H. Rogge
Fraunhofer FKIE
E. Baccelli
INRIA
A. Kaplan
Cert.at/NIC.at, Internet
Verwaltungs und
Betriebsgesellschaft m.b.H.

March 23, 2010

Packet Sequence Number based ETX Metric for Mobile Ad Hoc Networks
draft-funkfeuer-manet-olsrv2-etx-01

Abstract

This document specifies the ETX metric and its usage in OLSRv2.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The Funkfeuer [FUNKFEUER] and Freifunk networks [FREIFUNK] are OLSR-based [RFC3626] wireless community networks with hundreds of routers in permanent operation. The Vienna Funkfeuer network in Austria, for instance, consists of 400 routers (around 600 routes) covering the whole city of Vienna and beyond, spanning roughly 40km in diameter. It has been in operation since 2003 and supplies its users with Internet access. The particularity of the Vienna Funkfeuer network is that it manages to provide Internet access through a city wide, large scale Wi-Fi mesh cloud, with just a single uplink.

Operational experience with such a network has revealed that the use of hop-count as routing metric leads to unsatisfactory network performance (especially when going through a single uplink). Experiments with the ETX metric [MOBICOM03] were therefore undertaken in parallel in the Berlin Freifunk network as well as in the Vienna Funkfeuer network, and found satisfactory, i.e. sufficiently easy to implement and providing sufficiently good performance. This metric has now been in operational use in these networks for more than 2 years.

The ETX metric of a link is the estimated number of transmissions required to successfully send a packet (each packet smaller than MTU) over that link, until an acknowledgement is received. The ETX metric is additive, i.e. the ETX metric of a path is the sum of the ETX metrics for each link on this path.

This document describes the ETX metric as used by the Funkfeuer network, and specifies its usage in OLSRv2 [olsrv2]. More precisely, this document specifies additional signaling and processing to NHDP [nhdp] in order to establish the ETX metric value for a link.

In order to use the ETX metric for routers, this document assumes that the suggestions in [olsrv2-metric] are incorporated into [olsrv2].

2. Terminology

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[RFC2119].

The terminology introduced in [RFC5444], [olsrv2], [nhdp], and [olsrv2-metric], including the terms "packet", "message", "Address Block", "TLV Block","TLV", "address", "address prefix", and "address

object" are to be interpreted as described therein.

Additionally, this document uses the following terminology and notational conventions:

LIFO - a last in, first out queue of numbers.

LIFO[current] - the most recent entry added to the queue.

push(LIFO, value) - an operation which removes the oldest entry in the queue and place a new entry at the head of the queue.

sum(LIFO) - an operation which returns the sum of all elements in a LIFO.

diff_seqno(new, old) - an operation which returns the positive distance between two elements of the circular sequence number space defined in [RFC5444]. Its value is either (new - old) if this result is positive, or else its value is (new - old + 65536).

UNDEFINED - a constant for -1.

3. Applicability Statement

The mechanism specified in this document is used daily by hundreds of routers in the Funkfeuer network [FUNKFEUER], as well as in similar OLSR-based wireless community networks which use the OLSR.org [OLSR.org] code base, such as [FREIFUNK], [AWMN], [NINUX], [GUIFI], [OPENAIR]. Operational experience suggests that this mechanism is viable in (at least) these kinds of networks.

The ETX metric value of a link is established by measuring the rate of successful exchange of OLSRv2 control packets over that link, which use the format defined in [RFC5444]. ETX metric computation is thus based only on layer 3 signaling, and is therefore independent from underlying link layer technologies. Moreover, ETX metric computation does not require inspection of data traffic.

If a router does not implement the use of the ETX metric, it falls back to DEFAULT_METRIC as defined in [olsrv2-metric], and this default behaviour is taken into account by other routers, with which this router has a link.

4. Protocol Functioning & Overview

Router A computes the value of the ETX metric of its link to router B

by continuously estimating the loss rates over this link, in both directions: from B to A (this rate is called R_etx), and from A to B (this rate is called D_etx). Router A computes R_etx as the measured proportion of [RFC5444] packets successfully arriving from B, and signals this value in NHDP HELLO messages by inclusion of a R_etx TLV. Symmetrically, router B computes the proportion of [RFC5444] packets successfully arriving from A, and signals its value in NDHP HELLO messages by inclusion of a R_etx TLV, which router A can then take as D_etx value for this link.

The value of the ETX metric of the link is then ETX = R_etx * D_etx, which corresponds to the expected number of attempts to successfully receive and acknowledge a packet over this link. Note that this metric is symmetric, i.e. on a link connecting router A and router B, the ETX metric value for this link computed by router B will be the same as the ETX metric value computed by router A.

5. Data Structures

This specification extends the Link Set per Interface Information Base, as defined in [nhdp], with the following additional elements for each link tuple:

L_METRIC_received_lifo - a LIFO queue with ETX_MEMORY_LENGTH integer elements. Each entry contains the number of successfully received [RFC5444] packets within an interval of ETX_METRIC_INTERVAL.

L_METRIC_total_lifo - a LIFO queue with ETX_MEMORY_LENGTH integer elements. Each entry contains the estimated number of [RFC5444] packets transmitted by the neighbor, based on the received packet sequence numbers within an interval of ETX_METRIC_INTERVAL.

L_METRIC_last_pkt_seqno - the last received packet sequence number received from this link.

L_METRIC_r_etx - the current r_etx value for this link to the neighbor.

L_METRIC_d_etx - the last d_etx value received from the neighbor for this link.

L_METRIC_hello_time - time when the next hello will be expected. This is used to detect missing hellos by timeout.

L_METRIC_hello_interval - the interval between two hello messages of the links neighbor.

L_METRIC_lost_hellos - the estimated number of lost hello messages from this neighbor, based on the value of the hello interval.

6. Initial Values

When generating a new tuple in the Link Set, the values of the elements specified above are set as follows:

L_METRIC_received_lifo := 0, ..., 0.

L_METRIC_total_lifo := 0, ..., 0.

L_METRIC_last_pkt_seqno := UNDEFINED.

L_METRIC_r_etx := UNDEFINED.

L_METRIC_d_etx := UNDEFINED.

L_METRIC_hello_time := EXPIRED.

L_METRIC_hello_interval := UNDEFINED.

L_METRIC_lost_hellos := 0

7. Protocol Parameters

This specification uses the parameters defined in [olsrv2]. This specification defines the following additional parameters:

ETX_MEMORY_LENGTH - ETX algorithm memory length in seconds. All received and lost packets within this time period are used to calculate the delivery ratio R_etx.

ETX_METRIC_INTERVAL - interval in seconds between two metric recalculations as described in Section 10.

ETX_SEQNO_RESTART_DETECTION - threshold in number of missing packets (based on received packet sequence numbers) at which point the router considers the neighbor has restarted.

ETX_HELLO_TIMEOUT_FACTOR - timeout factor for HELLO interval at which point a HELLO is definitely considered lost. The value should be between 1.0 and (2.0 * (1 - HT_MAXJITTER/ HELLO_INTERVAL)).

ETX_PERFECT_METRIC - metric value for ETX 1.0.

8. Packets and Messages

Generated packets and messages use the format defined in [RFC5444]. The present specification adds the following constraints:

- o A packet MUST contain a packet sequence number as defined in [RFC5444]. This sequence number MUST be interface specific.

8.1. HELLO Message Generation

HELLO messages are generated as specified in [olsrv2]. In addition, the HELLO messages must comply with the following:

- o For each address included in a HELLO message with a TLV with type LINK_STATUS and value SYMMETRIC or HEARD, a TLV of type R_etx MUST also be included.

R_etx TLV formatting is specified in Table 1, whereby the value of the directional link cost is encoded as TimeTLV [RFC5497] encoded values with C = 1024.

Type	Value Length	Value
R_etx	1 octet	linkcost

Table 1

The value of the linkcost field of an R_etx TLV in a HELLO message is set to the L_METRIC_r_etx value of the corresponding link listed in this HELLO message.

8.2. HELLO Message Processing

HELLO messages are first processed as specified in [olsrv2]. This processing includes identifying (or creating) a Neighbor Tuple corresponding to the originator of the HELLO message (the "current Neighbor Tuple"). After this, the following MUST be performed:

1. If the IP address of this link local interface is included in the HELLO address block and the address block contains an R_etx address TLV, then:

    1. L_METRIC_d_etx := R_etx.

2. Otherwise:

    1. L_METRIC_d_etx := UNDEFINED.

3. If the HELLO message contains an INTERVAL_TIME TLV, then:

    1. L_METRIC_hello_interval := INTERVAL_TIME.

4. If L_METRIC_hello_interval != UNDEFINED, then:

    1. L_METRIC_hello_time := current_time + ETX_HELLO_TIMEOUT_FACTOR * INTERVAL_TIME.

5. L_METRIC_lost_hellos := 0.

8.3. Packet Processing

Each incoming packet is processed as defined in OLSRv2 [olsrv2]. Furthermore, the following additional processing MUST be carried out after the package has been processed on the link set tuple corresponding to the source of the packet:

1. If L_METRIC_last_pkt_seqno = UNDEFINED, then:

    1. L_METRIC_received_lifo[current] := 1.

    2. L_METRIC_total_lifo[current] := 1.

2. Otherwise:

    1. L_METRIC_received_lifo[current] := L_METRIC_received_lifo[current] + 1.

    2. diff := seq_diff(seqno, L_METRIC_last_pkt_seqno).

    3. If diff > ETX_SEQNO_RESTART_DETECTION, then:

        1. diff := 1.

    4. L_METRIC_total_lifo[current] := L_METRIC_total_lifo[current] + diff.

3. L_METRIC_last_pkt_seqno := seqno.

9. HELLO Timeout

When L_METRIC_hello_time has timed out, the following step MUST be done:

1. L_METRIC_lost_hellos := L_METRIC_lost_hellos + 1.

2. L_METRIC_hello_time := L_METRIC_hello_time + L_METRIC_hello_interval.

10. Periodic Metric Computation

This metric stores the number of received packets per link to a neighbor and use the package sequence number to calculate the total number of sent packages of the neighbor. The total number of packages divided by the number of received packages is used as a cost metric for the link.

If a link to a node is lost, no packets are received anymore and neither the received not total value of packages will change. To prevent a constant result in this case, the metric stores the number of HELLO message interval timeouts since the last received packet from a neighbor and use this value to reduce the received packet count proportionally to the length of the metric memory ETX_MEMORY_LENGTH.

Once every ETX_METRIC_INTERVAL, this protocol MUST recalculate of all L_METRIC_r_etx in all Link Set entries:

1. sum_received := sum(L_METRIC_total_lifo).

2. sum_total := sum(L_METRIC_received_lifo).

3. If L_METRIC_hello_interval != UNDEFINED and L_METRIC_lost_hellos > 0, then:

    1. penalty := L_METRIC_hello_interval * L_METRIC_lost_hellos / ETX_MEMORY_LENGTH.

    2. sum_received := sum_received - sum_received * penalty;

4. If sum_received < 1, then:

1. L_METRIC_r_etx := UNDEFINED.

2. L_in_metric := MAXIMUM_METRIC.

5. Otherwise:

1. L_METRIC_r_etx := sum_total / sum_received.

2. If L_METRIC_d_etc = UNDEFINED, then:

1. L_in_metric := DEFAULT_METRIC,

3. Otherwise:

1. L_in_metric := ETX_PERFECT_METRIC * L_METRIC_r_etx * L_METRIC_d_etx.

6. push(L_METRIC_total_lifo, 0)

7. push(L_METRIC_received_lifo, 0)

11. Proposed Values for Parameters and Constants

This section proposes values for various parameters used in this specification.

- ETX_MEMORY_LENGTH := 32 seconds

- ETX_METRIC_INTERVAL := 1 second

- ETX_SEQNO_RESTART_DETECTION := 256

- ETX_HELLO_TIMEOUT_FACTOR := 1.5

- ETX_PERFECT_METRIC := DEFAULT_METRIC * 0.1

12. Security Considerations

Artificial manipulation of metrics values can drastically alter network performance. In particular, advertising a higher R_etx value may decrease the amount of incoming traffic, while advertising lower R_etx may decrease the amount of incoming traffic. By artificially increasing or decreasing the R_etx values it advertises, a rogue router may thus attract or repulse data traffic. A rogue router may then potentially degrade data throughput by not forwarding data as it should or redirecting traffic into routing loops or bad links.

13. IANA Considerations

This specification defines one Address Block TLV Type, which have been allocated from the "Assigned Address Block TLV Types" repository of [RFC5444] as specified in Table 2.

13.1. Expert Review: Evaluation Guidelines

For the registries for TLV Type Extensions where an Expert Review is required, the designated expert SHOULD take the same general recommendations into consideration as are specified by [RFC5444].

13.2. Address Block TLV Types

Name	Type	Type extension	Description
R_etx	tbd	tbd	Loss rate of incoming [RFC5444] packets.

Table 2

14. Acknowledgements

The authors would like to acknowledge the network administrators from Freifunk Berlin [FREIFUNK] and Funkfeuer Vienna [FUNKFEUER] for endless hours of testing and suggestions to improve the quality of this ETX metric.

The authors would like to gratefully acknowledge the following people for intense technical discussions, early reviews and comments on the specification and its components (listed alphabetically): Teco Boot (Infinity Networks), Thomas Clausen (LIX), Christopher Dearlove (BAE Systems) Michael Gerharz (Fraunhofer FKIE), Ulrich Herberg (LIX), Markus Kittenberger (Funkfeuer Vienna) and Jens Toelle (FKIE).

15. References

15.1. Normative References

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

[RFC3626] Clausen, T. and P. Jacquet, "Optimized Link State Routing

Protocol", RFC 3626, October 2003.

[RFC5444] Clausen, T., Dearlove, C., Dean, J., and C. Adjih, "Generalized Mobile Ad Hoc Network (MANET) Packet/Message Format", RFC 5444, 2009.

[RFC5497] Clausen, T. and C. Dearlove, "Representing Multi-Value Time in Mobile Ad Hoc Networks", RFC 5497, March 2009.

15.2. Informative References

[olsrv2] Clausen, T., Jacquet, P., and C. Dearlove, "The Optimized Link State Routing Protocol version 2", draft-ietf-manet-olsrv2-10 , September 2009.

[nhdp] Clausen, T., Dearlove, C., and J. Dean, "Mobile Ad Hoc Network (MANET) Neighborhood Discovery Protocol (NHDP)", draft-ietf-manet-nhdp-11 , October 2009.

[MOBICOM03] De Couto, D., Aguayo, D., Bicket, J., and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing", Proceedings of the MOBICOM Conference , 2003.

[olsrv2-metric] Dearlove, C., Clausen, T., and P. Jacquet, "Link Metrics for OLSRv2".

[OLSR.org] "The olsr.org OLSR daemon, http://www.olsr.org".

[FREIFUNK] "Freifunk Wireless Community Networks, http://www.freifunk.net".

[FUNKFEUER] "Austria Wireless Community Network, http://www.funkfeuer.at".

[AWMN] "Athens Wireless Community Network, http://awmn.net".

[GUIFI] "Barcelona Wireless Community Network, http://www.guifi.net".

[NINUX] "Roma Wireless Community Network, http://www.ninux.org".

[OPENAIR] "Boston Wireless Community Network, http://openairboston.net/".

Authors' Addresses

Henning Rogge
Fraunhofer FKIE

Email: henning.rogge@fkie.fraunhofer.de

Emmanuel Baccelli
INRIA

Email: Emmanuel.Baccelli@inria.fr
URI: http://www.emmanuelbaccelli.org/

L. Aaron Kaplan
Cert.at/NIC.at, Internet Verwaltungs und Betriebsgesellschaft m.b.H.

Email: aaron@lo-res.org
URI: http://cert.at/