Internet Engineering Task Force (IETF)
Request for Comments: 6552
Category: Standards Track
ISSN: 2070-1721
P. Thubert, Ed.
Cisco Systems
March 2012

Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)

Abstract

The Routing Protocol for Low-Power and Lossy Networks (RPL) specification defines a generic Distance Vector protocol that is adapted to a variety of network types by the application of specific Objective Functions (OFs). An OF states the outcome of the process used by a RPL node to select and optimize routes within a RPL Instance based on the Information Objects available; an OF is not an algorithm.

This document specifies a basic Objective Function that relies only on the objects that are defined in the RPL and does not use any protocol extensions.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Routing Protocol for Low-Power and Lossy Networks (RPL) specification [RFC6550] defines a generic Distance Vector protocol that is adapted to a variety of Low-Power and Lossy Network (LLN) types by the application of specific Objective Functions (OFs).

A RPL OF states the outcome of the process used by a RPL node to select and optimize routes within a RPL Instance based on the Information Objects available. As a general concept, an OF is not an algorithm. For example, outside RPL, "shortest path first" is an OF where the least cost path between two points is derived as an outcome; there are a number of algorithms that can be used to satisfy the OF, of which the well-known Dijkstra algorithm is an example.

The separation of OFs from the core protocol specification allows RPL to be adapted to meet the different optimization criteria required by the wide range of deployments, applications, and network designs.

RPL forms Directed Acyclic Graphs (DAGs) as collections of Destination-Oriented DAGs (DODAGs) within instances of the protocol. Each instance is associated with a specialized Objective Function. A DODAG is periodically reconstructed as a new DODAG Version to enable a global reoptimization of the graph.

An instance of RPL running on a device uses an Objective Function to help it determine which DODAG and which Version of that DODAG it should join. The OF is also used by the RPL Instance to select a number of routers within the DODAG current and subsequent Versions to serve as parents or as feasible successors.

The RPL Instance uses the OF to compute a Rank for the device. This value represents an abstract distance to the root of the DODAG within the DODAG Version. The Rank is exchanged between nodes using RPL and allows other RPL nodes to avoid loops and verify forward progression toward the destination, as specified in [RFC6550]. Regardless of the particular OF used by a node, Rank will always increase; thus, post convergence, loop-free paths are always formed.

The Objective Function Zero (OF0) operates on parameters that are obtained from provisioning, the RPL DODAG Configuration option and the RPL DODAG Information Object (DIO) base container [RFC6550].

The Rank of a node is obtained by adding a strictly positive, indirectly normalized scalar, rank_increase (Section 6.1), to the Rank of a selected preferred parent. The rank_increase is based on a step_of_rank (Section 6.1) normalized scalar that can vary with a ratio from 1 (excellent) to 9 (worst acceptable) to represent the link properties. The step_of_rank can be multiplied by a configurable factor called rank_factor (Section 6.2) that amplifies the rank_increase to reflect the relative preferences between different link types that would be used in the same RPL Instance. The rank_increase can be further adapted as detailed in Section 4.1. By default, OF0 encodes the 2-octet Rank in units of 256, and the default settings allow for the encoding of a minimum of 28 (worst acceptable) hops and a maximum of 255 (excellent) hops.

The RPL specification [RFC6550] requires the use of a common OF by all nodes in a network. The possible use of multiple OFs with a single network is for further study.

The RPL specification [RFC6550] does not include any OF definitions. This is left for other documents specific to different deployments and application environments. Since there is no default OF or metric container in the RPL main specification, it might happen that, unless two given implementations follow the same guidance for a specific problem or environment, those implementations will not support a common OF with which they could interoperate.

OF0 is designed as a default OF that will allow interoperation between implementations in a wide spectrum of use cases. This is why OF0 does not specify how the link properties are transformed into a rank_increase and leaves that responsibility to the implementation; rather, OF0 enforces the values for the rank_increase by normalizing the step_of_rank for a normal link and its acceptable range, as opposed to formulating the details of the step_of_rank computation. This is also why OF0 ignores metric containers.

2. Terminology

The terminology used in this document is consistent with and incorporates that described in "Terminology in Low power And Lossy Networks" [ROLL-TERMS] and [RFC6550].

The term "feasible successor" is used to refer to a neighbor that can possibly be used as a next hop for Upward traffic following the loop avoidance and forwarding rules that the nodes implement and that are defined in the RPL specification [RFC6550].

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 RFC 2119 [RFC2119].

3. Objective Function Zero Overview

The RPL specification describes constraints on how nodes select potential parents, called a parent set, from their neighbors. All parents are feasible successors for upward traffic (towards the root). Additionally, RPL allows the use of parents in a subsequent Version of a same DODAG as feasible successors, in which case this node acts as a leaf in the subsequent DODAG Version.

The Goal of the OF0 is for a node to join a DODAG Version that offers good enough connectivity to a specific set of nodes or to a larger routing infrastructure though there is no guarantee that the path will be optimized according to a specific metric. This validation process for the connectivity is implementation and link type dependent and is out of scope. The validation involves but is not limited to application of [RFC6550], Sections 3.2.3 and 13, as appropriate and may involve deployment specific policies as well.

Thus, for the purpose of OF0, the term "Grounded" [RFC6550] means that the DODAG root provides such connectivity. How that connectivity is asserted and maintained is out of scope.

Objective Function Zero is designed to find the nearest Grounded root. This can be achieved if the Rank of a node is very close to an abstract function of its distance to the root. This need is balanced with the other need of maintaining some path diversity, which may be achieved by increasing the Rank. In the absence of a Grounded root, inner connectivity within the LLN is still desirable and floating DAGs will form, rooted at the nodes with the highest administrative preference.

OF0 selects a preferred parent and a backup feasible successor if one is available. All the upward traffic is normally routed via the preferred parent with no attempt to perform any load balancing. When the link conditions do not let an upward packet through the preferred parent, the packet is passed to the backup feasible successor.

A RPL node monitors links to a number of neighbor nodes and can use OF0 to assign a rank_increase to each link. Though the exact method for computing the rank_increase is implementation dependent, the computation must follow the rules that are specified in Section 4.1.

4. OF0 Operations

4.1. Computing Rank

An OF0 implementation first computes a variable step_of_rank (Section 6.1) associated with a given parent from relevant link properties and metrics. The step_of_rank is used to compute the amount by which to increase the rank along a particular link, as explained later in this section.

Computing a step_of_rank based on a static metric such as an administrative cost implies that the OF0 implementation only considers parents with good enough connectivity, and results in a Rank that is analogous to hop-count. In most LLNs, this favors paths with fewer but longer hops of poorer connectivity; it is thus RECOMMENDED to base the computation of the step_of_rank on dynamic link properties such as the expected transmission count (ETX) metric as introduced in [DeCouto03] and discussed in [RFC6551]. "Minimum Rank Objective Function with Hysteresis" [HYSTERESIS] provides guidance on how link cost can be computed and on how hysteresis can improve Rank stability.

OF0 allows an implementation to stretch the step_of_rank in order to enable the selection of at least one feasible successor and thus maintain path diversity. Stretching the step_of_rank is NOT RECOMMENDED, because it augments the apparent distance from the node to the root, distorts the DODAG from the optimal shape and may cause instabilities due to greedy behaviors whereby depending nodes augment their Ranks to use each other as parents in a loop. Still, an implementation may stretch the step_of_rank with at most a configurable stretch_of_rank (Section 6.2) of any value between 0 (no stretch) and the fixed constant MAXIMUM_RANK_STRETCH (Section 6.3).

An implementation MUST maintain the stretched step_of_rank between the fixed constants MINIMUM_STEP_OF_RANK and MAXIMUM_STEP_OF_RANK (Section 6.3). This range allows the reflection of a large variation of link quality.

The gap between MINIMUM_STEP_OF_RANK and MAXIMUM_RANK_STRETCH may not be sufficient in every case to strongly distinguish links of different types or categories in order to favor, say, powered over battery-operated or high-speed (wired) over lower-speed (wireless) links, within the same DAG. An implementation SHOULD allow the operator to configure a factor called rank_factor (Section 6.2) and to apply the factor on all links and peers to multiply the effect of the stretched step_of_rank in the rank_increase computation as further detailed below.

Additionally, an implementation MAY recognize categories of peers and links, such as different link types, in which case it SHOULD be able to configure a more specific rank_factor to those categories. The rank_factor MUST be set between the fixed constants MINIMUM_RANK_FACTOR and MAXIMUM_RANK_FACTOR (Section 6.3).

The variable rank_increase is represented in units expressed by the variable MinHopRankIncrease, which defaults to the fixed constant DEFAULT_MIN_HOP_RANK_INCREASE ([RFC6550]); with that setting, the least significant octet in the RPL Rank field in the DIO Base Object is not used.

The step_of_rank Sp that is computed for that link is multiplied by the rank_factor Rf and then possibly stretched by a term Sr that is less than or equal to the configured stretch_of_rank. The resulting rank_increase is added to the Rank of preferred parent R(P) to obtain that of this node R(N):

R(N) = R(P) + rank_increase where:

rank_increase = (Rf*Sp + Sr) * MinHopRankIncrease

Optionally, the administrative preference of a root MAY be configured to supersede the goal to join a Grounded DODAG. In that case, nodes will associate with the root with the highest preference available, regardless of whether or not that root is Grounded. Compared to a deployment with a multitude of Grounded roots that would result in the same multitude of DODAGs, such a configuration may result in possibly less but larger DODAGs, as many as roots configured with the highest priority in the reachable vicinity.

4.2. Parent Selection

4.2.1. Selection of the Preferred Parent

As it scans all the candidate neighbors, OF0 keeps the parent that is the best for the following criteria (in order):

1. [RFC6550], Section 8, spells out the generic rules for a node to re-parent and in particular the boundaries to augment its Rank within a DODAG Version. A candidate that would not satisfy those rules MUST NOT be considered.

2. Prior to selecting a router as the preferred parent, an implementation SHOULD validate the connectivity and suitability of the router as discussed in Section 3. This validation involves checking the Layer 2 connectivity to the router, the Layer 3 connectivity offered by the router, and may involve examination of other factors such as locally or globally configured policies.

   In most cases, a router that does not succeed in the validation process cannot be further considered for selection as preferred parent. In any case, a router that succeeded in that validation process SHOULD be preferred over one that did not succeed.

3. When multiple interfaces are available, a policy might be locally configured to order them and that policy applies first; that is, a router on a higher-order interface in the policy is preferable.

4. If the administrative preference of the root is configured to supersede the goal to join a Grounded DODAG, a router that offers connectivity to a more preferable root SHOULD be preferred.

5. A router that offers connectivity to a grounded DODAG Version SHOULD be preferred over one that does not.

6. A router that offers connectivity to a more preferable root SHOULD be preferred.

7. When comparing two parents that belong to the same DODAG, a router that offers connectivity to the most recent DODAG Version SHOULD be preferred.

8. The parent that causes the lesser resulting Rank for this node, as specified in Section 4.1, SHOULD be preferred.

9. A DODAG Version for which there is an alternate parent SHOULD be preferred. This check is OPTIONAL. It is performed by computing the backup feasible successor while assuming that the router that is currently examined is finally selected as preferred parent.

10. The preferred parent that was in use already SHOULD be preferred.

11. A router that has announced a DIO message more recently SHOULD be preferred.

These rules and their order MAY be varied by an implementation according to configured policy.

4.2.2. Selection of the Backup Feasible Successor

When selecting a backup feasible successor, the OF performs in order the following checks:

1. The backup feasible successor MUST NOT be the preferred parent.

2. The backup feasible successor MUST be either in the same DODAG Version as this node or in an subsequent DODAG Version.

3. Along with RPL rules, a Router in the same DODAG Version as this node and with a Rank that is higher than the Rank computed for this node MUST NOT be selected as a feasible successor.

4. A router with a lesser Rank SHOULD be preferred.

5. A router that has been validated as usable by an implementation-dependent validation process SHOULD be preferred.

6. When multiple interfaces are available, a router on a higher order interface is preferable.

7. The backup feasible successor that was in use already SHOULD be preferred.

These rules and their order MAY be varied by an implementation according to configured policy.

5. Abstract Interface to OF0

Objective Function Zero interacts for its management and operations in the following ways:

Processing DIO: When a new DIO is received, the OF that corresponds to the Objective Code Point (OCP) in the DIO is triggered with the content of the DIO. OF0 is identified by OCP 0 (see Section 8).

Providing DAG Information: The OF0 support provides an interface that returns information about a given instance. This includes material from the DIO base header, the role (router, leaf), and the Rank of this node.

Providing a Parent List: The OF0 support provides an interface that returns the ordered list of the parents and feasible successors for a given instance to the RPL core. This includes the material that is contained in the transit option for each entry.

Triggered Updates: The OF0 support provides events to inform it that a change in DAG information or Parent List has occurred. This can be caused by an interaction with another system component such as configuration, timers, and device drivers, and the change may cause the RPL core to fire a new DIO or reset Trickle timers.

6. OF0 Operands

On top of variables and constants defined in [RFC6550], this specification introduces the following variables and constants:

6.1. Variables

OF0 uses the following variables:

step_of_rank (strictly positive integer): an intermediate computation based on the link properties with a certain neighbor.

rank_increase (strictly positive integer): delta between the Rank of the preferred parent and self

6.2. Configurable Parameters

OF0 can use the following optional configurable values that are used as parameters to the rank_increase computation:

stretch_of_rank (unsigned integer): the maximum augmentation to the step_of_rank of a preferred parent to allow the selection of an additional feasible successor. If none is configured to the device, then the step_of_rank is not stretched.

rank_factor (strictly positive integer): A configurable factor that is used to multiply the effect of the link properties in the rank_increase computation. If none is configured, then a rank_factor of 1 is used.

6.3. Constants

Section 17 of [RFC6550] defines RPL constants. OF0 fixes the values of the following constants:

DEFAULT_STEP_OF_RANK: 3

MINIMUM_STEP_OF_RANK: 1

MAXIMUM_STEP_OF_RANK: 9

DEFAULT_RANK_STRETCH: 0

MAXIMUM_RANK_STRETCH: 5

DEFAULT_RANK_FACTOR: 1

MINIMUM_RANK_FACTOR: 1

MAXIMUM_RANK_FACTOR: 4

7. Manageability Considerations

Section 18 of [RFC6550] depicts the management of the protocol. This specification inherits from that section and its subsections, with the exception that metrics as specified in [RFC6551] are not used and do not require management.

7.1. Device Configuration

An implementation SHOULD allows the configuration of at least a global rank_factor that applies to all links. Additionally, the implementation may allow the grouping of interfaces, links, and/or neighbors and configure a more specific rank_factor to such groups.

An implementation MAY allow the configuration of a maximum stretch_of_rank that MUST be less than or equal to MAXIMUM_RANK_STRETCH as discussed in Section 4.1. If none is configured, a value of 0 is assumed and the step_of_rank is not stretched.

An OF0 implementation SHOULD support the DODAG Configuration option as specified in Section 6.7.6 of [RFC6550] and apply the parameters contained therein. As discussed in Section 16 of [RFC6550], this requirement might be overridden by further guidance for certain application scenarios. When the option is used, the parameters are configured to the nodes that may become DODAG roots, and the nodes are configured to redistribute the information using the DODAG Configuration option. In particular, the value of MinHopRankIncrease can be distributed with that option and override the fixed constant of DEFAULT_MIN_HOP_RANK_INCREASE that is defined in Section 17 of [RFC6550] with a fixed value of 256.

Out of the box, that is at initial factory time, the default constant values SHOULD be used, that is:

the rank_factor is set to the fixed constant DEFAULT_RANK_FACTOR (Section 6.3).

the maximum stretch_of_rank is set to the fixed constant DEFAULT_RANK_STRETCH (Section 6.3).

the MinHopRankIncrease is set to the fixed constant DEFAULT_MIN_HOP_RANK_INCREASE ([RFC6550]).

The values can be overridden at any time and apply at the next Version of the DODAG. As discussed in Section 16 of [RFC6550], this requirement might be overridden by further guidance for certain application scenarios.

7.2. Device Monitoring

As discussed in Section 5, the OF support must be able to provide information about its operations and trigger events when that information changes. At a minimum, the information should include:

DAG information as specified in Section 6.3.1 of [RFC6550], and including the DODAGID, the RPLInstanceID, the Mode of Operation, the Rank of this node, the current Version Number, and the value of the Grounded flag.

A list of neighbors indicating the preferred parent and an alternate feasible if available. For each neighbor, the Rank, the current Version Number, and the value of the Grounded flag should be indicated.

8. IANA Considerations

Per this specification, an Objective Code Point (OCP) for OF0 has been assigned in the Objective Code Point Registry as described in Section 20.5 of [RFC6550].

OCP code: 0

Description: A basic Objective Function that relies only on the objects that are defined in [RFC6550].

Defining RFC: RFC 6552

9. Security Considerations

This specification makes simple extensions to RPL and so is vulnerable to and benefits from the security issues and mechanisms described in [RFC6550] and [ROLL-SECURITY]. This document does not introduce new flows or new messages; thus, it requires no specific mitigation for new threats.

OF0 depends on information exchanged in the Rank and OCP protocol elements. If those elements were compromised, then an implementation of OF0 might generate the wrong path for a packet, resulting in it being misrouted. Therefore, deployments are RECOMMENDED to use RPL security mechanisms if there is a risk that routing information might be modified or spoofed.

10. Acknowledgements

Specific thanks to Philip Levis and Phoebus Chen for their help in finalizing this document.

Many thanks also to Adrian Farrel, Tim Winter, JP. Vasseur, Julien Abeille, Mathilde Durvy, Teco Boot, Navneet Agarwal, Meral Shirazipour, and Henning Rogge for in-depth review and first-hand implementers’ feedback.

11. References

11.1. Normative References

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

[RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, March 2012.

11.2. Informative References

[DeCouto03] De Couto, D., Aguayo, D., Bicket, J., and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing", MobiCom '03, The 9th ACM International Conference on Mobile Computing and Networking, San Diego, California, 2003, <http://pdos.csail.mit.edu/papers/grid:mobicom03/paper.pdf>.

[HYSTERESIS] Gnawali, O. and P. Levis, "The Minimum Rank Objective Function with Hysteresis", Work in Progress, May 2011.

[RFC6551] Vasseur, J., Ed., Kim, M., Ed., Pister, K., Dejean, N., and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks", RFC 6551, March 2012.

[ROLL-SECURITY] Tsao, T., Alexander, R., Dohler, M., Daza, V., and A. Lozano, "A Security Framework for Routing over Low Power and Lossy Networks", Work in Progress, March 2012.

[ROLL-TERMS] Vasseur, JP., "Terminology in Low power And Lossy Networks", Work in Progress, September 2011.

Author's Address

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

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