMPv6 error message is "Error in Projected Route".

IANA has defined an ICMPv6 "Code" Fields Registry for ICMPv6 Message Types. ICMPv6 Message Type 1 describes "Destination Unreachable" codes. This specification requires that a new code is allocated from the ICMPv6 Code Fields Registry for ICMPv6 Message Type 1, for "Error
in Projected Route", with a suggested code value of 8, to be confirmed by IANA.

8. Acknowledgments

The authors wish to acknowledge JP Vasseur and Patrick Wetterwald for their contributions to the ideas developed here.

9. References

9.1. Normative References


9.2. Informative References

[I-D.ietf-6tisch-architecture]

[I-D.ietf-detnet-architecture]

[PCE]

[RFC6997]

[RFC7102]
Appendix A. Applications

A.1. Loose Source Routing in Non-storing Mode

A RPL implementation operating in a very constrained LLN typically uses the Non-Storing Mode of Operation as represented in Figure 7. In that mode, a RPL node indicates a parent-child relationship to the root, using a Destination Advertisement Object (DAO) that is unicast from the node directly to the root, and the root typically builds a source routed path to a destination down the DODAG by recursively concatenating this information.

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

Figure 7: RPL non-storing mode of operation

Based on the parent-children relationships expressed in the non-storing DAO messages, the root possesses topological information about the whole network, though this information is limited to the structure of the DODAG for which it is the destination. A packet that is generated within the domain will always reach the root, which can then apply a source routing information to reach the destination if the destination is also in the DODAG. Similarly, a packet coming from the outside of the domain for a destination that is expected to be in a RPL domain reaches the root.

It results that the root, or then some associated centralized computation engine such as a PCE, can determine the amount of packets that reach a destination in the RPL domain, and thus the amount of energy and bandwidth that is wasted for transmission, between itself and the destination, as well as the risk of fragmentation, any potential delays because of a paths longer than necessary (shorter paths exist that would not traverse the root).
As a network gets deep, the size of the source routing header that the root must add to all the downward packets becomes an issue for nodes that are many hops away. In some use cases, a RPL network forms long lines and a limited amount of well-targeted routing state would allow to make the source routing operation loose as opposed to strict, and save packet size. Limiting the packet size is directly beneficial to the energy budget, but, mostly, it reduces the chances of frame loss and/or packet fragmentation, which is highly detrimental to the LLN operation. Because the capability to store a routing state in every node is limited, the decision of which route is installed where can only be optimized with a global knowledge of the system, a knowledge that the root or an associated PCE may possess by means that are outside of the scope of this specification.

This specification enables to store source-routed or storing mode state in intermediate routers, which enables to limit the excursion of the source route headers in deep networks. Once a P-DAO exchange has taken place for a given target, if the root operates in non-storing mode, then it may elide the sequence of routers that is installed in the network from its source route headers to destination that are reachable via that target, and the source route headers effectively become loose.

A.2. Transversal Routes in storing and non-storing modes

RPL is optimized for Point-to-Multipoint (P2MP) and Multipoint-to-Point (MP2P), whereby routes are always installed along the RPL DODAG respectively from and towards the DODAG Root. Transversal Peer to Peer (P2P) routes in a RPL network will generally suffer from some elongated (stretched) path versus the best possible path, since routing between 2 nodes always happens via a common parent, as illustrated in Figure 8:

- In non-storing mode, all packets routed within the DODAG flow all the way up to the root of the DODAG. If the destination is in the same DODAG, the root must encapsulate the packet to place a Routing Header that has the strict source route information down the DODAG to the destination. This will be the case even if the destination is relatively close to the source and the root is relatively far off.

- In storing mode, unless the destination is a child of the source, the packets will follow the default route up the DODAG as well. If the destination is in the same DODAG, they will eventually reach a common parent that has a route to the destination; at worse, the common parent may also be the root. From that common parent, the packet will follow a path down the DODAG that is
optimized for the Objective Function that was used to build the DODAG.

```
+-----+---------
|     | Internet|
+-----+---------
      |
      |
      X
      ^  v  o  o
      ^  ^  o  v  o  o  o  o
      ^  ^  ^  ^  o  v  o  o  o  o
      ^  ^  ^  ^  ^  ^  o  v  o  o  o  o  o
      S  o  o  o  D  o  o  o  o
      o  o  o  o  o  o  o  o  o
      LLN
```

Figure 8: Routing Stretch between S and D via common parent X

It results that it is often beneficial to enable transversal P2P routes, either if the RPL route presents a stretch from shortest path, or if the new route is engineered with a different objective. For that reason, earlier work at the IETF introduced the "Reactive Discovery of Point-to-Point Routes in Low Power and Lossy Networks" [RFC6997], which specifies a distributed method for establishing optimized P2P routes. This draft proposes an alternate based on a centralized route computation.

```
+-----+---------
|     | Internet|
+-----+---------
      |
      |
      |
      S>>>A>>>B>>>C>>>D  o  o  o  o
      o  o  o  o  o  o  o  o  o  o
      o  o  o  o  o  o  o  o  o  o
      o  o  o  o  o  o  o  o  o  o
      o  o  o  o  o  o  o  o  o  o
      LLN
```

Figure 9: Projected Transversal Route
This specification enables to store source-routed or storing mode state in intermediate routers, which enables to limit the stretch of a P2P route and maintain the characteristics within a given SLA. An example of service using this mechanism could be a control loop that would be installed in a network that uses classical RPL for asynchronous data collection. In that case, the P2P path may be installed in a different RPL Instance, with a different objective function.

Appendix B. Examples

B.1. Using storing mode P-DAO in non-storing mode MOP

In non-storing mode, the DAG root maintains the knowledge of the whole DODAG topology, so when both the source and the destination of a packet are in the DODAG, the root can determine the common parent that would have been used in storing mode, and thus the list of nodes in the path between the common parent and the destination. For instance in the diagram shown in Figure 10, if the source is node 41 and the destination is node 52, then the common parent is node 22.

```
+-----+       +-----+       +-----+
|     |       |     |       |     |
| Internet | Border Router | (RPL Root) |
|      |       |     |       |     |
|      |       |     |       |     |
+-----+       +-----+       +-----+
    |       |       |       |       |
    | \\     | \\     | \\     | \\     |
    |  o 11  |  o 12  |  o 13  |  o 22  |
    | /       | /       | /       | /       |
    |  o 23  |  o 24  |  o 25  |  o 31  |
    | /       | /       | /       | /       |
    |  o 32  |  o 35  |  o 41  |  o 42  |
    | /       | /       | /       | /       |
    |  o 45  |  o 46  |  o 51  |  o 52  |
    | /       | /       | /       | /       |
    |  o 55  |  o 56  |  o 53  |  o 54  |
    |         |         |         |         |
LLN
```

Figure 10: Example DODAG forming a logical tree topology

With this draft, the root can install a storing mode routing states along a segment that is either from itself to the destination, or from one or more common parents for a particular source/destination.
pair towards that destination (in this particular example, this would be the segment made of nodes 22, 32, 42).

In the example below, say that there is a lot of traffic to nodes 55 and 56 and the root decides to reduce the size of routing headers to those destinations. The root can first send a DAO to node 45 indicating target 55 and a Via segment (35, 45), as well as another DAO to node 46 indicating target 56 and a Via segment (35, 46). This will save one entry in the routing header on both sides. The root may then send a DAO to node 35 indicating targets 55 and 56 a Via segment (13, 24, 35) to fully optimize that path.

Alternatively, the root may send a DAO to node 45 indicating target 55 and a Via segment (13, 24, 35, 45) and then a DAO to node 46 indicating target 56 and a Via segment (13, 24, 35, 46), indicating the same DAO Sequence.

B.2. Projecting a storing-mode transversal route

In this example, say that a PCE determines that a path must be installed between node S and node D via routers A, B and C, in order to serve the needs of a particular application.

The root sends a P-DAO with a target option indicating the destination D and a sequence Via Information option, one for S, which is the ingress router of the segment, one for A and then for B, which are an intermediate routers, and one for C, which is the egress router.

| Internet-Draft | Root initiated routing state in RPL | May 2019 |

Figure 11: P-DAO from root
Upon reception of the P-DAO, C validates that it can reach D, e.g. using IPv6 Neighbor Discovery, and if so, propagates the P-DAO unchanged to B.

B checks that it can reach C and of so, installs a route towards D via C. Then it propagates the P-DAO to A.

The process recurses till the P-DAO reaches S, the ingress of the segment, which installs a route to D via A and sends a DAO-ACK to the root.

As a result, a transversal route is installed that does not need to follow the DODAG structure.
Internet-Draft     Root initiated routing state in RPL          May 2019

Figure 13: Projected Transversal Route

Authors’ Addresses

Pascal Thubert (editor)
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

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

Phone: +91-080-49160700
Email: rahul.iétf@gmail.com
Matthew Gillmore
Itron, Inc
Building D
2111 N Molter Road
Liberty Lake  99019
United States
Phone: +1.800.635.5461
Email: matthew.gillmore@itron.com

James Pylakutty
Cisco Systems
Cessna Business Park
Kadubeesanahalli
Marathalli ORR
Bangalore, Karnataka  560087
INDIA
Phone: +91 80 4426 4140
Email: mundenma@cisco.com
Common Ancestor Objective Functions and Parent Set DAG Metric Container Extension
draft-ietf-roll-nsa-extension-04

Abstract

Implementing Packet Replication and Elimination from / to the RPL root requires the ability to forward copies of packets over different paths via different RPL parents. Selecting the appropriate parents to achieve ultra-low latency and jitter requires information about a node’s parents. This document details what information needs to be transmitted and how it is encoded within a packet to enable this functionality. This document also describes Objective Functions which take advantage of this information to implement multi-path routing.

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 https://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 January 9, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

Network-enabled applications in the industrial context must provide stringent guarantees in terms of reliability and predictability. To achieve this they typically leverage 1+1 redundancy, also known as Packet Replication and Elimination (PRE) [I-D.papadopoulos-6tisch-pre-reqs]. Allowing these kinds of applications to function over wireless networks requires the application of the principles of Deterministic Networking [I-D.ietf-detnet-architecture]. This results in designs which aim at optimizing packet delivery rate and bounding latency. Additionally, given that the network nodes often do not have an unlimited power supply, energy consumption needs to be minimized as well.

As an example, to meet this goal, IEEE Std. 802.15.4 [IEEE802154] provides Time-Slotted Channel Hopping (TSCH), a mode of operation which uses a common communication schedule based on timeslots to allow deterministic medium access as well as channel hopping to work
around radio interference. However, since TSCH uses retransmissions in the event of a failed transmission, end-to-end delay and jitter performance can deteriorate.

Furthermore, the 6TiSCH working group, focusing on IPv6 over IEEE Std. 802.15.4-TSCH, has worked on the issues previously highlighted and produced the "6TiSCH Architecture" [I-D.ietf-6tisch-architecture] to address that case. Building on this architecture, "Exploiting Packet Replication and Elimination in Complex Tracks in 6TiSCH LLNs" [I-D.papadopoulos-6tisch-pre-reqs] leverages PRE to improve the Packet Delivery Ratio (PDR), to provide a hard bound to the end-to-end latency, and to limit jitter.

PRE is a general method of maximizing packet delivery rate and potentially minimizing latency and jitter, not limited to 6TiSCH. More specifically, PRE achieves controlled redundancy by laying multiple forwarding paths through the network and using them in parallel for different copies of a same packet. PRE can follow the Destination-Oriented Directed Acyclic Graph (DODAG) formed by RPL from a node to the root. Building a multi-path DODAG can be achieved based on the RPL capability of having multiple parents for each node in a network, a subset of which is used to forward packets. In order for this subset to be defined, a RPL parent subset selection mechanism, which is among the responsibilities of the RPL Objective Function (OF), needs to have specific path information. This document describes OFs which implement multi-path routing for PRE and specifies the transmission of this specific path information.

For the OFs, this document specifies a group of OFs called Common Ancestor (CA) OFs. A detailed description is made of how the path information is used within the CA OF and how the subset of parents for forwarding packets is selected. This specification defines new Objective Code Points (OCPs) for these CA OFs.

For the path information, this specification focuses on the extensions to the DAG Metric Container [RFC6551] required for providing the PRE mechanism a part of the information it needs to operate. This information is the RPL [RFC6550] parent address set of a node and it must be sent to potential children of the node. The RPL DIO Control Message is the canonical way of broadcasting this kind of information and therefore its DAG Metric Container [RFC6551] field is used to append a Node State and Attribute (NSA) object. The node's parent address set is stored as an optional TLV within the NSA object. This specification defines the type value and structure for the parent address set TLV.
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 [RFC2119].

The draft uses the following Terminology:

Packet Replication and Elimination (PRE): A method which transmits multiple copies of a packet using multi-path forwarding over a multi-hop network and which consolidates multiple received packet copies to control flooding. See "Exploiting Packet Replication and Elimination in Complex Tracks in 6TiSCH LLNs" [I-D.papadopoulos-6tisch-pre-reqs] for more details.

Alternative Parent (AP) Selection: The mechanism for choosing the next hop node to forward a packet copy when replicating packets.

3. Common Ancestor Objective Functions

In the RPL protocol, each node maintains a list of potential parents. For PRE, the Preferred Parent (PP) node is defined to be the same as the RPL DODAG Preferred Parent node. Furthermore, to construct an alternative path toward the root, in addition to the PP node, each node in the network registers an AP node as well from its Parent Set (PS).

There are multiple alternative methods of selecting the AP node. This functionality is included in the operation of the RPL Objective Function (OF). A group of OFs which allow the two paths to remain correlated is detailed here. More specifically, when using these OFs a node will select an AP node close to its PP node to allow the operation of overhearing between parents. For more details about overhearing and its use in this context see Section 4.3. "Promiscuous Overhearing" in "Exploiting Packet Replication and Elimination in Complex Tracks in 6TiSCH LLNs" [I-D.papadopoulos-6tisch-pre-reqs]. If multiple potential APs match this condition, the AP with the lowest rank will be registered.

The OFs described here are an extension of the The Minimum Rank with Hysteresis Objective Function [RFC6719] (MRHOF) OF. In general, these OFs extend MRHOF by specifying how an AP is selected. The selection of the PP is kept the same as in MRHOF.

The ways in which the CA OFs modify MRHOF in a section-by-section manner follows:
3. The Minimum Rank with Hysteresis Objective Function: Same as MRHOF extended to AP selection. Minimum Rank path selection and switching applies correspondingly to the AP with the extra CA requirement of having some match between ancestors, depending on the specific variant of CA OF used.

3.1. Computing the Path Cost: Same as MRHOF extended to AP selection. If a candidate neighbor does not fulfill the CA requirement then the path through that neighbor SHOULD be set to MAX_PATH_COST. As a result, the node MUST NOT select the candidate neighbor as its AP.

3.2. Parent Selection: Same as MRHOF extended to AP selection. To allow hysteresis, AP selection maintains a variable, `cur_ap_min_path_cost`, which is the path cost of the current AP.

3.2.1. When Parent Selection Runs: Same as MRHOF.

3.2.2. Parent Selection Algorithm: Same as MRHOF extended to AP selection. If the smallest path cost for paths through the candidate neighbors is smaller than `cur_ap_min_path_cost` by less than `PARENT_SWITCH_THRESHOLD`, the node MAY continue to use the current AP. Additionally, if there is no PP selected, there MUST NOT be any AP selected as well. Finally, as with MRHOF, a node MAY include up to `PARENT_SET_SIZE-1` additional candidate neighbors in its alternative parent set.

3.3. Computing Rank: Same as MRHOF.

3.4. Advertising the Path Cost: Same as MRHOF.

3.5. Working without Metric Containers: It is not possible to work without metric containers, since CA AP selection requires information from parents regarding their parent sets, which is transmitted via the NSA object in the DIO Metric Container.

4. Using MRHOF for Metric Maximization: Same as MRHOF.

5. MRHOF Variables and Parameters: Same as MRHOF extended to AP selection. The CA OFs operate like MRHOF for AP selection by maintaining separate:

   AP: Corresponding to the MRHOF PP. Hysteresis is configured for AP with the same `PARENT_SWITCH_THRESHOLD` parameter as in MRHOF. The AP MUST NOT be the same as the PP.

   Alternative parent set: Corresponding to the MRHOF parent set. The size is defined by the same `PARENT_SET_SIZE` parameter as in
MRHOF. The Alternative parent set MUST be a non-strict subset of the parent set.

cur_ap_min_path_cost: Corresponding to the MRHOF cur_min_path_cost variable. To support the operation of the hysteresis function for AP selection.

6. Manageability: Same as MRHOF.

6.1. Device Configuration: Same as MRHOF.

6.2. Device Monitoring: Same as MRHOF.

Three OFs are defined which perform AP selection based on common ancestors, named Common Ancestor Strict, Common Ancestor Medium, and Common Ancestor Relaxed, depending on how restrictive the selection process is. A more restrictive method will limit flooding but might fail to select an appropriate AP, while a less restrictive one will more often find an appropriate AP but might increase flooding.

All three OFs apply their corresponding common ancestor criterion to filter the list of candidate neighbours in the alternative parent set. The AP is then selected from the alternative parent set based on Rank and using hysteresis as is done for the PP in MRHOF.

3.1. Common Ancestor Strict

In the CA Strict OF, represented with Objective Code Point (OCP) TBD1, the node will check if its Preferred Grand Parent (PGP), the PP of its PP, is the same as the PP of the potential AP.
For example, in Figure 1, the source node $S$ must know its grandparent sets through nodes $A$, $B$, $C$, and $D$. The Parent Sets (PS) and the Preferred Parents (PP) of nodes $A$, $B$, $C$, and $D$ are shown on the side of the figure. The CA Strict parent selection method will select an AP for node $S$ for which $\text{PP}(\text{PP}(S)) = \text{PP}(\text{AP})$. Given that $\text{PP}(\text{PP}(S)) = Y$:

- Node $A$: $\text{PP}(A) = X$ and therefore it is different than $\text{PP}(\text{PP}(S))$
- Node $B$: $\text{PS}(B) = Y$ and therefore it is equal to $\text{PP}(\text{PP}(S))$
- Node $D$: $\text{PS}(D) = Z$ and therefore it is different than $\text{PP}(\text{PP}(S))$

Node $S$ can decide to use node $B$ as its AP node, since $\text{PP}(\text{PP}(S)) = Y = \text{PP}(B)$.

3.2. Common Ancestor Medium

In the CA Medium OF, represented with Objective Code Point (OCP) TBD2, the node will check if its Preferred Grand Parent (PGP), the PP of its PP, is contained in the PS of the potential AP.

Using the same example, in Figure 1, the CA Medium parent selection method will select an AP for node $S$ for which $\text{PP}(\text{PP}(S))$ is in $\text{PS}(\text{AP})$. Given that $\text{PP}(\text{PP}(S)) = Y$:
o Node A: PS(A) = {W, X} and therefore PP(PP(S)) is not in the set
o Node B: PS(B) = {W, X, Y} and therefore PP(PP(S)) is in the set
o Node D: PS(D) = {Y, Z} and therefore PP(PP(S)) is in the set

node S can decide to use node B or D as its AP node.

3.3. Common Ancestor Relaxed

In the CA Relaxed OF, represented with Objective Code Point (OCP) TBD3, the node will check if the Parent Set (PS) of its Preferred Parent (PP) has a node in common with the PS of the potential AP.

Using the same example, in Figure 1, the CA Relaxed parent selection method will select an AP for node S for which PS(PP(S)) has at least one node in common with PS(AP). Given that PS(PP(S)) = {X, Y, Z}:

o Node A: PS(A) = {W, X} and the common nodes are {X}
o Node B: PS(B) = {W, X, Y} and the common nodes are {X, Y}
o Node D: PS(D) = {Y, Z} and the common nodes are {Y, Z}

node S can decide to use node A, B or D as its AP node.

3.4. Usage

The PS information can be used by any of the described AP selection methods or other ones not described here, depending on requirements. It is optional for all nodes to use the same AP selection method. Different nodes may use different AP selection methods, since the selection method is local to each node. For example, using different methods can be used to vary the transmission reliability in each hop.

4. Node State and Attribute (NSA) object type extension

In order to select their AP node, nodes need to be aware of their grandparent node sets. Within RPL [RFC6550], the nodes use the DODAG Information Object (DIO) Control Message to broadcast information about themselves to potential children. However, RPL [RFC6550], does not define how to propagate parent set related information, which is what this document addresses.

DIO messages can carry multiple options, out of which the DAG Metric Container option [RFC6551] is the most suitable structurally and semantically for the purpose of carrying the parent set. The DAG Metric Container option itself can carry different nested objects,
out of which the Node State and Attribute (NSA) [RFC6551] is appropriate for transferring generic node state data. Within the Node State and Attribute it is possible to store optional TLVs representing various node characteristics. As per the Node State and Attribute (NSA) [RFC6551] description, no TLV has been defined for use. This document defines one TLV for the purpose of transmitting a node’s parent set.

Figure 2: Example DIO Message with a DAG Metric Container option

Figure 2 shows the structure of the DIO Control Message when a DAG Metric Container option is included. The DAG Metric Container option type (DAGMC Type in Figure 2) has the value 0x02 as per the IANA registry for the RPL Control Message Options, and is defined in [RFC6550]. The DAG Metric Container option length (DAGMC Length in Figure 2) expresses the DAG Metric Container length in bytes. DAG Metric Container data holds the actual data and is shown expanded in Figure 3.
The structure of the DAG Metric Container data in the form of a Node State and Attribute (NSA) object with a TLV in the NSA Optional TLVs field is shown in Figure 3. The first 32 bits comprise the DAG Metric Container header and all the following bits are part of the Node State and Attribute object body, as defined in [RFC6551]. This document defines a new TLV, which CAN be carried in the Node State and Attribute (NSA) object Optional TLVs field. The TLV is named Parent Set and is abbreviated as PS in Figure 3.

**Figure 3: DAG Metric Container (MC) data with Node State and Attribute (NSA) object body and a TLV**

**PS type:** The type of the Parent Set TLV. The value is TBD4.

**PS Length:** The total length of the TLV value field (PS IPv6 address(es)) in bytes.

4.1. Usage

The PS SHOULD be used in the process of parent selection, and especially in AP selection, since it can help the alternative path to not significantly deviate from the preferred path. The Parent Set is information local to the node that broadcasts it.

The PS is used only within NSA objects configured as constraints and is used as per [RFC6551].

5. Controlling PRE

PRE is very helpful when the aim is to increase reliability for a certain path, however its use creates additional traffic as part of the replication process. It is conceivable that not all paths have stringent reliability requirements. Therefore, a way to control whether PRE is applied to a path’s packets SHOULD be implemented. For example, a traffic class label can be used to determine this behavior per flow type as described in Deterministic Networking Architecture [I-D.ietf-detnet-architecture].
6. Security Considerations

The structure of the DIO control message is extended, within the pre-defined DIO options. Therefore, the security mechanisms defined in RPL [RFC6550] apply to this proposed extension.

7. IANA Considerations

This proposal requests the allocation of new values TBD1, TBD2, TBD3 from the "Objective Code Point (OCP)" sub-registry of the "Routing Protocol for Low Power and Lossy Networks (RPL)" registry. This proposal also requests the allocation of a new value TBD4 for the "Parent Set" TLV from the Routing Metric/Constraint TLVs sub-registry from IANA.

8. References

8.1. Informative references


8.2. Other Informative References


8.3. URIs


[2] https://code.wireshark.org/review/gitweb?p=wireshark.git;a=commit;h=e2f6ba229f45d8ccae2a6405e0ef41f1e61da138

Appendix A. Implementation Status

A research-stage implementation of the PRE mechanism using the proposed extension as part of a 6TiSCH IOT use case was developed at IMT Atlantique, France by Tomas Lagos Jenschke and Remous-Aris Koutsiamanis. It was implemented on the open-source Contiki OS and tested with the Cooja simulator. The DIO DAGMC NSA extension is implemented with a configurable number of parents from the parent set of a node to be reported.
The simulation setup is:

Topology: 32 nodes structured in regular grid as shown in Figure 4. Node S (source) is the only data packet sender, and sends data to node R (root). The parent set of each node (except R) is all the nodes in the immediately higher row, the immediately above 6 nodes. For example, each node in \{51, 52, 53, 54, 55, 56\} is connected to all of \{41, 42, 43, 44, 45, 46\}. Node 11, 12, 13, 14, 15, 16 have a single upwards link to R.

MAC: TSCH with 1 retransmission

Platform: Cooja

Schedule: Static, 2 timeslots per link from each node to each parent in its parent set, 1 broadcast EB slot, 1 sender-based shared timeslot (for DIO and DIS) per node (total of 32).

Simulation lifecycle: Allow link formation for 100 seconds before starting to send data packets. Afterwards, S sends data packets to R. The simulation terminates when 1000 packets have been sent by S.

Radio Links: Every 60 s, a new Packet Delivery Rate is randomly drawn for each link, with a uniform distribution spanning the 70% to 100% interval.
Traffic Pattern: CBR, S sends one non-fragmented UDP packet every 5 seconds to R.

PS extension size: 3 parents.

Routing Methods:

* RPL: The default RPL non-PRE implementation in Contiki OS.

* 2nd ETX: PRE with a parent selection method which picks as AP the 2nd best parent in the parent set based on ETX.

* CA Strict: As described in Section 3.1.

* CA Medium: As described in Section 3.2.

Simulation results:

<table>
<thead>
<tr>
<th>Routing Method</th>
<th>Average Packet Delivery Rate (%)</th>
<th>Average Traversed Nodes/packet (#)</th>
<th>Average Duplications/packet (#)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPL</td>
<td>82.70</td>
<td>5.56</td>
<td>7.02</td>
</tr>
<tr>
<td>2nd ETX</td>
<td>99.38</td>
<td>14.43</td>
<td>31.29</td>
</tr>
<tr>
<td>CA Strict</td>
<td>97.32</td>
<td>9.86</td>
<td>18.23</td>
</tr>
<tr>
<td>CA Medium</td>
<td>99.66</td>
<td>13.75</td>
<td>28.86</td>
</tr>
</tbody>
</table>

Links:

- Contiki OS DIO DAGMC NSA extension (draft-koutsiamanis-roll-nsa-extension branch) [1]

- Wireshark dissectors (for the optional PS TLV) - currently merged / in master [2]

Authors’ Addresses
Abstract

This specification leverages 6LoWPAN ND to provide a unicast and multicast routing service in a RPL domain to 6LNs that do not participate to RPL.

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 https://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 January 5, 2020.

Copyright Notice

Copyright (c) 2019 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 (https://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.
1. Introduction

The design of Low Power and Lossy Networks (LLNs) is generally focused on saving energy, which is the most constrained resource of all. Other design constraints, such as a limited memory capacity, duty cycling of the LLN devices and low-power lossy transmissions, derive from that primary concern.

The IETF produced the "Routing Protocol for Low Power and Lossy Networks" [RFC6550] (RPL) to provide routing services within such constraints. RPL is a Distance-Vector protocol, which, compared to link-state protocols, limits the amount of topological knowledge that needs to be installed and maintained in each node. In order to operate in constrained networks, RPL allows a Routing Stretch (see [RFC6687]), whereby routing is only performed along a DODAG as opposed to straight along a shortest path between 2 peers, whatever that would mean in a given LLN. This trades the quality of peer-to-peer (P2P) paths for a vastly reduced amount of control traffic and routing state that would be required to operate a any-to-any shortest path protocol. Finally, broken routes may be fixed lazily and on-
demand, based on dataplane inconsistency discovery, which avoids wasting energy in the proactive repair of unused paths.

In order to cope with lossy transmissions, RPL forms Direction-Oriented Directed Acyclic Graphs (DODAGs) using DODAG Information Solicitation (DIS) and DODAG Information Object (DIO) messages. For most of the nodes, though not all, a DODAG provides multiple forwarding solutions towards the Root of the topology via so-called parents. RPL is designed to adapt to fuzzy connectivity, whereby the physical topology cannot be expected to reach a stable state, with a lazy control that creates routes proactively but only fixes them when they are used by actual traffic. It results that RPL provides reachability for most of the LLN nodes, most of the time, but does not really converge in the classical sense. RPL provides unicast and multicast routing services back to RPL-Aware nodes (RANs). A RAN will inject routes to self using Destination Advertisement Object (DAO) messages sent to either their parents in Storing Mode or to the Root indicating their parent in Non-Storing Mode. This process effectively forms a DODAG back to the device that is a subset of the DODAG to the Root with all links reversed.

When a routing protocol such as RPL is used to maintain reachability within a Non-Broadcast Multi-Access (NBMA) subnet, some nodes may act as routers and participate to the routing operations whereas others may be plain hosts. In RPL terms, a plain host that does not participate to the routing protocol is called a Leaf. It must be noted that a 6LN could participate to RPL and inject DAO routes to self, but refrain from advertising DIO and get children. In that case, the 6LN is still a host but not a Leaf.

This specification enables a RPL-Unaware Leaf (RUL) to announce itself as a host and demand that the 6LR that accepts the registration also inject the relevant routing information for the Registered Address in the RPL domain on its behalf. The unicast packet forwarding operation by the 6LR serving a Leaf 6LN is described in "When to use RFC 6553, 6554 and IPv6-in-IPv6" [I-D.ietf-roll-useofrplinfo]. This document adds the capability by a 6LR to advertise the Global, Unique-Local and Multicast IPv6 address(es) of the 6LN in the RPL protocol.

Examples of routing-agnostic 6LN may include lightly-powered sensors such as window smash sensor (alarm system), or the kinetically powered light switch. Other application of this specification may include a smart grid network that controls appliances - such as washing machines or the heating system - in the home. Appliances may not participate to the RPL protocol operated in the smart grid network but can still receive control packet from the smart grid.
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 (LLNs). [RFC7102].

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

A glossary of classical 6LoWPAN 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" [RFC6550] specification.

This document introduces the term RPL-Unaware Leaf (RUL) 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 Leaf (RAL) is used to refer to a host or a router that participates to RPL and advertises its addresses of prefixes by itself.

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

Readers are expected to be familiar with all the terms and concepts that are discussed in

- "Neighbor Discovery for IP version 6" [RFC4861],
- "IPv6 Stateless Address Autoconfiguration" [RFC4862],
- "Problem Statement and Requirements for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing" [RFC6606],
2.3. Glossary

This document often uses the following acronyms:

AR: Address Resolution (aka Address Lookup)
6BBR: 6LoWPAN Backbone Router (proxy ND)
6LBR: 6LoWPAN Border Router (an Address Registrar that is authoritative on DAD)
6LN: 6LoWPAN Node (a Low Power host or router)
6LR: 6LoWPAN Router
6CIO: Capability Indication Option
(E)ARO: (Extended) Address Registration Option
(E)DAR: (Extended) Duplicate Address Request
(E)DAC: (Extended) Duplicate Address Confirmation
DAD: Duplicate Address Detection
DODAG: Destination-Oriented Directed Acyclic Graph
LLN: Low-Power and Lossy Network
NA: Neighbor Advertisement
NCE: Neighbor Cache Entry
ND: Neighbor Discovery
NDP: Neighbor Discovery Protocol
NS: Neighbor Solicitation
RA: Router Advertisement
ROVR: Registration Ownership Verifier (pronounced rover)
RPI: RPL Packet Information (an Option in the Hop-By-Hop header)
RAL: RPL-Aware Leaf
RS: Router Solicitation
RPL: IPv6 Routing Protocol for LLNs (pronounced ripple)
RUL: RPL-Unaware Leaf
TID: Transaction ID (a sequence counter in the EARO)

3. 6LoWPAN Neighbor Discovery

The IPv6 [RFC8200] Neighbor Discovery (IPv6 ND) Protocol (NDP) suite [RFC4861] [RFC4862] defined for fast media such as Ethernet, relies heavily on multicast operations for address discovery and duplicate address detection (DAD).

"Neighbor Discovery Optimizations for 6LoWPAN networks" [RFC6775] (6LoWPAN ND) adapts IPv6 ND for operations over energy-constrained LLNs. In particular, 6LoWPAN ND introduces a unicast host address registration mechanism that contributes to reduce the use of multicast messages that are present in the classical IPv6 ND protocol. 6LoWPAN ND defines a new Address Registration Option (ARO) that is carried in the unicast Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages between the 6LoWPAN Node (6LN) and the 6LoWPAN Router (6LR). 6LoWPAN ND also defines the Duplicate Address Request (DAR) and Duplicate Address Confirmation (DAC) messages between the 6LR and the 6LoWPAN Border Router (6LBR). In an LLN, the 6LBR is the central repository of all the Registered Addresses in its domain.

"Registration Extensions for 6LoWPAN Neighbor Discovery" [RFC8505] updates the behavior of RFC 6775 to enable a generic registration to routing services and defines an Extended ARO (EARO). The format of the EARO is shown in Figure 1:
Figure 1: EARO Option Format

The ‘R’ flag that is set if the Registering Node expects that the 6LR ensures reachability for the Registered Address, e.g., by means of routing or proxying ND.

The EARO also includes a sequence counter called Transaction ID (TID), which maps to the Path Sequence Field found in Transit Options in RPL DAO messages. It is a prerequisite for this specification.

Finally, the EARO transports an Opaque field and an ‘I’ field that describes what the Opaque field transports and how to use it. This specification requires that the I field is left to 0 and to use the Opaque field to carry the RPL InstanceID if one is known, else to leave the Opaque field to zero.

4. Updating RFC 6550

This document specifies a new behavior whereby a 6LR injects DAO messages for unicast addresses registered through the updated 6LoWPAN ND [RFC8505] on behalf of 6LN nodes that are not RPL-aware.

Upon the renewal of a 6LoWPAN ND registration, this specification changes the behavior of the 6LR as follows. If the ‘R’ flag is set, the 6LR injects a DAO targeting the Registered Address, and refrains from sending a DAR message. the DAR/DAC exchange that refreshes the state in the 6LBR happens instead between the RPL Root and the 6LBR. In that flow, the RPL Root acts as a proxy on behalf of the 6LR upon the reception of the DAO propagation initiated at the 6LR.

5. Updating RFC 8505

The behavior defined in this specification whereby the 6LR that processes the registration advertises the Registered Address in DAO messages and bypasses the DAR/DAC process for the renewal of a registration, is only triggered by an NS(EARO) that has the ‘R’ flag
set. If the ‘R’ flag is not set, then the Registering Node is expected to be a RAN router that handles the reachability of the Registered Address by itself.

This document also specifies a keep-alive EDAR message that the RPL Root may use to maintain an existing state in the 6LBR upon receiving DAO messages. The keep-alive EDAR message may only act as a refresher and can only update the Lifetime and the TID of the state in the 6LBR.

This document similarly specifies a keep-alive NS(EARO) message that the RPL Root may use to maintain an existing state in a 6BBR upon receiving DAO messages. The keep-alive NS(EARO) message may only act as a refresher and can only update the Lifetime and the TID of the state in the 6BBR.

As prescribed by [RFC8505], a RPL router SHOULD NOT set the ‘R’ flag.

6. Dependencies on the 6LN

This document provides RPL routing for a 6LN acting as a plain host and not aware of RPL. Still, a minimal RPL-independent functionality is expected from the 6LN in order to operate properly as a RLU; in particular:

- The 6LN MUST implement [RFC8505] and set the ‘R’ flag in the EARO option. The ‘R’ flag is used to determine whether the Registering Node is a RUL, not aware of the RPL operation in the network, and thus does not participate to it. A 6LN is considered to be a RUL if and only if it sets the ‘R’ flag in the EARO.

- RPL data packets are often encapsulated using IP in IP and in Non-Storing Mode, packets going down will carry an SRH as well. RPL data packets also typically carry a Hop-by-Hop Header to transport a RPL Packet Information (RPI) [RFC6550]. These additional headers are called RPL artifacts.

- An arbitrary 6LN is expected to support IPv6-in-IPv6 encapsulation when it is the destination of the outer header. If the 6LN is a host, it is expected to drop the inner packet if it is not the destination of the inner header.

- An arbitrary 6LN is expected to process an unknown Option Type in a Hop-by-Hop Header as prescribed by section 4.2 of [RFC8200]. This means in particular that an RPI with an Option Type of 0x23 [I-D.ietf-roll-useofrplinfo] is ignored when not understood.
An arbitrary 6LN is expected to process an unknown Routing Header Type as prescribed by section 4.4 of [RFC8200]. This means in particular that Routing Header with a Routing Type of 3 [RFC6553] is ignored when the Segments Left is zero, and dropped otherwise.

When IP-in-IP is used and the outer headers terminate at the 6LR that generated the DAO, then the 6LR decapsulates the packet to the 6LN (see Appendix A for the format in Storing Mode). In that case the 6LN gets a packet that is free of RPL artifacts. IP-in-IP to the 6LR MUST be used if the 6LN cannot handle or ignore the RPL artifacts or the way they are compressed [RFC8138]. It SHOULD be used if there is a particular bandwidth or power constraint at the 6LN that justifies saving the encapsulation at the last hop.

In order to save the IP-in-IP encapsulation and to support Storing Mode of operation, it is preferred that the 6LN can ignore an RPI and consume a routing header in both the native and compressed forms. In order to enable IP-in-IP to a 6LN in Non-Storing Mode, it is also of interest that the 6LN supports decapsulating IP-in-IP in both forms. But since the preferred behaviour when using IP-in-IP is that the outer headers terminate at the 6LR, supporting this capability is secondary.

7. Protocol Operations for Unicast Addresses

7.1. General Flow

This specification enables to save the exchange of Extended Duplicate Address messages, EDAR and EDAC, from a 6LN all the way to the 6LBR across a RPL mesh, for the sole purpose of refreshing an existing state in the 6LBR. Instead, the EDAR/EDAC exchange is proxied by the RPL Root upon a DAO message that refreshes the RPL routing state. To achieve this, the lifetimes and sequence counters in 6LoWPAN ND and RPL are aligned. In other words, the Path Sequence and the Path Lifetime in the DAO message are derived from the Transaction ID and the registration lifetime in the NS(EARO) message from the 6LN.

From the perspective of the 6LN, the registration flow happens transparently; it is not delayed by the proxy RPL operation, so the device does not need to wait more whether RPL proxy operation happens or not. The flows below are RPL Non-Storing Mode examples. In Storing Mode, the DAO ACK may not be present, and the DAO messages cascade from child to parent all the way to the DODAG Root.

On the first registration, illustrated in Figure 2, from the perspective of the 6LR in Non-Storing Mode, the Extended Duplicate Address message takes place as prescribed by [RFC8505]. When successful, the flow creates a Neighbor Cache Entry (NCE) in the 6LR,
and the 6LR injects the Registered Address in RPL using DAO/DAO-ACK exchanges all the way to the RPL DODAG Root. The protocol does not carry a specific information that the Extended Duplicate Address messages were already exchanged, so the Root proxies them anyway. Note that in Storing Mode the DAO ACK is generated from the parent that does not necessary wait for the grand parent to acknowledge, so the DAO-ACK is no guarantee that the keep-alive EDAR succeeded. On the other hand, the flows can be nested in Non-Storing Mode, and it is possible to carry information such as an updated lifetime from the 6LBR all the way to the 6LN.

Figure 2: First Registration Flow

A re-registration is performed by the 6LN to maintain the NCE in the 6LR alive before lifetime expires. Upon a re-registration, as illustrated in Figure 3, the 6LR redistributes the Registered Address NS(EARO) in RPL. This causes the RPL DODAG Root to refresh the state in the 6LBR with a keep-alive EDAC message. The keep-alive EDAC lacks the Registration Ownership Verifier (ROVR) information, since it is not present in RPL DAO messages, but the EDAC message sent in response by the 6LBR contains the actual value of the ROVR field for that registration. This enables the RPL Root to perform the proxy-registration for the Registered Address and attract traffic captured over the backbone by the 6BBR and route it back to the device.
### 7.2. 6LN Operation

This specification does not alter the operation of a 6LoWPAN ND-compliant 6LN, which is expected to operate as follows:

- The 6LN obtains an IPv6 global address, for instance using autoconfiguration [RFC4862] based on a Prefix Information Option (PIO) [RFC4861] found in a Router Advertisement message or by some other means such as DHCPv6 [RFC3315].

- Once it has formed an address, the 6LN (re)registers its address periodically, within the Lifetime of the previous registration, as prescribed by [RFC8505].

- Upon each consecutive registration, the 6LN MUST increase the TID field.

---

**Figure 3: Next Registration Flow**

Note that any of the functions 6LR, Root and 6LBR might be collapsed in a single node, in which case the flow above happens internally, and possibly through internal API calls as opposed to messaging.
o If the 6LN is aware of the RPL Instance the packet should be injected into, then it SHOULD set the Opaque field to the InstanceID, else it MUST leave the Opaque field to zero. In any fashion the 6LN MUST set the ‘I’ field to zero.

o A 6LN acting as a RUL MUST set the ‘R’ flag in the EARO whereas a 6LN acting as a RAN SHOULD NOT set the ‘R’ flag.

o The 6LN MAY register to more than one 6LR at the same time. In that case, a same value of TID is used for each registration.

o The 6LN MAY use any of the 6LRs to which it register to forward its packets.

o the 6LN is not expected to be aware of RPL so it is not expected to produce RPL artifacts in the data packets.

7.3. 6LR Operation

Also as prescribed by [RFC8505], the 6LR generates a DAR message upon reception of a valid NS(EARO) message for the registration of a new IPv6 Address by a 6LN. If the Duplicate Address exchange succeeds, then the 6LR installs a Neighbor Cache Entry (NCE). If the ‘R’ flag was set in the EARO of the NS message, and this 6LR can manage the reachability of Registered Address, then the 6LR sets the ‘R’ flag in the ARO of the response NA message.

From then on, the 6LN periodically sends a new NS(EARO) to refresh the NCE state before the lifetime indicated in the EARO expires, with TID that is incremented each time till it wraps in a lollipop fashion. As long as the ‘R’ flag is set and this router can still manage the reachability of Registered Address, the 6LR keeps setting the ‘R’ flag in the EARO of the response NA message, but the exchange of Extended Duplicate Address messages is skipped.

The Opaque field in the EARO hints the 6LR on the RPL Instance that should be used for the DAO advertisements, and for the forwarding of packets sourced at the registered address when there is no RPL Packet Information (RPI) in the packet, in which case the 6LR SHOULD add one to the packet. if the ‘I’ field is not zero, then the 6LR MUST consider that the Opaque field is left to zero. If the Opaque field is not set to zero, then it should carry a RPL InstanceID for the Instance suggested by the 6LN. If the 6LR does not participate to the associated Instance, then the 6LR MUST consider that the Opaque field is left to zero. If the Opaque field left to zero, the 6LR is free to use the default Instance (zero) for the registered address or to select an Instance of its choice; else, that is if the 6LR
participates to the suggested Instance, then the 6LR SHOULD use that
Instance for the registered address.

Upon a successful NS/NA(EARO) exchange: if the ‘R’ flag was set in
the EARO of the NS message, then the 6LR SHOULD inject the Registered
Address in RPL by sending a DAO message on behalf of the 6LN; else
the 6LR MUST NOT inject the Registered Address into RPL.

The DAO message advertising the Registered Address MUST be
constructed as follows:

- The Registered Address is placed in a RPL Target Option in the DAO
  message as the Target Prefix, and the Prefix Length is set to 128
- The External ‘E’ flag in the Transit Information Option (TIO)
  associated to the Target Option is set to indicate that the 6LR
  redistributes an external target into the RPL network. This is
  how the Root knows in Non-Storing Mode to use IP-in-IP and
  terminate the outters headers at the 6LR that generated the DAO.
- The Path Lifetime in the TIO is computed from the Lifetime in the
  EARO Option to adapt it to the Lifetime Units used in the RPL
  operation. Note that if the lifetime is 0, then the 6LR generates
  a No-Path DAO message that cleans up the routes down to the
  Address of the 6LN.
- The Path Sequence in the TIO is set to the TID value found in the
  EARO option.
- Additionally, in Non-Storing Mode the 6LR indicates one of its
  global IPv6 unicast addresses as the Parent Address in the TIO.

If a 6LR receives a valid NS(EARO) message with the ‘R’ flag reset
and the 6LR was redistributing the Registered Address due to previous
NS(EARO) messages with the flag set, then it MUST stop injecting the
address. It is up to the Registering Node to maintain the
respective route from then on, either keeping it active by sending
further DAO messages, or destroying it using a No-Path DAO.

7.4.  RPL Root Operation

In RPL Storing Mode of Operation (MOP), the DAO message is propagated
from child to parent all the way to the Root along the DODAG,
populating routing state as it goes. In Non-Storing Mode, The DAO
message is sent directly to the route. Upon reception of a DAO
message that creates or updates an existing RPL state:
the Root notifies the 6LBR using an internal API if they are collocated, or performs a keep-alive DAR/DAC exchange on behalf of the registering node if they are separated.

In an extended topology with a Backbone Link, the Root notifies the 6LBR by proxying a keep-alive NS(EARO) on behalf of the 6LN that owns the address indicated in the Target Option.

The keep-alive EDAR and the NS(EARO) messages MUST be constructed as follows:

- The Target IPv6 address from in the RPL Target Option is placed in the Registered Address field of the EDAR message and in the Target field of the NS message, respectively.

- the ROVR field in the keep-alive EDAR is set to 64-bits of all ones to indicate that it is not provided and this is a keep-alive EDAR. The actual value of the ROVR for that registration is returned by the 6LBR in an EDAC, and used in the proxy NS(EARO).

- the Registration Lifetime is adapted from the Path Lifetime in the TIO by converting the Lifetime Units used in RPL into units of 60 seconds used in the 6LoWPAN ND messages.

- The RPL Root indicates its own MAC Address as Source Link Layer Address (SLLA) in the NS(EARO).

- the TID value is set to the Path Sequence in the TIO. The ‘T’ flag and an ICMP code of 1 are used in the NS(EARO) and the DAR message, respectively.

Upon a status in a DAC message that is not "Success", the Root MAY destroy the formed paths using a No-Path DAO downwards as specified in [I-D.ietf-roll-efficient-npdao].

In Non-Storing Mode, the outer IPv6 header that is used by the Root to transport the source routing information in data packets down the DODAG has the 6LR that serves the 6LN as final destination. This way, when the final 6LR decapsulates the outer header, it also removes all the RPL artifacts from the packet.

7.5. 6LBR Operation

Upon reception of a DAR message with the Owner Unique ID field is set to all ones, the 6LBR checks whether an entry exists for the and computes whether the TID in the DAR message is fresher than that in the entry as prescribed in section 4.2.1. of [RFC8505].
If the entry does not exist, the 6LBR does not create the entry, and answers with a Status "Removed" in the DAC message.

If the entry exists but is not fresher, the 6LBR does not update the entry, and answers with a Status "Success" in the DAC message.

If the entry exists and the TID in the DAR message is fresher, the 6LBR updates the TID in the entry, and if the lifetime of the entry is extended by the Registration Lifetime in the DAR message, it also updates the lifetime of the entry. In that case, the 6LBR replies with a Status "Success" in the DAC message.

8. Protocol Operations for Multicast Addresses

Section 12 of [RFC6550] details the RPL support for multicast flows. This support is not source-specific and only operates as an extension to the Storing Mode of Operation for unicast packets. Note that it is the RPL model that the multicast packet is passed as a Layer-2 unicast to each if the interested children. This remains true when forwarding between the 6LR and the listener 6LN.

"Multicast Listener Discovery (MLD) for IPv6" [RFC2710] and its updated version "Multicast Listener Discovery Version 2 (MLDv2) for IPv6" [RFC3810] provide an interface for a listener to register to multicast flows. MLDv2 is backwards compatible with MLD, and adds in particular the capability to filter the sources via black lists and white lists. In the MLD model, the router is a "querier" and the host is a multicast listener that registers to the querier to obtain copies of the particular flows it is interested in.

On the first registration, as illustrated in Figure 4, the 6LN, as an MLD listener, sends an unsolicited Report to the 6LR in order to start receiving the flow immediately. Since multicast Layer-2 messages are avoided, it is important that the asynchronous messages for unsolicited Report and Done are sent reliably, for instance using an Layer-2 acknowledgement, or attempted multiple times.

The 6LR acts as a generic MLD querier and generates a DAO for the multicast target. The lifetime of the DAO is set to be in the order of the Query Interval, yet larger to account for variable propagation delays.

The Root proxies the MLD exchange as listener with the 6BBR acting as the querier, so as to get packets from a source external to the RPL domain. Upon a DAO with a multicast target, the RPL Root checks if it is already registered as a listener for that address, and if not, it performs its own unsolicited Report for the multicast target.
A re-registration is pulled by 6LR acting as querier. Note that the message may be sent unicast to all the known individual listeners. Upon a timeout of the Query Interval, the 6LR sends a Query to each of its listeners, and gets a Report back that is mapped into a DAO, as illustrated in Figure 5,

```
6LN       6LR       Root       6LBR
Query    <------------ Dao
          |              Dao ACK
          |<-------------< unsolicited Report
          |                 <if not listening>
          |                    >
```

Figure 5: Next Registration Flow

Note that any of the functions 6LR, Root and 6LBR might be collapsed in a single node, in which case the flow above happens internally, and possibly through internal API calls as opposed to messaging.
9. Implementation Status

10. Security Considerations

The LLN nodes depend on the 6LBR and the RPL participants for their operation. A trust model must be put in place to ensure that the right devices are acting in these roles, so as to avoid threats such as black-holing, or bombing attack whereby an impersonated 6LBR would destroy state in the network by using the "Removed" Status code. This trust model could be at a minimum based on a Layer-2 access control, or could provide role validation as well. This is a generic 6LoWPAN requirement, see Req5.1 in Appendix of [RFC8505].

The keep-alive EDAR message does not carry a valid Registration Unique ID [RFC8505] and it cannot be used to create a binding state in the 6LBR. The 6LBR MUST NOT create an entry based on a keep-alive EDAR that does not match an existing entry. All it can do is refresh the lifetime and the TID of an existing entry.

11. IANA Considerations

This specification has no requirement on IANA.

12. Acknowledgments

The author wishes to thank Michael Richardson and Georgios Papadopoulos for their early reviews of and contributions to this document.

13. References

13.1. Normative References


Internet-Draft                Routing for RPL Leaves                July 2019

[RFC4861]  Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
"Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,
DOI 10.17487/RFC4861, September 2007,

Address Autoconfiguration", RFC 4862,
DOI 10.17487/RFC4862, September 2007,

[RFC4919]  Kushalnagar, N., Montenegro, G., and C. Schumacher,
"IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs):
Overview, Assumptions, Problem Statement, and Goals",
RFC 4919, DOI 10.17487/RFC4919, August 2007,

Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur,
Low-Power and Lossy Networks", RFC 6550,
DOI 10.17487/RFC6550, March 2012,

[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,

[RFC6606]  Kim, E., Kaspar, D., Gomez, C., and C. Bormann,
"Problem Statement and Requirements for IPv6 over Low-Power
Wireless Personal Area Network (6LoWPAN) Routing",
RFC 6606, DOI 10.17487/RFC6606, May 2012,

Bormann, "Neighbor Discovery Optimization for IPv6 over
Low-Power Wireless Personal Area Networks (6LoWPANs)",
RFC 6775, DOI 10.17487/RFC6775, November 2012,

[RFC8138]  Thubert, P., Ed., Bormann, C., Toutain, L., and R. Cragie,
"IPv6 over Low-Power Wireless Personal Area Network
(6LoWPAN) Routing Header", RFC 8138, DOI 10.17487/RFC8138,
13.2. Informative References

[I-D.ietf-roll-efficient-npdao]

[I-D.ietf-roll-useofrplinfo]

[RFC3315]

[RFC6282]

[RFC6687]

[RFC7102]
Appendix A.  Example Compression

Figure 6 illustrates the case in Storing mode where the packet is received from the Internet, then the root encapsulates the packet to insert the RPI and deliver to the 6LR that is the parent and last hop to the final destination, which is not known to support [RFC8138]. The difference with the format presented in Figure 19 of [RFC8138] is the addition of a SRH-6LoRH before the RPI-6LoRH to transport the destination address of the outer IPv6 header.

```
+--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ...+-+
|      1|11110001|SRH-6LoRH| RPI- | IP-in-IP | NH=1 |11110CPP| UDP | UDP |
|Page 1 | Type1 S=0| 6LoRH |  6LoRH   |LOWPAN_IPHC| UDP    | hdr |Payld |
+--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ... +--------+ ...+-+
```

In Figure 6, the source of the IP-in-IP encapsulation is the Root, so it is elided in the IP-in-IP 6LoRH. The destination is the parent 6LR of the destination of the inner packet so it cannot be elided. In Storing Mode, it is placed as the single entry in an SRH-6LoRH as the first 6LoRH. Since there is a single entry so the SRH-6LoRH Size is 0. In this particular example, the 6LR address can be compressed to 2 bytes so a Type of 1 is used. It results that the total length of the SRH-6LoRH is 4 bytes.

In Non-Storing Mode, the encapsulation from the root would be similar to that represented in Figure 6 with possibly more hops in the SRH-6LoRH and possibly multiple SRH-6LoRHs if the various addresses in the routing header are not compressed to the same format. Note that on the last hop to the parent 6LR, the RH3 is consumed and removed from the compressed form, so the use of Non-Storing Mode vs. Storing Mode is indistinguishable from the packet format.
Follows the RPI-6LoRH and then the IP-in-IP 6LoRH. When the IP-in-IP 6LoRH is removed, all the router headers that precede it are also removed.

The Paging Dispatch [RFC8025] may also be removed if there was no previous Page change to a Page other than 0 or 1, since the LOWPAN_IPHC is encoded in the same fashion in the default Page 0 and in Page 1. The resulting packet to the destination is the inner packet compressed with [RFC6282].

Author’s Address

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

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

RPL allows different mode of operations which allows nodes to have a consensus on the basic primitives that must be supported to join the network. The MOP field in RFC6550 is of 3 bits and is fast depleting. This document extends the MOP field specification and adds a notion of capabilities using which the nodes can further advertise their support for, possibly optional, capabilities.

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 https://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 December 11, 2019.

Copyright Notice

Copyright (c) 2019 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 (https://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

Jadhav & Thubert Expires December 11, 2019 [Page 1]
1. Introduction

RPL [RFC6550] specifies a proactive distance-vector based routing scheme. The protocol creates a DAG-like structure which operates with a given "Mode of Operation" (MOP) determining the minimal and mandatory set of primitives to be supported by all the participating nodes.

MOP as per [RFC6550] is a 3-bit value carried in DIO messages and is specific to the RPL Instance. The recipient of the DIO message can join the specified network as a router only when it can support the primitives as required by the mode of operation value. For example, in case of MOP=3 (Storing MOP with multicast support) the nodes can join the network as routers only when they can handle the DAO advertisements from the peers and manage routing tables. The 3-bit value is already exhausted and requires replenishment. This document introduces a mechanism to extend mode of operation values.

This document further adds a notion of capabilities using which the nodes in the network could inform its peers about its additional
capabilities/features. This document highlights the differences of capabilities from that of Mode of operation and explains the necessity of it.

1.1. Requirements Language and 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 RFC 2119 [RFC2119].

MOP: Mode of Operation. Identifies the mode of operation of the RPL Instance as administratively provisioned at and distributed by the DODAG root.

MOPex: Extended MOP: This document extends the MOP values over a bigger range. This extension of MOP is called MOPex.

Capabilities: Additional features or capabilities which might possibly be optional that are supported by the node.

DAO: DODAG Advertisement Object. An RPL message used to advertise the target information in order to establish routing adjacencies.

DIO: DODAG Information Object. An RPL message initiated by the root and is used to advertise the network configuration information.

Current parent: Parent 6LR node before switching to the new path.

NPDAO: No-Path DAO. A DAO message which has target with lifetime 0.

MOPex: MOP extension as defined in this document.

This document uses terminology described in [RFC6550]. For the sake of readability all the known relevant terms are repeated in this section.

2. Requirements for this document

Following are the requirements considered for this documents:

REQ1: MOP extension. Current MOP of 3-bit is fast depleting. An MOP extension needs to extend the possibility of adding new MOPs in the future.

REQ2: Backwards compatibility. The new options and new fields in the DIO message should be backward compatible i.e. if there are nodes which support old MOPs they could still operate in their own instances.
REQ3: Optional capabilities handshake. Capabilities are features, possibly optional, which could be handshaked between the nodes and the root within an RPL Instance.

REQ4: Capabilities handshake could be optionally added with existing MOPs. Capabilities been optional in nature could be put to use with existing MOPs. Capabilities and MOP-extension is mutually independent i.e. a DIO can have a capabilities option, MOP-extension option or both in the same message.

3. Extended MOP Control Message Option

This document reserves existing MOP value 7 to be used as an extender. DIO messages with MOP value of 7 may refer to the Extended MOP (MOPex) option in the DIO message.

```
+----------+-------+-----------+
| Base MOP | MOPex | Final MOP |
|----------+-------+-----------|
| 0        | NA    | 0         |
| 1        | NA    | 1         |
|          | :     | :         |
| 6        | NA    | 6         |
| 7        | 0     | 7         |
| 7        | 1     | 8         |
| 7        | 2     | 9         |
|          | :     | :         |
```

Table 1: Final MOP calculation
3.2. Handling MOPex

If the MOPex option is absent in the DIO whose MOP is 7, then the MOPex value can be assumed to be zero (thus the final MOP in this case will be 7). The MOPex value should be referred only if the base MOP value is 7 and if the MOPex option is present. In case the base MOP is 7 and if the MOPex option is present, then the implementation MUST calculate the final MOP after considering the value in MOPex.

Note that [RFC6550] allows the node who does not support the received MOP to still join the network as a leaf node. This semantic continues to be true even in case of MOPex.

4. Capabilities

Currently RPL specification does not have a mechanism whereby a node can signal the set of features that are available on its end. Such a mechanism could help the root to advertise its capabilities and in response also determine some advanced information about the capabilities of the joining nodes. The Mode of Operation field in RPL mandates the operational requirement and does not allow loose coupling of additional capabilities. This document defines Capabilities as additional features which could be supported by the nodes and handshaked as part of RPL signaling. Capabilities are embedded as RPL control message option as defined Section 6.7 of [RFC6550] in the base messages of DIO, DAO and DAO-ACK signaling.

Note that capabilities and MOPex are mutually exclusive and it is possible for an implementation to support either or both of the options.

4.1. Capability Control Message Option

```
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = TODO  |           Capabilities Flags                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Capabilities Option

There are no capability flags defined by this document.

4.2. Capabilities Handshake

The root node could advertise the set of capabilities it supports in the DIO message. A node could take advantage of the knowledge that the root supports a particular capability. Similarly a node could
advertise its capabilities in the DAO message using the capability control message option defined in this document. Capabilities advertised by non-root nodes are strictly a subset of the capabilities advertised by the root.

In storing MOP, the DAO message from the 6LR could contain multiple target options. The targets of the capabilities option are indicated by one or more Target options that precede the Capabilities Option. This handling is similar to the Transit Information Option as supported in Section 6.7.8. of [RFC6550].

5. Implementations Consideration

The MOP-extension could cause 3-byte increase in memory in the RPL-Instance. The MOP field in the RPL-Instance needs to be upgraded to a 32 bit integer.

[RFC6550], it was possible to discard an unsupported DIO-MOP just by inspecting the base message. With this document, the MOPex is a different control message option and thus the discarding of the DIO message could happen after inspecting the message options.

A node in storing MOP could independently construct a DAO message with target options containing its child/sub-childs. Thus with capabilities it needs to reconstruct the capabilities field as well. This may result in increase in the memory requirement on per routing-entry basis.

6. Acknowledgements

Thanks to Georgios Papadopoulos for the review and feedback.

7. IANA Considerations

7.1. Mode of operation: MOPex

IANA is requested to assign a new Mode of Operation, named "MOPex" for MOP extension under the RPL registry. The value of 7 is to be assigned from the "Mode of Operation" space [RFC6550]

+-----------------+-----------------+-----------------+
| Value | Description | Reference       |
+-----------------+-----------------+-----------------+
| 7 | MOPex | This document |
+-----------------+-----------------+-----------------+

Mode of Operation
7.2. New options: MOPex and Capabilities

Two new entries are required for new supporting new options "MOPex", "Capabilities" from the "RPL Control Message Options" space [RFC6550].

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD1</td>
<td>MOPex</td>
<td>This document</td>
</tr>
<tr>
<td>TBD2</td>
<td>Capabilities</td>
<td>This document</td>
</tr>
</tbody>
</table>

7.3. New Registry for Extended-MOP-value

IANA is requested to create a registry for the extended-MOP-value (MOPex). This registry should be located in TODO. New MOPex values may be allocated only by an IETF review. Currently no values are defined by this document. Each value is tracked with the following qualities:

- MOPex value
- Description
- Defining RFC

7.4. New Registry for Capabilities Flags

IANA is requested to create a registry for the Capabilities flags as described in Section 4 of this document. This registry should be located in TODO. New Capabilities flags may be allocated only by an IETF review. Currently no flags are defined by this document. Each value is tracked with the following qualities:

- Flag
- Description
- Defining RFC

8. Security Considerations

The options defined in this document are carried in the base message objects as defined in [RFC6550]. The RPL control message options are
protected by the same security mechanisms that protect the base messages.

Capabilities flag can reveal that the node has been upgraded or is running a old feature set. This document assumes that the base messages that carry these options are protected by RPL security mechanisms and thus are not visible to a malicious node.

9. References

9.1. Normative References


9.2. Informative References


Appendix A. Capability Handshake Example

```
Root          6LR          6LN
|             |            |            |
|------------|>            |------------|
|   DIO(CS1) |            |   DAO(CS2) |
|------------|            |<-----------|
|   DAO(CS2) |            |            |
|<------------|            |

CS: Capabilities Set
CS1: Capabilities set advertised by root
CS2: Capabilities set advertised by node. CS2 is a subset of CS1.

Figure 3: Capabilities Option
```
Authors’ Addresses

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

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

Pascal Thubert
Cisco Systems, Inc
Building D
45 Allee des Ormes - BP1200
MOUGINS - Sophia Antipolis 06254
France

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

This document complements RFC 8138 and dedicates a bit in the RPL configuration option defined in RFC 6550 to indicate whether RFC 8138 compression is used within the RPL instance.

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 https://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 January 9, 2020.

Copyright Notice

Copyright (c) 2019 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 (https://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.
1. Introduction

The transition to [RFC8138] in a network can only be done when all
nodes support the specification. In a mixed case with both
RFC8138-capable and non-capable nodes, the compression should be
turned off.

This document complements RFC 8138 and dedicates a bit in the RPL
configuration option to indicate whether RFC 8138 compression should
be used within the RPL instance. When the bit is not set, source
nodes that support RFC 8138 should refrain from using the compression
unless the information is superseded by configuration.

2. 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.

3. Updating RFC 6550

RPL defines a configuration option that is registered to IANA in
section 20.14. of [RFC6550]. This specification defines a new flag
"Enable RFC8138 Compression" (T) that is encoded in one of the
reserved control bits in the option. The new flag is set to turn on
the use of the compression of RPL artifacts with RFC 8138.
4. Updating RFC 8138

This document specifies controls that enable and disable the use of the [RFC8138] compression in a RPL Instance. Arguably, this could have been done in [RFC8138] itself.

A node that supports this specification SHOULD source packets in the compressed form using [RFC8138] if the new "T" flag is set in the RPL configuration option from its parents. Failure to do so will result in larger packets, yields higher risks of loss and may cause a fragmentation.

A node that supports this specification SHOULD refrain from sourcing packets in the compressed form using [RFC8138] if the "T" flag is reset. This behavior can be overridden by a configuration of the node in order to cope with intermediate implementations of the root that support [RFC8138] but not this specification and cannot set the "T" flag.

The decision of using RFC 8138 to compress a packet is made at the source depending on its capabilities and its knowledge of the state of the "T" flag. A router MUST forward the packet in the form that the source used, either compressed or uncompressed. A router that encapsulates a packet is the source of the resulting packet and the rules above apply to it in that case.

5. Transition Scenarios

It is RECOMMENDED to only deploy nodes that support [RFC8138] in a network where the compression is turned on. A node that does not support [RFC8138] MUST only be used as a leaf in that network.

[RFC6550] states that "Nodes other than the DODAG root MUST NOT modify this information when propagating the DODAG Configuration option". In other words, the configuration option is a way for the root to configure the LLN nodes but it cannot be used by a parent to advertise its capabilities down the DODAG. It results whether a parent supports RFC 8138 is not known by the child with the current level of specifications, and a child cannot favor a parent based on a particular support.

Sections 8.5 and 9.2 of [RFC6550] also suggests that a RPL-aware node may attach to a DODAG as a leaf node only, e.g., when a node does not support the Mode of Operation of a RPL Instance, the Objective Function (OF) as indicated by the Objective Code Point (OCP) or some other parameters in the configuration option. But the node is also free to refrain from joining an Instance when a parameter is not suitable. This means that changing the OCP in a DODAG can be used to
force nodes that do not support a particular feature to join as leaf only. This specification reiterates that a node that is configured to operate in an Instance but does not support a value for a known parameter that is mandatory for routing MUST NOT operate as a router but MAY still joins as a leaf. Note that a legacy node will not recognize when a reserved field is now used and will not turn to a leaf when that happens.

A node that supports [RFC8138] but not this specification can only be used in an homogeneous network and an upgrade requires a "flag day" where all nodes are updated and then the network is rebooted with implicitly RFC 8138 compression turned on with the "T" flag set on.

A node that supports this specification can work in a network with RFC 8138 compression turned on or off with the "T" flag set accordingly and in a network in transition from off to on or on to off (see Section 5.1).

A node that does not support [RFC8138] can interoperate with a node that supports this specification in a network with RFC 8138 compression turned off. But it cannot forward compressed packets and therefore it cannot act as a router in a network with RFC 8138 compression turned on. It may remain connected to that network as a leaf and generate uncompressed packets as long as incoming packets are decapsulated by the parent and delivered in uncompressed form.

The intent for this specification is to perform a migration once and for all without the need for a flag day. In particular it is not the intention to undo the setting of the "T" flag, and though it is possible to roll back (see Section 5.4), adding nodes that do not support [RFC8138] after a roll back may be problematic if the roll back is not fully complete (see caveats in Section 5.2).

5.1. Inconsistent State While Migrating

When the "T" flag is turned on in the configuration option by the root, the information slowly percolates through the DODAG as the DIO gets propagated. Some nodes will see the flag and start sourcing packets in the compressed form while other nodes in the same instance are still not aware of it. Conversely, in non-storing mode, the root will start using RFC 8138 with a SRH-6LoRH that routes all the way to the last router or possibly to the leaf, if the leaf supports RFC 8138.

This is why it is required that all the routers in the Instance support [RFC8138] at the time of the switch, and all nodes that do not support [RFC8138] only operate as leaves.
Setting the "T" flag is ultimately the responsibility of the network administrator. In a case of upgrading a network to turn the compression on, the network SHOULD be operated with the "T" flag reset until all targeted nodes are upgraded to support this specification. Section 5.2 and Section 5.3 provide possible transition scenarios where this can be enforced.

5.2. Single Instance Scenario

In a single instance scenario, nodes that support RFC 8138 are configured with a new OCP, that may use the same OF operation or a variation of it. When it finally sets the "T" flag, the root also migrates to the new OCP. As a result, nodes that do not support RFC 8138 join as leaves and do not forward packets anymore. The leaves generate packets without compression. The parents - which supports RFC 8138 - may encapsulate the packets using RFC 8138 if needed. The other way around, the root encapsulates packets to the leaves all the way to the parent, which decapsulates and distribute the uncompressed inner packet to the leaf, as illustrated in Section 4.3 of [I-D.ietf-roll-useofrplinfo].

This scenario presents a number of caveats:

o The method consumes an extra OCP. It also requires a means to signal the capabilities of the leaf, e.g., using "RPL Mode of Operation extension" [I-D.rahul-roll-mop-ext].

o If an implementation does not move to a leaf mode when the OCP is changed to an unknown one, then the node may be stalled.

o If the only possible parents of a node are nodes that do not support RFC 8138, then that node will loose all its parent at the time of the migration and it will be stalled until a parent is deployed with the new capability.

o Nodes that only support RFC8138 for forwarding may not parse the RPI in native form. If such nodes are present, the parent needs to encapsulate with RFC8138.

5.3. Double Instance Scenario

An alternate to the Single Instance Scenario is to deploy an additional Instance for the nodes that support [RFC8138]. The two instances operate as ships-in-the-night as specified in [RFC6550]. The preexisting Instance that does not use [RFC8138], whereas the new Instance does. This is signaled by the "T" flag which is only set in the configuration option in DIO messages in the new Instance.
The legacy nodes would not be configured to participate to the second instance, and islands that are only connected via legacy nodes would not be reachable over the second instance.

Nodes that support RFC 8138 participate to both Instances but favor the new Instance for the traffic that they source. On the other hand, nodes that only support the uncompressed format would either not be configured for the new instance, or would be configured to join it as leaves only.

This method eliminates the risks of nodes being stalled that are described in Section 5.2 but requires implementations to support at least two RPL Instances and demands management capabilities to introduce new Instances and deprecate old ones.

5.4. Rolling Back

After downgrading a network to turn the [RFC8138] compression off, the administrator SHOULD make sure that all nodes have converged to the "T" flag reset before allowing nodes that do not support the compression in the network (see caveats in Section 5.2). This also requires a means to signal the current state of the setting of the logic that controls the compression in the node, also using [I-D.rahul-roll-mop-ext].

6. IANA Considerations

This specification updates the "Registry for the DODAG Configuration Option Flags" that was created for [RFC6550] as follows:

<table>
<thead>
<tr>
<th>Bit Number</th>
<th>Meaning</th>
<th>Defining Spec</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 (suggested)</td>
<td>Turn on RFC8138 Compression (T)</td>
<td>This</td>
</tr>
</tbody>
</table>

Table 1: New DODAG Configuration Option Flag

7. Security Considerations

No specific threat was identified with this specification.

8. Acknowledgments
9. References

9.1. Normative References

[I-D.ietf-roll-useofrplinfo]

[I-D.rahul-roll-mop-ext]


9.2. Informative References


Authors' Addresses