Workgroup: ROLL
Updates: 6550 (if approved)
Published: 29 September 2020
Intended Status: Standards Track
Expires: 2 April 2021
Authors: P. Thubert, Ed. R.A. Jadhav M. Gillmore
Cisco Systems Huawei Tech Itron

Root initiated routing state in RPL

Abstract

This document updates RFC 6550 to enable a RPL Root to install and maintain Projected 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 or resilient than those obtained with the classical 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.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

RPL, the "Routing Protocol for Low Power and Lossy Networks" [RPL]
(LLNs), is a generic Distance Vector protocol that is well suited

for application in a variety of 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. 6TiSCH uses RPL for its routing operations.

The "6TiSCH Architecture" [6TiSCH-ARCHI] also leverages the "Deterministic Networking Architecture" [RFC8655] centralized 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. With DetNet and 6TiSCH, the component of the controller that is responsible of computing routes is called a Path Computation Element ([PCE]).

Based on heuristics of usage, path length, and knowledge of device capacity and available resources such as battery levels and reservable buffers, the PCE with a global visibility on the system can compute direct Peer to Peer (P2P) routes that are optimized for the needs expressed by an objective function. This document specifies protocol extensions to RPL [RPL] that enable the Root of a main DODAG to install centrally-computed routes inside the DODAG on behalf of a PCE.

This specification expects that the main RPL Instance is operated in RPL Non-Storing Mode of Operation (MOP) to sustain the exchanges with the Root. In that Mode, the Root has enough information to build a basic DODAG topology based on parents and children, but lacks the knowledge of siblings. This document adds the capability for nodes to advertise sibling information in order to improve the topological awareness of the Root.

As opposed to the classical RPL operations where routes are injected by the Target nodes, the protocol extensions enable the Root of a DODAG to project the routes that are needed onto the nodes where they should be installed. This specification uses the term Projected Route to refer to those routes. Projected Routes can be used to reduce the size of the source routing headers with loose source routing operations down the main RPL DODAG. Projected Routes can also be used to build transversal routes for route optimization and Traffic Engineering purposes, between nodes of the DODAG.

A Projected Route may be installed in either Storing and Non-Storing Mode, potentially resulting in hybrid situations where the Mode of the Projected Route is different from that of the main RPL Instance. A Projected Route may be a stand-alone end-to-end path or a Segment in a more complex forwarding graph called a Track.

The concept of a Track was introduced in the 6TiSCH architecture, as a potentially complex path with redundant forwarding solutions along the way. A node at the ingress of more than one Segment in a Track may use any combination of those Segments to forward a packet towards the Track Egress.

The "Reliable and Available Wireless (RAW) Architecture/Framework" [RAW-ARCHI] defines the Path Selection Engine (PSE) that adapts the use of the path redundancy within a Track to defeat the diverse causes of packet loss.

The PSE is a dataplane extension of the PCE; it controls the forwarding operation of the packets within a Track, using Packet ARQ, Replication, Elimination, and Overhearing (PAREO) functions over the Track segments, to provide a dynamic balance between the reliability and availability requirements of the flows and the need to conserve energy and spectrum.

The time scale at which the PCE (re)computes the Track can be long, using long-term statistical metrics to perform global optimizations at the scale of the whole network. Conversely, the PSE makes forwarding decisions at the time scale of one or a small collection of packets, based on a knowledge that is limited in scope to the Track itself, so it can be refreshed at a fast pace.

Projected Routes must be used with the parsimony to limit the amount of state that is installed in each device to fit within the device resources, and to maintain the amount of rerouted traffic within the capabilities of the transmission links. The methods used to learn the node capabilities and the resources that are available in the devices and in the network are out of scope for this document.

This specification uses the RPL Root as a proxy to the PCE. The PCE may be collocated with the Root, or may reside in an external Controller.

In that case, the PCE exchanges control messages with the Root over a Southbound API that is out of scope for this specification. The algorithm to compute the paths and the protocol used by an external PCE to obtain the topology of the network from the Root are also out of scope.

2. Terminology

2.1. Requirements Language

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. Glossary

This document often uses the following acronyms:

CMO: Control Message Option
DAO: Destination Advertisement Object
DAG: Directed Acyclic Graph
DODAG: Destination-Oriented Directed Acyclic Graph; A DAG with only one vertex (i.e., node) that has no outgoing edge (i.e., link)
LLN: Low-Power and Lossy Network
MOP: RPL Mode of Operation
P-DAO: Projected DAO
PDR: P-DAO Request
RAN: RPL-Aware Node (either a RPL Router or a RPL-Aware Leaf)
RAL: RPL-Aware Leaf
RPI: RPL Packet Information
RPL: IPv6 Routing Protocol for LLNs [RPL]
RPO: A Route Projection Option; it can be a VIO or an SRVIO.
RTO: RPL Target Option
RUL: RPL-Unaware Leaf
SIO: RPL Sibling Information Option
SRVIO: A Source-Routed Via Information Option, used in Non-Storing Mode P-DAO messages.
SubDAG: A DODAG rooted at a node which is a child of that node and a subset of a larger DAG
TIO: RPL Transit Information Option
VIO: A Via Information Option, used in Storing Mode P-DAO messages.

2.3. Other Terms

Projected Route: A Projected Route is a path segment that is computed remotely, and installed and maintained by a RPL Root.
Projected DAO: A DAO message used to install a Projected Route.
Track: A complex path with redundant Segments to a destination.
TrackID: A RPL Local InstanceID with the 'D' bit set. The TrackID is associated with a IPv6 Address to the Track Egress Node.

2.4. References

In this document, readers will encounter terms and concepts that are discussed in the "Routing Protocol for Low Power and Lossy Networks" [RPL] and "Terminology in Low power And Lossy Networks" [RFC7102].

3. Updating RFC 6550

Section 6 of [RPL] introduces the RPL Control Message Options (CMO), including the RPL Target Option (RTO) and Transit Information Option

(TIO), which can be placed in RPL messages such as the Destination Advertisement Object (DAO). This specification extends the DAO message with the Projected DAO (P-DAO); a P-DAO message signals a Projected Route using new CMOs presented therein.

A Projected Route can be an additional route of higher precedence within the main DODAG, in which case it is installed with the RPLInstanceID and DODAGID of the main DODAG.

A Projected Route can also be a Segment within a Track. A stand-alone Segment can be used as a Serial (end-to-end) Track. Segments can also be combined to form a Complex Track. The Root uses a local RPL Instance rooted at the Track Egress to establish and maintain the Track. The local RPLInstanceID of the Track is called the TrackID, more in Section 4.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   TrackID     |K|D|   Flags   |   Reserved    | DAOSequence   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+            IPv6 Address of the Track Egress                   +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Option(s)...
+-+-+-+-+-+-+-+-+
```

Figure 1: Projected DAO Format for a Track

A P-DAO message signals the IPv6 Address of the Track Egress in the DODAGID field of the DAO Base Object, and the TrackID in the RPLInstanceID field, as shown in Figure 1.

In RPL Non-Storing Mode, the TIO and RTO are combined in a DAO message to inform the DODAG Root of all the edges in the DODAG, which are formed by the directed parent-child relationships. Options may be factorized; multiple RTOs may be present to signal a collection of children that can be reached via the parent(s) indicated in the TIO(s) that follows the RTOs.

This specification generalizes the case of a parent that can be used to reach a child with that of a whole Track through which both children and siblings may be reached.

New CMOs called the Route Projection Options (RPO) are introduced for use in P-DAO messages as a multihop alternative to the TIO. One RPO is the Via Information Option (VIO); the VIO installs a state at each hop along a Storing Mode Projected Route. The other is the Source-Routed VIO (SRVIO); the SRVIO installs a source-routing state at the Segment ingress, which uses that state to encapsulate a packet with a Source-Routing Header along a Non-Storing Mode Projected Route.

Like in a DAO message, the RTOs can be factorized in a P-DAO, but the CMOs cannot. A P-DAO contains one or more RTOs that indicate the destinations that can be reached via the Track, and either one SRVIO or one VIO signal the sequence of hops between the Track Ingress and the (penultimate) node before the Track Egress. In Non-Storing Mode, the Root sends the P-DAO to the Track Ingress where the source-routing state is stored. In Storing Mode, the P-DAO is sent to the Track Egress and forwarded along the Segment in the reverse direction, installing a Storing Mode state at each hop.

This specification adds another CMO called the Sibling Information Option (SIO) that is used by a RPL Aware Node (RAN) to advertise a selection of its candidate neighbors as siblings to the Root, more in Section 5.4. The sibling selection process is out of scope.

Two new RPL Control Messages are also introduced, to enable a RAN to request the establishment of a Track between self as the Track Ingress Node and a Track Egress. The RAN makes its request by sending a new P-DAO Request (PDR) Message to the Root. The Root confirms with a new PDR-ACK message back to the requester RAN, see Section 5.1 for more. A positive PDR-ACK indicates that the Track was built and that the Roots commits to maintain the Track for a negotiated lifetime.

In the case of a complex Track, each Segment is maintained independently and asynchronously by the Root, with its own lifetime that may be shorter, the same, or longer than that of the Track. The Root may use an asynchronous PDR-ACK with an negative status to indicate that the Track was terminated before its time.

4. Identifying a Track

RPL defines the concept of an Instance to signal an individual routing topology 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 within one routing topology.

This draft conforms the RPL Instance model as follows:

* The PCE MAY use P-DAO messages to add better routes in the main (Global) Instance in conformance with the routing objectives in that Instance. To achieve this, the PCE MAY install a Storing Mode Projected Route along a path down the main (Non-Storing Mode) DODAG. This enables a loose source routing and reduces the size of the Source Routing Header, see Appendix A.1.

  When adding a Storing Mode Projected Route to the main RPL Instance, the Root MUST set the RPLInstanceID field of the DAO message (see section 6.4.1. of [RPL]) to the RPLInstanceID of the main DODAG, and set the DODAGID field to the Segment Egress. The Projected Route provides a longer match to the Egress than the default route via the Root, so it is preferred. Once the Projected Route is installed, the intermediate nodes listed in the VIO between the first (excluded) and the last (included) can be elided in a Source-Route Header that signals that Segment.

* The Root MAY also use P-DAO messages to install a specific (say, Traffic Engineered) path as a Serial of a Complex Track, to a particular endpoint that is the Track Egress. In that case, the Root MUST use a Local RPL Instance (see section 5 of [RPL]) as TrackID.

  The TrackID MUST be unique for the Global Unique IPv6 Address (GUA) or Unique-Local Address (ULA) of the Track Egress that serves as DODAGID for the Track. This way, a Track is uniquely identified by the tuple (Track Egress Address, TrackID) where the TrackID is always represented with the 'D' flag set. The Track Egress Address and the TrackID are signaled in the P-DAO message as shown in Figure 1.

* In the data packets, the Track Egress Address and the TrackID are respectively signaled in IPv6 Address of the final destination and the RPLInstanceID field of the RPL Packet Information (RPI) (see [USEofRPLinfo]) in the outer chain of IPv6 Headers.

  If the outer chain of IPv6 Headers contains a Source-Routing Header that is not fully consumed, then the final destination is the last entry in the Source-Routing Header; else it is the Destination Address in the IPv6 Header. When using the [RFC8138] compression, it is the last hop of the last SRH-6LoRH of the outer header in either case.

  The 'D' flag in the RPLInstanceID MUST be set to indicate that the final destination address in the IPv6 header owns the local RPLInstanceID, more in Section 6.2.

*A packet that is being routed over the RPL Instance associated to a first Non-Storing Mode Track MAY be placed (encapsulated) in a second Track to cover one loose hop of the first Track. On the other hand, a Storing Mode Track must be strict and a packet that it placed in a Storing Mode Track MUST follow that Track till the Track Egress.

When a Track Egress extracts a packet from a Track (decapsulates the packet), the Destination of the inner packet MUST be either this node or a direct neighbor, otherwise the packet MUST be dropped. That Destination may be the next Hop in a Non-Storing Mode Track.

All properties of a Track operations are inherited form the main RPL Instance that is used to install the Track. For instance, the use of compression per [RFC8138] is determined by whether it is used in the main instance, e.g., by setting the "T" flag [TURN-ON_RFC8138] in the RPL configuration option.

5. New RPL Control Messages and Options

5.1. New P-DAO Request Control Message

The P-DAO Request (PDR) message is sent to the Root to request a new that the PCE establishes a new a projected route from self to the Track Egress indicated in the TIO as a full path of a collection of Segments in a Track. Exactly one TIO MUST be present, more in Section 6.1.

The RPL Control Code for the PDR is 0x09, to be confirmed by IANA. The format of PDR Base Object is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   TrackID     |K|R|   Flags   |  ReqLifetime  | PDRSequence   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Option(s)...
+-+-+-+-+-+-+-+-+
```

Figure 2: New P-DAO Request Format

TrackID: 8-bit field indicating the RPLInstanceID associated with the Track. It is set to zero upon the first request for a new Track and then to the TrackID once the Track was created, to either renew it of destroy it.

K: The 'K' flag is set to indicate that the recipient is expected to send a PDR-ACK back.

R:
 The 'R' flag is set to indicate that the Requested path should be redundant.

Flags: Reserved. The Flags field MUST initialized to zero by the sender and MUST be ignored by the receiver

ReqLifetime: 8-bit unsigned integer.

The requested lifetime for the Track expressed in Lifetime Units (obtained from the DODAG Configuration option).

A PDR with a fresher PDRSequence refreshes the lifetime, and a PDRLifetime of 0 indicates that the track should be destroyed.

PDRSequence: 8-bit wrapping sequence number, obeying the operation in section 7.2 of [RPL].

The PDRSequence is used to correlate a PDR-ACK message with the PDR message that triggered it. It is incremented at each PDR message and echoed in the PDR-ACK by the Root.

5.2. New PDR-ACK Control Message

The new PDR-ACK is sent as a response to a PDR message with the 'K' flag set. The RPL Control Code for the PDR-ACK is 0x0A, to be confirmed by IANA. Its format is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    TrackID    |     Flags     | Track Lifetime|  PDRSequence  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| PDR-ACK Status|                Reserved                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option(s)...
+-+-+-+-+-+-+-+-+
```

Figure 3: New PDR-ACK Control Message Format

TrackID:
The RPLInstanceID of the Track that was created. The value of 0x00 is used to when no Track was created.

Flags: Reserved. The Flags field MUST initialized to zero by the sender and MUST be ignored by the receiver

Track Lifetime: Indicates that remaining Lifetime for the Track, expressed in Lifetime Units; a value of zero (0x00) indicates that the Track was destroyed or not created.

PDRSequence: 8-bit wrapping sequence number. It is incremented at each PDR message and echoed in the PDR-ACK.

PDR-ACK Status: 8-bit field indicating the completion.

The PDR-ACK Status is substructured as indicated in Figure 4:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|E|R|  Value    |
+-+-+-+-+-+-+-+-+
```

Figure 4: PDR-ACK status Format

E: 1-bit flag. Set to indicate a rejection. When not set, a value of 0 indicates Success/Unqualified acceptance and other values indicate "not an outright rejection".

R: 1-bit flag. Reserved, MUST be set to 0 by the sender and ignored by the receiver.

Status Value: 6-bit unsigned integer. Values depending on the setting of the 'E' flag as indicated respectively in Table 4 and Table 5.

Reserved: The Reserved field MUST initialized to zero by the sender and MUST be ignored by the receiver

5.3. Route Projection Options

The RPOs indicate a series of IPv6 addresses that can be compressed using the method defined in the "6LoWPAN Routing Header" [RFC8138] specification using the address of the Root found in the DODAGID field of DIO messages as Compression Reference.

An RPO indicates a Projected Route that can be a Serial Track in full or a Segment of a more Complex Track. In Non-Storing Mode, multiple RPO may be placed after a TIO to reflect different Segments originated at this node. The Track is identified by a TrackID that is a Local RPLInstanceID to the Track Egress of the Track.

The format of RPOs is as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Type        | Option Length |     Flags     |   SegmentID   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |Segm. Sequence | Seg. Lifetime |      SRH-6LoRH header         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   .                                                               .
   .                      Via Address 1                            .
   .                                                               .
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   .                               ....                            .
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   .                                                               .
   .                      Via Address n                            .
   .                                                               .
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: Route Projection Option format (uncompressed form)

Option Type:
0x0B for VIO, 0x0C for SRVIO (to be confirmed by IANA)

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

Flags: Reserved. The Flags field MUST initialized to zero by the sender and MUST be ignored by the receiver

SegmentID: 8-bit field that identifies a Segment within a Track or the main DODAG as indicated by the TrackID field. A Value of 0 is used to signal a Serial Track, i.e., made of a single segment.

Segment Sequence: 8-bit unsigned integer. The Segment Sequence obeys the operation in section 7.2 of [RPL] and the lollipop starts at 255. When the Root of the DODAG needs to refresh or update a Segment in a Track, it increments the Segment Sequence individually for that Segment. The Segment information indicated in the RTO deprecates any state for the Segment indicated by the SegmentID within the indicated Track and sets up the new information. A RTO with a Segment Sequence that is not as fresh as the current one is ignored. a RTO for a given Track Egress with the same (TrackID, SegmentID, Segment Sequence) indicates a retry; it MUST NOT change the Segment and MUST be propagated or answered as the first copy.

Segment Lifetime: 8-bit unsigned integer. The length of time in Lifetime Units (obtained from the Configuration option) that the Segment is usable. The period starts when a new Segment 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 Segment Lifetime of zero for a Track Egress is referred as a No-Path (for that Track Egress) in this document.

SRH-6LoRH header: The first 2 bytes of the SRH-6LoRH as shown in Figure 6 of [RFC8138]. A 6LoRH Type of 4 means that the VIA Addresses are provided in full with no compression.

Via Address: A Luistof Via Addresses along one Segment, indicated in the order of the path from the ingress to the egress nodes.

In a VIO, the list is a strict path between direct neighbors, whereas for an SRVIO, the list may be loose, provided that each listed node has a path to the next listed node, e.g., via another Track.

In the case of a VIO, or if [RFC8138] is turned off, then the Root MUST use only one SRH-6LoRH per RPO, and the compression is the same for all the addresses, as shown in Figure 5.

If [RFC8138] is turned on, then the Root SHOULD optimize the size of the SRVIO; in that case, more than one SRH-6LoRH may be needed if the compression of the addresses changes inside the Segment and different SRH-6LoRH Types are used.

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.

5.4. Sibling Information Option

The Sibling Information Option (SIO) provides indication on siblings that could be used by the Root to form Projected Routes. The format of SIOs is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type        | Option Length |Comp.|B|D|Flags|    Opaque     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Step of Rank      |          Reserved             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
.                                                               .
.         Sibling DODAGID (if 'D' flag not set)                 .
.                                                               .
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
.                                                               .
.                       Sibling Address                         .
.                                                               .
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: Sibling Information Option Format

Option Type: 0x0D (to be confirmed by IANA)

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

Compression Type: 3-bit unsigned integer. This is the SRH-6LoRH Type as defined in figure 7 in section 5.1 of [RFC8138] that corresponds to the compression used for the Sibling Address.

Reserved for Flags: MUST be set to zero by the sender and MUST be ignored by the receiver.

B: 1-bit flag that is set to indicate that the connectivity to the sibling is bidirectional and roughly symmetrical. In that case, only one of the siblings may report the SIO for the hop. If 'B' is not set then the SIO only indicates connectivity from the sibling to this node, and does not provide information on the hop from this node to the sibling.

D: 1-bit flag that is set to indicate that sibling belongs to the same DODAG. When not set, the Sibling DODAGID is indicated.

Flags: Reserved. The Flags field MUST initialized to zero by the sender and MUST be ignored by the receiver

Opaque: MAY be used to carry information that the node and the Root understand, e.g., a particular representation of the Link properties such as a proprietary Link Quality Information for packets received from the sibling. An industraial Alliance that uses RPL for a particular use / environment MAY redefine the use of this field to fit its needs.

Step of Rank: 16-bit unsigned integer. This is the Step of Rank [RPL] as computed by the Objective Function between this node and the sibling.

Reserved: The Reserved field MUST initialized to zero by the sender and MUST be ignored by the receiver

Sibling DODAGID: 2 to 16 bytes, the DODAGID of the sibling in a [RFC8138] compressed form as indicated by the Compression Type field. This field is present when the 'D' flag is not set.

Sibling Address: 2 to 16 bytes, the IPv6 Address of the sibling in a [RFC8138] compressed form as indicated by the Compression Type field.

An SIO MAY be immediately followed by a DAG Metric Container. In that case the DAG Metric Container provides additional metrics for the hop from the Sibling to this node.

6. Projected DAO

This draft adds a capability to RPL whereby the Root of a DODAG projects a Track by sending one or more extended DAO message called Projected-DAO (P-DAO) messages to chosen routers in the DODAG, indicating one or more sequence(s) of routers inside the DODAG via which the Target(s) indicated in the RPL Target Option(s) (RTO) 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 exactly one RTO and either one VIO or one or more SRVIOs following it. There can be at most one such sequence of RTOs and then RPOs.

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

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

* The Non-Storing Mode is discussed in Section 6.3. It uses an SRVIO that carries a list of Via Addresses to be used as a source-routed Segment to the Track Egress. The recipient of the P-DAO is the ingress router of the source-routed Segment. Upon a Non-Storing Mode P-DAO, the ingress router installs a source-routed state to the Track Egress and replies to the Root directly with a DAO-ACK message.

* The Storing Mode is discussed in Section 6.4. It uses a single VIO, within which are signaled 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 Track Egress 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 Projected Route, an ICMP error is sent to the Root with a new Code "Error in Projected Route" (See Section 8.9). The Root can then modify or remove the Projected 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 outter 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 6.3 and Section 6.4 respectively.

6.1. Requesting a Track

A Node is free to ask the Root for a new Track with a PDR message, for a duration indicated in a Requested Lifetime field. Upon that Request, the Root install the necessary Segments and answers with a PDR-ACK indicated the granted Track Lifetime. When the Track Lifetime returned in the PDR-ACK is close to elapse, the resquesting Node needs to resend a PDR using the TrackID in the PDR-ACK to get the lifetime of the Track prolonged, else the Track will time out and the Root will tear down the whole structure.

The Segment Lifetime in the P-DAO messages does not need to be aligned to the Requested Lifetime in the PDR, or between P-DAO messages for different Segments. The Root may use shorter lifetimes for the Segments and renew them faster than the Track is, or longer lifetimes in which case it will need to tear down the Segments if the Track is not renewed.

The Root is free to install which ever Segments it wants, and change them overtime, to serve the Track as needed, without notifying the resquesting Node. If the Track fails and cannot be reestablished, the Root notifies the resquesting Node asynchronously with a PDR-ACK with a Track Lifetime of 0, indicating that the Track has failed, and a PDR-ACK Status indicating the reason of the fault.

All the Segments MUST be of a same mode, either Storing or Non-Storing. All the Segments MUST be created with the same TrackID and Track Egress in the P-DAO.

6.2. Routing over a Track

Sending a packet over a Track implies the addition of a RPI to indicate the Track, in association with the IPv6 destination. In case of a Non-Storing Mode Projected Route, a Source Routing Header is needed as well.

The Destination IPv6 Address of a packet that is placed in a Track MUST be that of the Track Egress of Track. The outer header of the packet MUST contain an RPI that indicates the TrackID as RPL Instance ID.

If the Track Ingress is the originator of the packet and the Track Egress is the destination of the packet, there is no need for an encapsulation. Else, i.e., if the Track Ingress is forwarding a packet into the Track, or if the the final destination is reached via is not the Track Egress, but reached over the Track via the Track Egress, then an IP-in-IP encapsulation is needed.

6.3. Non-Storing Mode Projected Route

As illustrated in Figure 7, a P-DAO that carries an SRVIO enables the Root to install a source-routed path towards a Track Egress in any particular router; with this path information the router can add a source routed header reflecting the Projected Route to any packet for which the current destination either is the said Track Egress or can be reached via the Track Egress.

```
        ------+---------
              |          Internet
              |
           +-----+
           |     | Border Router
           |     |  (RPL Root)
           +-----+                  |  P  ^            |
              |                     | DAO | ACK        | Loose
        o    o   o    o     router  V     |            | Source
    o o   o  o   o  o  o      o  o           | P-DAO   . Route
   o  o o  o o    o   o   o  o  o            | Source  . Path
   o   o    o  o     o  o    o  o  o         | Route   . From
  o  o   o  o   o         o   o o            | Path    . Root
     o  o  o  o             o    Track Egress V        . To
    o       o               o    o                     | Desti-
  o          o             o     o                     | nation
                                destination            V

                   LLN
```

Figure 7: 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 Projected Route does not have a connected route (a direct adjacency) to the next soure routed hop and fails to locate it as a neighbor or a neighbor of a neighbor, then it MUST ensure that it has another Projected 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 Track Egress, the router inserts the source routing header in the packet with the destination set to the Track Egress. 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 Track Egress.

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 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 [RFC8138]. 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 SRVIO and then the Track Egress in the SRH 6LoRH Header.

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 [RPL]. 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.

6.4. Storing Mode Projected Route

As illustrated in Figure 8, the Storing Mode route projection is used by the Root to install a routing state 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 Egress node, for instance a loose source routed packet for which the next loose hop is the Egress node, or a packet for which the router has a routing state to the final destination via the Egress node.

```
      ------+---------
            |          Internet
            |
         +-----+
         |     | Border Router
         |     |  (RPL Root)
         +-----+                      |     ^                    |
            |                         | DAO | ACK                |
      o    o   o    o                 |     |                    |
  o o   o  o   o  o  o o   o          |  ^      | Projected      .
 o  o o  o o    o   o   o  o  o       |  | DAO  | Route          .
 o   o    o  o     o  o    o  o  o    | ^       |                .
o  o   o  o   o         o   o o       v | DAO    v                .
o          o   LLN   o   o     o                                 |
    o o   o        o     o              Loose Source Route Path |
 o       o      o    o                 From Root To Destination v
```

Figure 8: 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 Segment 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 last Via Address in the VIO. This is how the egress recognizes its role. In a similar fashion, the ingress node recognizes its role as it matches first Via Address in the 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 Projected 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 its address in the VIO, and uses as next hop the address found in the previous Via Address field in the VIO. The router MAY install additional routes towards the VIA Addresses that are the VIO after the next one, if any, but in case of a conflict or a lack of resource, the route(s) to the Target(s) have 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 Segment 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 Projected 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 Projected Route that is broken.

7. Security Considerations

This draft uses messages that are already present in RPL [RPL] 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?

8. IANA Considerations

8.1. New RPL Control Codes

This document extends the IANA Subregistry created by RFC 6550 for RPL Control Codes as indicated in Table 1:

Code	Description	Reference
0x09	Projected DAO Request (PDR)	This document
0x0A	PDR-ACK	This document

Table 1: New RPL Control Codes

8.2. New RPL Control Message Options

This document extends the IANA Subregistry created by RFC 6550 for RPL Control Message Options as indicated in Table 2:

Value	Meaning	Reference
0x0B	Via Information option	This document
0x0C	Source-Routed Via Information option	This document
0x0D	Sibling Information option	This document

Table 2: RPL Control Message Options

8.3. SubRegistry for the Projected DAO Request Flags

IANA is required to create a registry for the 8-bit Projected DAO Request (PDR) Flags field. Each bit is tracked with the following qualities:

* Bit number (counting from bit 0 as the most significant bit)

* Capability description

* Reference

Registration procedure is "Standards Action" [RFC8126]. The initial allocation is as indicated in Table 3:

Bit number	Capability description	Reference
0	PDR-ACK request (K)	This document
1	Requested path should be redundant (R)	This document

Table 3: Initial PDR Flags

8.4. SubRegistry for the PDR-ACK Flags

IANA is required to create an subregistry for the 8-bit PDR-ACK Flags field. Each bit is tracked with the following qualities:

* Bit number (counting from bit 0 as the most significant bit)

* Capability description

* Reference

Registration procedure is "Standards Action" [RFC8126]. No bit is currently defined for the PDR-ACK Flags.

8.5. Subregistry for the PDR-ACK Acceptance Status Values

IANA is requested to create a Subregistry for the PDR-ACK Acceptance Status values.

* Possible values are 6-bit unsigned integers (0..63).

* Registration procedure is "Standards Action" [RFC8126].

* Initial allocation is as indicated in Table 4:

Value	Meaning	Reference
0	Unqualified acceptance	This document

Table 4: Acceptance values of the PDR-ACK Status

8.6. Subregistry for the PDR-ACK Rejection Status Values

IANA is requested to create a Subregistry for the PDR-ACK Rejection Status values.

* Possible values are 6-bit unsigned integers (0..63).

* Registration procedure is "Standards Action" [RFC8126].

* Initial allocation is as indicated in Table 5:

Value	Meaning	Reference
0	Unqualified rejection	This document

Table 5: Rejection values of the PDR-ACK Status

8.7. SubRegistry for the Route Projection Options Flags

IANA is requested to create a Subregistry for the 5-bit Route Projection Options (RPO) Flags field. Each bit is tracked with the following qualities:

*Bit number (counting from bit 0 as the most significant bit)

*Capability description

*Reference

Registration procedure is "Standards Action" [RFC8126]. No bit is currently defined for the Route Projection Options (RPO) Flags.

8.8. SubRegistry for the Sibling Information Option Flags

IANA is required to create a registry for the 5-bit Sibling Information Option (SIO) Flags field. Each bit is tracked with the following qualities:

*Bit number (counting from bit 0 as the most significant bit)

*Capability description

*Reference

Registration procedure is "Standards Action" [RFC8126]. The initial allocation is as indicated in Table 6:

Bit number	Capability description	Reference
0	Connectivity is bidirectional (B)	This document

Table 6: Initial SIO Flags

8.9. Error in Projected Route ICMPv6 Code

In some cases RPL will return an ICMPv6 error message when a message cannot be forwarded along a Projected 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.

9. Acknowledgments

The authors wish to acknowledge JP Vasseur, Remy Liubing, James Pylakutty and Patrick Wetterwald for their contributions to the ideas developed here.

10. Normative References

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

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RPL] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <https://www.rfc-editor.org/info/rfc6550>.

[RFC6554] Hui, J., Vasseur, JP., Culler, D., and V. Manral, "An IPv6 Routing Header for Source Routes with the Routing Protocol for Low-Power and Lossy Networks (RPL)", RFC 6554, DOI 10.17487/RFC6554, March 2012, <https://www.rfc-editor.org/info/rfc6554>.

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

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

11. Informative References

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

[RFC6997] Goyal, M., Ed., Baccelli, E., Philipp, M., Brandt, A., and J. Martocci, "Reactive Discovery of Point-to-Point Routes in Low-Power and Lossy Networks", RFC 6997, DOI

10.17487/RFC6997, August 2013, <https://www.rfc-editor.org/info/rfc6997>.

[6TiSCH-ARCHI] Thubert, P., "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4", Work in Progress, Internet-Draft, draft-ietf-6tisch-architecture-29, 27 August 2020, <https://tools.ietf.org/html/draft-ietf-6tisch-architecture-29>.

[RAW-ARCHI] Thubert, P., Papadopoulos, G., and R. Buddenberg, "Reliable and Available Wireless Architecture/Framework", Work in Progress, Internet-Draft, draft-pthubert-raw-architecture-04, 6 July 2020, <https://tools.ietf.org/html/draft-pthubert-raw-architecture-04>.

[TURN-ON_RFC8138]
Thubert, P. and L. Zhao, "Configuration option for RFC 8138", Work in Progress, Internet-Draft, draft-thubert-roll-turnon-rfc8138-03, 8 July 2019, <https://tools.ietf.org/html/draft-thubert-roll-turnon-rfc8138-03>.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <https://www.rfc-editor.org/info/rfc8655>.

[RFC8025] Thubert, P., Ed. and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch", RFC 8025, DOI 10.17487/RFC8025, November 2016, <https://www.rfc-editor.org/info/rfc8025>.

[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, April 2017, <https://www.rfc-editor.org/info/rfc8138>.

[USEofRPLinfo] Robles, I., Richardson, M., and P. Thubert, "Using RPI Option Type, Routing Header for Source Routes and IPv6-in-IPv6 encapsulation in the RPL Data Plane", Work in Progress, Internet-Draft, draft-ietf-roll-useofrplinfo-40, 25 June 2020, <https://tools.ietf.org/html/draft-ietf-roll-useofrplinfo-40>.

[PCE] IETF, "Path Computation Element", <https://datatracker.ietf.org/doc/charter-ietf-pce/>.

Appendix A. Applications

A.1. Loose Source Routing

A RPL implementation operating in a very constrained LLN typically uses the Non-Storing Mode of Operation as represented in Figure 9. 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.

```
           ------+---------
                 |          Internet
                 |
              +-----+
              |     | Border Router
              |     |  (RPL Root)
              +-----+                      ^     |        |
                 |                         | DAO | ACK    |
           o    o   o    o                 |     |        | Strict
       o o   o  o   o  o  o o   o          |     |        | Source
      o  o o  o o    o   o   o  o  o       |     |        | Route
      o   o    o  o     o  o    o  o  o    |     |        |
     o  o   o  o   o         o   o o       |     v        v
     o          o             o     o
                       LLN
```

Figure 9: 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

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 10:

* 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.

* 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.

```
                ------+---------
                 |          Internet
                 |
              +-----+
              |     | Border Router
              |     |  (RPL Root)
              +-----+
                 X
           ^    v   o    o
       ^ o   o  v   o  o  o o   o
      ^  o o  o v    o   o   o  o  o
      ^   o    o  v     o  o    o  o  o
     S  o   o  o   D         o   o o
     o          o             o     o
                       LLN
```

Figure 10: 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, and that it is even more critical in Non-Storing Mode than it is in Storing Mode, because the routing stretch is wider. 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
                 |
              +-----+
              |     | Border Router
              |     |  (RPL Root)
              +-----+
                 |
           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
     S>>A>>>B>>C>>>D         o   o o
     o          o             o     o
                       LLN
```

Figure 11: 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 oculd 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 12, 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   o     o 35
     /      /      |    \    |    \
    o 41   o 42    o     o   o 45   o 46
    |      |       |     |    \     |
    o 51   o 52    o 53  o     o 55 o 56

                       LLN
```

Figure 12: 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 I and node D via routers A, B and E, in order to serve the needs of a particular application.

The Root sends a P-DAO to node E, with an RTO indicating the destination D, a TIO optionally indicating the Track Egress in the Parent Address field, and a sequence of Via Information options indicating the hops, one for S, which is the ingress router of the Segment, one for A, and then one for B, which are respectively the intermediate and penultimate routers.

```
              ------+---------
                    |          Internet
                    |
                 +-----+
                 |     | Border Router
                 |     |  (RPL Root)
                 +-----+
                    | P-DAO message to C
              o    |   o    o
          o o   o |    o  o  o o   o
         o  o o  |o    o   o   o  o  o
         o   o   V  o     o  o    o  o  o
        S  A  B  E   D         o   o o
        o          o             o     o
                          LLN
```

Figure 13: 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.

```
           ------+---------
                 |          Internet
                 |
              +-----+
              |     | Border Router
              |     |  (RPL Root)
              +-----+
               ^ P-DAO-ACK from S
           /    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
     S  A  B  C   D         o   o o
     o          o             o     o
                       LLN
```

Figure 14: P-DAO-ACK to Root

As a result, a transversal route is installed that does not need to follow the DODAG structure.

```
           ------+---------
                 |          Internet
                 |
              +-----+
              |     | Border Router
              |     |  (RPL Root)
              +-----+
                 |
           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
     S>>A>>>B>>C>>>D         o   o o
     o          o             o     o
                       LLN
```

Figure 15: Projected Transversal Route

Authors' Addresses

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

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

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

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