The Direction Field in Routing Metric/Constraint Objects Used in RPL

draft-goyal-roll-metrics-direction-00

The Direction (D) Field in Routing Metric/Constraint Header

- A 2-bit field that indicates the direction associated with the routing metric/ constraint:
  - D = 0x00: undefined
  - D = 0x01: Up
  - D = 0x02: Down
  - D = 0x03: Bidirectional.
- Backward compatible with Metrics draft
  - If the D field has value 0x00, the direction associated with the routing metric/constraint is undefined as in Metrics draft.

Goal

- Many link-level routing metrics have a directional aspect.

- Some applications in the building area networks need to discover directional routes between sensors

- (need similar to former Ticket #32)

A Mechanism to Measure the Quality of a Point-to-point Route in a Low Power and Lossy Network

draft-goyal-roll-p2p-measurement-01

Goal

- This mechanism involves traversing an existing (DAG or P2P based) route and accumulating the routing metrics values along the route.

- The mechanism is required

  – To assess whether a new path needs to be discovered or not

  – To determine the constraints that the new path being discovered must satisfy (P2P)

Functional Overview

- Measuring "origin-to-target" routes:
  - accumulate the metric values en route towards target.
  - target unicasts a Measurement Reply message, carrying the accumulated values

- Measuring "target-to-origin" routes:
  - Unicast Measurement Request towards target.
  - target unicasts a Measurement Reply message, accumulating metric values on the way back to origin

Draft Status

- WG document?

Identifying Defunct DAGs in RPL

draft-goyal-roll-defunct-dags-00

Goal

- No "garbage collection" mechanism currently defined in RPL

- The draft specifies a DIS-based mechanism that allows an RPL node to identify defunct DAGs and delete the state it maintains for such DAGs.

- Useful in LLNs where multiple DAGs are in existence and some DAGs are no longer operational.

Functional Overview

- When no DIO received from any parent in a DAG for a while:
  - send a multicast DIS with N flag set
  - new N flag to avoid unnecessary trickle timer reset
  - if no DIO received from a parent, remove the parent
  - if no parent left, mark the DAG as defunct
    - schedule the deletion of the DAG's state after some time.