Networking Working Group
Internet-Draft
Intended status: Informational
Expires: January 3, 2013

JP. Vasseur, Ed.
J. Hui, Ed.
S. Dasgupta
G. Yoon
Cisco Systems, Inc
July 5, 2012

RPL deployment experience in large scale networks
draft-hui-vasseur-roll-rpl-deployment-01

Abstract

Low power and Lossy Networks (LLNs) exhibit characteristics unlike other more traditional IP links. LLNs are a class of network in which both routers and their interconnect are resource constrained. LLN routers are typically resource constrained in processing power, memory, and energy (i.e. battery power). LLN links are typically exhibit high loss rates, low data rates, are are strongly affected by environmental conditions that change over time. LLNs may be composed of a few dozen to thousands of routers. A new protocol called the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) has been specified for routing in LLNs supporting multipoint-to-point, point-to-multipoint traffic, and point-to-point traffic. Since RPL's publication as an RFC, several large scale networks have been succesfully deployed. The aim of this document is to provide deployment experience on real-life deployed RPL-based networks.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 3, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
2. Objective of this document 6
3. RPL Paramaters Settings 7
4. Network Characteristics 8
5. Performance Results 8
6. IANA Considerations 9
7. Security Considerations 9
8. Acknowledgements 9
9. References 9
9.1. Normative references 9
9.2. Informative references 10
Authors' Addresses 11

1. Introduction

Low power and Lossy Networks (LLNs) exhibit characteristics unlike other more traditional IP links. LLNs are a class of network in which both routers and their interconnect are resource constrained. LLN routers are typically resource constrained in processing power, memory, and energy (i.e. battery power). LLN links are typically exhibit high loss rates, low data rates, are are strongly affected by environmental conditions that change over time. LLNs may be composed of a few dozen to thousands of routers.

A new protocol called the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) has been specified for routing in LLNs supporting multipoint-to-point, point-to-multipoint traffic, and point-to-point traffic [RFC6550]. Since RPL's publication as an RFC, several large scale networks have been successfully deployed. The aim of this document is to provide deployment experience on real-life deployed RPL-based networks.

In addition to [RFC6550], companion documents have been defined that specify IPv6 packet options required for the proper operation of RPL, including [RFC6553] and [RFC6554].

This document makes use of the terminology defined in [I-D.ietf-roll-terminology].

RPL is a distance-vector routing protocol that builds a Destination Oriented Directed Acyclic Graph (DODAG) according to an Objective Function (OF). The OF is a defined set of rules that optimize paths against a set of metrics and/or constraints. A very basic OF, known as OF0, is specified in [RFC6552]. More involved OFs may be specified, such as the Minimum Rank with Hysteresis Objective Function specified in [I-D.ietf-roll-minrank-hysteresis-of].

Routing requirements documents spelled out in [RFC5673], [RFC5826], [RFC5548] and [RFC5867]) observe that it must be possible to take into account a variety of node metrics and/or constraints during path computation. Thus, a number of routing metrics and constraints for RPL have been specified in [RFC6551] for maximum flexibility according to the objectives and environment of the LLN.

RPL supports efficient loop detection using data-path route validation and supports both local and global route repair operations.

RPL makes use of the Trickle algorithm, which provides a density-aware mechanism for distributing and maintaining state within a network [RFC6206]. With simple local rules, the Trickle algorithm

adjusts the transmission period and suppresses redundant transmissions to minimize control traffic overhead in the steady state while propagating new information quickly. Trickle's suppression mechanisms ensures that control message overhead grows logrithmically with node density.

In maintaining point-to-multipoint routes, RPL supports two modes of operations: non-storing and storing. In both cases, the DODAG built by RPL according to the OF is used for hop-by-hop upstream routing towards the DAG Root. In non-storing mode, only the DAG Root maintains downward routes and all data packets must traverse the DAG Root. In storing mode, LLN routers also maintain downward routing state, allowing each LLN router to forward data packets to devices in their sub-DAG. LLN constraints, the network objective, and overall environment typically drives the choice of non-storing or storing mode and is left to the network administrator.

RPL, like many other routing protocols, is designed to be deployed in a number of different operational environments and [RFC6550] specifies a number of configuration parameters. Section 17 of [RFC6550] lists the following RPL constants and variables:

- o DEFAULT_PATH_CONTROL_SIZE: This is the default value used to configure PCS in the DODAG Configuration option, which dictates the number of significant bits in the Path Control field of the Transit Information option. DEFAULT_PATH_CONTROL_SIZE has a value of 0. This configures the simplest case limiting the fan-out to 1 and limiting a node to send a DAO message to only one parent.

- o DEFAULT_DIO_INTERVAL_MIN: This is the default value used to configure Imin for the DIO Trickle timer. DEFAULT_DIO_INTERVAL_MIN has a value of 3. This configuration results in Imin of 8 ms.

- o DEFAULT_DIO_INTERVAL_DOUBLINGS: This is the default value used to configure Imax for the DIO Trickle timer. DEFAULT_DIO_INTERVAL_DOUBLINGS has a value of 20. This configuration results in a maximum interval of 2.3 hours.

- o DEFAULT_DIO_REDUNDANCY_CONSTANT: This is the default value used to configure k for the DIO Trickle timer. DEFAULT_DIO_REDUNDANCY_CONSTANT has a value of 10. This configuration is a conservative value for Trickle suppression mechanism.

- o DEFAULT_MIN_HOP_RANK_INCREASE: This is the default value of MinHopRankIncrease. DEFAULT_MIN_HOP_RANK_INCREASE has a value of 256. This configuration results in an 8-bit wide integer part of

Rank.

- o DEFAULT_DAO_DELAY: This is the default value for the DelayDAO Timer. DEFAULT_DAO_DELAY has a value of 1 second.

- o DIO Timer: One instance per DODAG of which a node is a member. Expiry triggers DIO message transmission. A Trickle timer with variable interval in [0, DIOIntervalMin..2^DIOIntervalDoublings].

- o DAG Version Increment Timer: Up to one instance per DODAG of which the node is acting as DODAG root. May not be supported in all implementations. Expiry triggers increment of DODAGVersionNumber, causing a new series of updated DIO message to be sent. Interval should be chosen appropriate to propagation time of DODAG and as appropriate to application requirements (e.g., response time versus overhead).

- o DelayDAO Timer: Up to one timer per DAO parent (the subset of DODAG parents chosen to receive destination advertisements) per DODAG. Expiry triggers sending of DAO message to the DAO parent.

- o RemoveTimer: Up to one timer per DAO entry per neighbor (i.e., those neighbors that have given DAO messages to this node as a DODAG parent). Expiry may trigger No-Path advertisements or immediately deallocate the DAO entry if there are no DAO parents.

Please refer to the .pdf version of this document to see the figures referred in further sections.

2. Objective of this document

Since its specification as a standard track RFC in March 2012, a number of RPL-based networks have been deployed in the field, some of small size, others of large scale. The aim of this document is to describe the successful deployment of a RPL-based LLN with 1,000 nodes. Other networks of even larger scale (5,000 to 10,000 nodes) are in progress and further revisions of this document will include their details.

It is nearly impossible to characterize the absolute performance of a protocol without looking at all the environmental factors and a large number of performance metrics. Furthermore such performance metric not only depends on the environment but also how the various protocol parameters have been configured. Similarly it would not make any sense to provide hard numbers on a performance characteristic of a protocol. For example, Open Shortest Path First (OSPF) routing protocol [RFC2328] may provide convergence times varying between few dozens of milliseconds to seconds depending on the network characteritics and protocol parameters. At one end of the spectrum, fast failure detection with fast Hellos or the use of other protocols

such as Bidirectional Forwarding Detection (BFD) [RFC5880], combined with fast LSA generation, LSA prioritization, fast SPF triggering and an optimized SPF calculation (potentially combined with incremental SPF) would lead to a few dozens of milliseconds of convergence times. At the other end of the spectrum slow detection of failure, combined with low priority trigger of Link State Advertisement (LSA), poor implementation of the Shortest Path First (SPF) algorithm, long propagation delays, lack of LSA control plane packets may lead to covergence times of seconds!

While convergence time is not the critical performance metric in many LLN deployments, the convergence time example provided above is one that illustrates the challenge of providing performance results. This challenge generally applies to most other performance metrics.

As a result, the aim of this document is not to provide absolute performance numbers or parameter setting recommendations, but rather to share successful experience of the large scale deployment of RPL in a real-life deployment scenario.

To that end, we first provide several network characteristics such as the network topology, distribution of the link quality providing the link quality distribution according to the Expected Transmission count (ETX) link metric computed by the RPL nodes. Then we provide indications of how RPL was used in that particular network before showing several performance metrics observed in this network.

3. RPL Paramaters Settings

This RPL network includes the following parameter settings:

- o The Mode of Operation (MoP) is set to non-storing mode.
- o Both local and global repair mechanisms are implemented. Note, however, because the network operates in non-storing mode, local repair simply poisons routes and does not create floating DAGs.
- o MaxRankIncrease is set to 0, which significantly reduces the possibility of routing loops but also limits the capabilities of local repair.
- o The OF is the Minimum Rank with Hysteresis Object Function using the ETX metric.

4. Network Characteristics

This network comprises one thousand nodes and the distribution of the hop counts is shown is Figure 1. This has been obtained by observing the to pology (shown in Figure 2) for a period of 24 hours and tracking the hop count of all the nodes every 5 minutes. It can be seen that approximately 51% of the nodes are 1 hop away and 30% of nodes are 2 hops away on average. Another way of saying this is that approximately 81% of the nodes are 2 hops or less from the root. A snapshot of the network topology (the DODAG built by RPL) is depicted in Figure 2.

Figure 1: Distribution of average hop count of nodes observed over a 24-hour period.

Figure 2: DODAG Topology built by RPL

As with any LLN, one can observe that the some links are of good quality while others provides low path delivery rates: this can be seen by observing the link ETX in Figure 3. Note that we observed transient periods during which the ETX was much higher with links providing even intermittent connectivity (which is not always reflected in the ETX value due to the computation of the ETX is using moving averages to avoid network oscillations and over-reactions). Figure 3 was obtained by observing all the nodes periodically for a 24 hour period. We tracked the maximum and minimum ETX seen by the node as well. From the figure, we can see that almost 90% of the nodes had an average ETX of 1.25 or less over the 24 hour period.

Figure 3: Distribution of average, maximum and minimum ETX observed over a 24-hour period.

The LLN routers communicate using IEEE 802.15.4g links. Operating in the 902-928 MHz US ISM band, the links have an effective data rate of 75 kbps and employ frequency hopping to communicate across 64 channels with 400 kHz channel spacing.

In summary, it can be observed that the network is indeed a LLN, with lossy links.

The network is made of constrained nodes with limited processing power and available memory. The root slightly less constrained and main powered.

It is worth pointing out that the high density of this topology added stress on the routing protocol.

5. Performance Results

As pointed out in Section 1, there is not a single performance metric that could be provided to characterize the routing protocol performance.

Deep analysis of a number of network management events, logs on routers, and packet inspection operation have shown that the routing topology was quite stable even during unstable conditions. More importantly the observed packet delivery rate was always above 99%: by contrast with non LLN networks where the routing protocol is

rarely responsible for non packet delivery because of the absence of

routes to reach a destination, several LLN routing protocols have reported low delivery packet rates because of routing issues. In this particular network, the packet delivery rate was as high as 99% in all cases (link local packet retransmissions were handled by the IEEE 802.15.4 reliable link layer).

Since questions were raised in the past about the RPL control plane overhead although RPL has been designed for low overhead, we paid a particular attention to this performance metric. RPL has been designed to optimize the control plane overhead (thanks to the use of Neighbor Unreachability Detection (NUD) instead of routing hello packet to detect link/node failure, use of trickle algorithm for the transmission of the DIO packets, ...).

Thus we show in Figure 4 the RPL control traffic overhead (both the DIO and the DAO are shown in the network) relative to the available bandwidth provided by the links. Note that the RPL control plane traffic was observed on the most congested area of the network (on the DODAG root).

6. IANA Considerations

No action is required from IANA.

7. Security Considerations

This document provides informational data about existing deployments, thus security considerations do not apply.

8. Acknowledgements

The authors would like to acknowledge the contributions of Ibrahim Mortada for his very valuable contribution.

9. References

9.1. Normative references

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

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC6550] Winter, T., Thubert, P., 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.

9.2. Informative references

[I-D.ietf-roll-minrank-hysteresis-of] Gnawali, O. and P. Levis, "The Minimum Rank with Hysteresis Objective Function", draft-ietf-roll-minrank-hysteresis-of-11 (work in progress), June 2012.

[I-D.ietf-roll-terminology] Vasseur, J., "Terminology in Low power And Lossy Networks", draft-ietf-roll-terminology-06 (work in progress), September 2011.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[RFC5548] Dohler, M., Watteyne, T., Winter, T., and D. Barthel, "Routing Requirements for Urban Low-Power and Lossy Networks", RFC 5548, May 2009.

[RFC5673] Pister, K., Thubert, P., Dwars, S., and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks", RFC 5673, October 2009.

[RFC5826] Brandt, A., Buron, J., and G. Porcu, "Home Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5826, April 2010.

[RFC5867] Martocci, J., De Mil, P., Riou, N., and W. Vermeylen, "Building Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5867, June 2010.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.

[RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko, "The Trickle Algorithm", RFC 6206, March 2011.

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

[RFC6552] Thubert, P., "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)",

RFC 6552, March 2012.

[RFC6553] Hui, J. and JP. Vasseur, "The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams", RFC 6553, March 2012.

[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, March 2012.

Authors' Addresses

JP Vasseur (editor)
Cisco Systems, Inc
11, Rue Camille Desmoulins
Issy Les Moulineaux, 92782
France

Email: jpv@cisco.com

Jonathan Hui (editor)
Cisco Systems, Inc
560 McCarthy Blvd.
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: johui@cisco.com

Sukrit Dasgupta
Cisco Systems, Inc
300 Beaver Brook Road
BOXBOROUGH, MASSACHUSETTS 01719
UNITED STATES

Email: sukdasgu@cisco.com

Giyoung Yoon
Cisco Systems, Inc
560 McCarthy Blvd.
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: giyoon@cisco.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: February 1, 2013

M. Richardson
SSW

July 31, 2012

ROLL Applicability Statement Template
draft-richardson-roll-applicability-template-00

Abstract

This document is a template applicability statement for the Routing over Low-power and Lossy Networks (ROLL) WG.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 1, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
1.1. Requirements Language 4
1.2. Required Reading 4
1.3. Out of scope requirements 4
2. Deployment Scenario 5
2.1. Network Topologies 5
2.2. Network Topologies 5
2.2.1. Traffic Characteristics 5
2.2.2. General 5
2.2.3. Source-sink (SS) communication paradigm 5
2.2.4. Publish-subscribe (PS, or pub/sub) communication paradigm 5
2.2.5. Peer-to-peer (P2P) communication paradigm 5
2.2.6. Peer-to-multipeer (P2MP) communication paradigm . . . 5
2.2.7. Additional considerations: Duocast and N-cast 5
2.2.8. RPL applicability per communication paradigm 5
2.3. Layer 2 applicability. 5
3. Using RPL to Meet Functional Requirements 6
4. RPL Profile 7
4.1. RPL Features 7
4.1.1. RPL Instances 7
4.1.2. Storing vs. Non-Storing Mode 7
4.1.3. DAO Policy 7
4.1.4. Path Metrics 7
4.1.5. Objective Function 7
4.1.6. DODAG Repair 7
4.1.7. Multicast 7
4.1.8. Security 7
4.1.9. P2P communications 7
4.2. Layer-two features 7
4.2.1. Need layer-2 expert here. 7
4.2.2. Security functions provided by layer-2. 7
4.2.3. 6LowPAN options assumed. 7
4.2.4. MLE and other things 7
4.3. Recommended Configuration Defaults and Ranges 7
4.3.1. Trickle Parameters 7
4.3.2. Other Parameters 7
5. Manageability Considerations 8
6. Security Considerations 9
6.1. Security Considerations during initial deployment 9
6.2. Security Considerations during incremental deployment . . 9
7. Other Related Protocols 10
8. IANA Considerations 11
9. Acknowledgements 12
10. References 13
10.1. Informative References 13

10.2. Normative References 13
11. Normative references 14
Author’s Address 15

1. Introduction

Hello.

1.1. Requirements Language

(RFC2119 reference)

1.2. Required Reading

References/Overview of requirements documents, both IETF and industry group. (two pages maximum. This text should be (very) technical, should be aimed at IETF *participants*, not industry group participants, and should explain this industries' specific issues)

1.3. Out of scope requirements

This should list other documents (if any) which deal with situations where things are not in scope for this document.

(For instance, the AMI document tries to cover both line-powered urban metering networks, and energy-constrained metering networks, and also tries to deal with rural requirements. This should be three or four documents, so this section should list the limits of what this document covers)

2. Deployment Scenario

2.1. Network Topologies

describe a single scenario, with possibly multiple topologies that a single utility would employ.

2.2. Network Topologies

2.2.1. Traffic Characteristics

Explain what kind of traffic is being transmitted, where it is initiated, and what kinds of protocols (CoAP, multicast, HTTPS, etc.) are being used. Explain what assumptions are being made about authentication and authorization in those protocols.

2.2.2. General

2.2.3. Source-sink (SS) communication paradigm

2.2.4. Publish-subscribe (PS, or pub/sub) communication paradigm

2.2.5. Peer-to-peer (P2P) communication paradigm

2.2.6. Peer-to-multipeer (P2MP) communication paradigm

2.2.7. Additional considerations: Duocast and N-cast

2.2.8. RPL applicability per communication paradigm

2.3. Layer 2 applicability.

Explain what layer-2 technologies this statement applies to, and if there are options, they should be listed generally here, and specifically in section 4.2.

3. Using RPL to Meet Functional Requirements

This should explain in general terms how RPL is going to be used in this network topology. If trees that are multiple layers deep are expected, then this should be described so that the fan out is understood. Some sample topologies (from simulations) should be explained, perhaps with images references from other publications.

This section should tell an *implementer* in a lab, having a simulation tool or a building/city/etc. to use as a testbed, how to construct an LLN of sufficient complexity (but not too much) to validate an implementation.

4. RPL Profile

This section should list the various features of RPL plus other layers of the LLN, and how they will be used.

4.1. RPL Features

4.1.1. RPL Instances

4.1.2. Storing vs. Non-Storing Mode

4.1.3. DAO Policy

4.1.4. Path Metrics

4.1.5. Objective Function

4.1.6. DODAG Repair

4.1.7. Multicast

4.1.8. Security

4.1.9. P2P communications

4.2. Layer-two features

4.2.1. Need layer-2 expert here.

4.2.2. Security functions provided by layer-2.

4.2.3. 6LowPAN options assumed.

4.2.4. MLE and other things

4.3. Recommended Configuration Defaults and Ranges

4.3.1. Trickle Parameters

4.3.2. Other Parameters

5. Manageability Considerations

6. Security Considerations

6.1. Security Considerations during initial deployment

(This section explains how nodes get their initial trust anchors, initial network keys. It explains if this happens at the factory, in a deployment truck, if it is done in the field, perhaps like http://www.lix.polytechnique.fr/hipercom/SmartObjectSecurity/papers/CullenJennings.pdf)

6.2. Security Considerations during incremental deployment

(This section explains how that replaces a failed node takes on the dead nodes' identity, or not. How are nodes retired. How are nodes removed if they are compromised)

7. Other Related Protocols

8. IANA Considerations

9. Acknowledgements

10. References

10.1. Informative References

10.2. Normative References

11. Normative references

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

Author’s Address

Michael C. Richardson
Sandelman Software Works
470 Dawson Avenue
Ottawa, ON K1Z 5V7
CA

Email: mcr+ietf@sandelman.ca
URI: http://www.sandelman.ca/

ROLL
Internet-Draft
Intended status: Informational
Expires: January 12, 2013

P. van der Stok, Ed.
vanderstok consultancy
E. Dijk
A. Lelkens
Philips Research
July 11, 2012

Multicast requirements for control over LLN
draft-vanderstok-roll-mcreq-02

Abstract

This is a working document intended to focus discussion on requirements for multicast in Low-power and Lossy Networks in the area of M2M communication for control applications. The Trickle algorithm, which uses random re-broadcasting to assure that messages arrive at all destinations, has been proposed in the Trickle Multicast Forwarding ROLL WG draft as the basis for a multicast routing protocol. In this draft additional requirements on multicast routing are presented, such as timeliness, motivated by building control. Random re-broadcasting and timeliness can be difficult to reconcile. This draft presents some simulation results in typical control settings which show that achieving latencies below 400 ms is feasible with Trickle. Recommendations are proposed for the current Trickle Multicast Forwarding draft to achieve optimal performance and meet the stated requirements.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 12, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
  1.1. Terminology 3
  1.2. Motivation 4
2. Application characteristics 4
3. Multicast requirements 6
4. Performance of Trickle-based multicast 7
  4.1. Reasons for using Trickle 7
  4.2. Simulation setup 7
  4.3. Simulation results 8
  4.4. Simulation conclusions 9
5. Performance issues of Trickle Multicast Forwarding 9
  5.1. Redundancy of Trickle ICMP message 10
  5.2. Ability to configure forwarders as data sinks 11
  5.3. Issues in the 'consistency' definition 11
  5.4. Window handling without ICMP 12
6. Summary of Recommendations for Trickle Multicast Forwarding . 12
7. IANA Considerations 12
8. Security Considerations 13
9. Acknowledgments 13
10. References 13
  10.1. Normative References 13
  10.2. Informative References 14
Authors' Addresses 14

1. Introduction

The ROLL working group is chartered to design and standardize a routing protocol for resource constrained devices in Low-power and Lossy Networks (LLN) [RFC6550]. The requirements for ROLL are documented in [RFC5548] [RFC5673] [RFC5826] [RFC5867]. For building control it is recognized that most communication is local to the wireless mesh network, and does not necessarily pass through the edge router. The point-to-point RPL routing algorithm is developed to efficiently support unicast routing in such applications [I-D.ietf-roll-p2p-rpl]. The Trickle algorithm was initially developed to support the RPL routing algorithm [RFC6206], and later proposed to support general multicast delivery in LLNs in Trickle Multicast Forwarding (TMF) [I-D.ietf-roll-trickle-mcast].

This draft discusses the multicast requirements for constrained devices participating in M2M building control networks. An important requirement is the delivery of control commands to a subset (group) of neighbouring devices in the LLN within some latency bound. Also, analyses are provided of how well Trickle algorithm and TMF can meet these requirements and suggestions for improvement are made.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119]. Addtional privileged words are described below.

"TMF" is used as an abbreviation for Trickle Multicast Forwarding as described in [I-D.ietf-roll-trickle-mcast].

A "device" is a physical processor connected to at least one link through a network interface. Each interface has at least one IP unicast address. The IP address is optionally bound to a host name, which may be a Fully Qualified Domain Name (FQDN).

One device communicates directly with another device by wirelessly transmitting packets to it over a link. The link quality is divided in three regions [Zhao]:

1. good: where a transmitted packet will be correctly received by a destination with a probability of say 95% or more.
2. transitional: where the probability of correct reception fluctuates.
3. bad: where almost no transmission is successfully received.

It is empirically known that good links can become bad occasionally

(e.g. once a week for a few minutes)due to dynamic effects such as multipath interference.

A distinction is made between reception and delivery of a message. A message is received when it is stored in the reception buffer of the receiver after transmission and all error checks have been succesfully passed. The message is delivered when the message is passed from the reception buffer to the destination application. We also say the application accepts the message.

Broadcasting is used for the link-local sending of one packet to all reachable 1-hop neigbours. This is equivalent to the term link-local multicast.

1.2. Motivation

In this draft, we focus and develop discussions on requirements pertaining to IP multicasting requirements and IP multicast routing, in the context of building control applications on LLNs. This draft aims to show potential (latency) improvements for current proposed multicast routing approaches, that can be easily attained.

2. Application characteristics

Multicast is important for building control applications. Two types of applications are considered:

1. Discovery messages to (a subset of) the members of the mesh (multicast GET)
2. Control messages to a subset of the mesh (multicast PUT)

The first type requires the message to be sent to a (sub)set which may be randomly distributed over the building area. Some of the destinations return unicast response messages to the source.

The second type requires a Non-Confirmable message mostly to be sent to a closely spaced subset. No return messages are generated. This second type is the subject of this draft, although most of the requirements equally apply to case 1.

GET and PUT and Confirmable/Non-Confirmable are message types defined for CoAP [I-D.ietf-core-coap]. They are thought representative for the two applications types, as the multicast GET SHOULD return a unicast response and the multicast PUT typically does not return a response in control applications.

An office building typically consist of multiple floors, divided in

working areas. The working areas can be open or enclosed by walls. Within a working area sensors measure temperature, presence, humidity and other parameters. On the basis of these measurements, equipment within the working area can receive commands to change settings. A well-known example is presence detection to switch on or dim lights. The equipment configuration is quite stable, because devices are installed in the ceiling, and modifying (or servicing) the installation can be costly.

The equipment is interconnected in a wireless network. The RF transmissions pass through the walls and generate interference to the wireless equipment in other working areas.

The lay-out of a network may be different from installation to installation. However, it is expected that many wireless networks extend over one floor and include several working areas. Another working hypothesis is that most of the time sensors will multicast their values to a group of devices within the working area. Consequently, multicast messages are often meant for a subset of neigbouring (not necessarily 1-hop) devices.

A LoWPAN is a mesh of wireless devices that share the same IPv6 address prefix. A typical LowPAN in a building may cover the area of an entire floor. A commercial installation may cover 1000 m2 per floor. A length of 50 m can easily mean a hop count >5 for a message to pass from end to end. For example, devices may be installed in the ceiling in a grid with a grid pattern distance of 2 m between devices.

Messages may consist of sensor measurements performed or commands issued in a given working area, which then must be acted upon by neigbouring devices in the same working area. Under this control pattern, source and sink are located in one working area, and accordingly sink and source of a multicast message are often between 3 - 6 m from each other. Consequently, it is required to send a multicast to a subset of the devices in the LoWPAN.

In case of commands to luminaries, a command message must be delivered to all LoWPAN-local multicast group members within a clear deadline of about 200ms. In [RFC5867] a deadline of 120 ms is suggested for other building applications.

Although control messages are frequently exchanged between closely spaced (less than 6 m) devices, it is sometimes necessary to send a message to a subset of devices covering the whole building. In that case the multicast message will need to pass the edge router of the LoWPAN and to propagate to other subnets. This case is discussed in more detail in [I-D.ietf-core-groupcomm].

3. Multicast requirements

The multicast requirements are derived from the characteristics of the aforementioned applications. A device is said to be correct it it follows the selected multicast routing algorithm. The application characteristics and the network installation make it possible to add an additional set of network properties to make the multicast algorithm more efficient.

The basic traditional multicast requirements (applicable to both PUT and GET) are [Mullender]:

- o Validity: If sender S sends message, m, to a group, g, of destinations, a path exists between S and any destination D, and if S and D are correct, D eventually accepts m.
- o Integrity: A destination D accepts m at most once from sender S and only if S sent m to a group including D.
- o Agreement: If a correct destination of g accepts m, then all correct destinations of g accept m.

The set of intended destination devices is identified by the multicast (group) IP address. Every device in the associated multicast group is a destination of the multicast. Each destination accepts messages with as destination the specified IP multicast address. Additional multicast requirements are:

- o Timeliness: There is a known constant C such that if m is sent at time t, no correct destination accepts m after t+C.

For lighting control applications the value of C is taken as 200 ms. This requirement only holds for the PUT case without response from a destination, but not for the GET case where a response is returned.

- o Ordering: When m1 and m2 sent to the same group g, and a receiver in g accepts message m1 before m2, every receiver in g accepts m1 before accepting m2

Ordering applies to both the PUT and GET cases. Ordering can be partial or total. Partial ordering means that for specified message pairs, one message of the pair precedes the other. In case of total ordering, every message pair is ordered. Partial ordering is obtained by adding message counters in the message such that destinations can order the messages of a given sender. Messages from different sources are not ordered. Total ordering can be obtained with vector clocks or using synchronized clocks. Vector clocks require a large overhead that increases linearly with the number of devices in the network. As long as no synchronized clocks are available, partial ordering seems the most realistic. Total Ordering

is interesting for the discovery application. When two devices announce themselves simultaneously with conflicting properties, all participants can come to the same decision by favoring the first arrival. Partial ordering is necessary when a multicast message needs multiple packets (for example discovery messages) or when multicast messages are sent with intervals shorter than the maximum throughput delay.

4. Performance of Trickle-based multicast

In this section we investigate the behavior of the Trickle algorithm [RFC6206] when used for multicast routing. Rebroadcasting as defined in Trickle makes meeting tight deadlines a challenge. Simulation results in this section show for particlar configurations and parameter settings which end-to-end communication delays can be expected.

4.1. Reasons for using Trickle

The simplest approach to IP multicast is to broadcast from a source to a set of devices reachable over good links in one hop. This is not sufficient however, because the set of reachable devices is often a subset of the set of destination devices. Consequently, additional measures are needed to make sure that the Agreement requirement is met. A standard technique, to reach all devices instead of a subset, stipulates that every receiver of a broadcast message rebroadcasts this message (flooding). When the multicast destination address of the message corresponds with a specified multicast address in the receiver device, the message is delivered. Thanks to this technique it is assured that when a path exists between the source and the destination device, the destination device will eventually receive the message from the sender.

Given the network density described in section 2, the multicast can generate a broadcast storm with lots of interfering senders. The technique to prevent the storm, also used in Trickle, is to randomly delay a message rebroadcast. However, long delays can seriously jeopardize the Timeliness requirement. The following sections give insight under which conditions the Timeliness requirement can be met.

4.2. Simulation setup

The simulations were done on a general rectangular network topology and on an approximation of known building installations. The IEEE 802.15.4 protocol is simulated with CSMA and the standard back-off intervals specified by IEEE 802.15.4. Packets between A and B arrive with a probability dependent on the distance but independent of the

direction. A distance of 70m is at the limit of the transmission range. Two rectangular meshes were tried: (1) 5 x 5 nodes and (2) 10 x 10 nodes. The distance between two adjoining neigbors was varied between 5 and 70 m. The total surface for the 10 x 10 mesh varied accordingly between 45 x 45 m^2 and 630 x 630 m^2. The building installation approximation consist of a rectangular grid of 14 x 7 nodes over a surface of 35 x 15 m^2. Parameters Imin, Imax and k and variables I, t and c are defined as in [RFC6206].

4.3. Simulation results

The table below presents some of the results on the 5 x 5 mesh.

Imax	k	Parameter	Distance 10m	40m	70m
250ms	1	hopcount	1	2-4	5-9
250ms	1	avg delay	5 ms	40 ms	110 ms
250ms	1	max delay	18 ms	90 ms	1050 ms
250ms	1	msgs sent	0-5	0-11	1-12
250ms	1	msgs received	18-36	3-20	0-20
250ms	3	hopcount	1	2-4	5-9
250ms	3	avg delay	5 ms	40 ms	130 ms
250ms	3	max delay	25	90 ms	260 ms
250ms	3	msgs sent	1-7	3-12	7-13
250ms	3	msgs received	40-60	14-32	9-23
500ms	1	hopcount	1	3-5	5-10
500ms	1	avg delay	5 ms	40 ms	110 ms
500ms	1	max delay	19 ms	100 ms	1500 ms
500ms	1	msgs sent	0-4	0-8	0-10
500ms	1	msgs received	12-26	0-16	0-16
500ms	3	hopcount	1	3-5	5-10
500ms	3	avg delay	5 ms	40 ms	120 ms
500ms	3	max delay	22	80 ms	240 ms
500ms	3	msgs sent	1-8	2-9	5-10
500ms	3	msgs received	28-44	8-27	5-18

The observed behavior is close to what is observed on the 10 x 10 mesh and on the installation configuration. Behavior on, for example, a single row of nodes tends to be quite different and requires quite different parameter settings. The results in the table concern node (4,4) which had the longest end-to-end delays of all nodes. Node (0,0) sent a message every 2 seconds. Individual packets were lost but all messages arrived at all nodes eventually. The Imin was taken to 10 ms and Imax was taken to 250 ms and 500 ms with quite similar results. Changing the Imax has measurable influence on the maximum end-to-end delay. The table shows how many copies of a given message were received by node (4,4) and how many times a given message was rebroadcast. For k=3 more messages were

received and sent. Receiving more messages leads to lower maximum delays because the probability of receiving the message early increases with increasing rebroadcast frequency.

The causes for the large maximum delays (>400ms), occurring at d=70m, have been investigated in more detail. It is shown that a new packet does not always arrive after the first transmission. This is probably due to the synchronization of nodes when a new message arrives, resulting in hidden terminal effects at the destination node by overlapping sending intervals of its neighbors. For d=70 m, packets are only received by the direct neighbor along the x-axis or the y-axis. Consequently, when node (x, y) receives a new message, it originates probably from (x-1, y) or (x, y-1). When node (x, y) sends, packets are received in nodes (x+1, y) and (x,y+1). Given a Imin value of 10ms there is a large probability that the sending by nodes (x+1, y) and (x, y+1) overlap, leading to collision of the messages at node (x+1, y+1). In the following intervals, nodes (x+1, y) and nodes (x, y+1) receive the last message from their neighbors and do not repeat the message because c is larger than k, thus leading to long delays. The receiving node (x+1, y+1) sends at regular intervals, determined by the Imax value, its last received 'old' message. Often the reception of the old message by a neighbor leads to resending the new message. For that reason the maximum delay is linked to the maximum interval Imax. Increasing the value of k increases the probability of reciveing rebroadcast messages.

4.4. Simulation conclusions

The results indicate that for the network configurations we foresee, with Trickle it is quite possible to reach average message delivery latency within the 200 ms range, meeting the Timeliness requirement for most nodes, and to limit the maximum latency by tuning parameter k.

5. Performance issues of Trickle Multicast Forwarding

The Trickle Multicast Forwarding (TMF) draft [I-D.ietf-roll-trickle-mcast] differs from direct application of [RFC6206] in the introduction of multi-source, sliding windows, and use of ICMP messages. For Trickle parameter k finite, a transmission event consists of sending a Trickle ICMP advertizement (that summarizes a forwarder's state i.e. buffered IP multicast packets) and in addition any multicast messages that need rebroadcasting. This section analyzes some issues of TMF, in particular its ability to meet the Timeliness requirement for building control scenarios, and proposes improvements to address the issues.

5.1. Redundancy of Trickle ICMP message

Summarizing state in an ICMP message is clearly useful to reduce network traffic, if many IP multicast packets are being buffered in Trickle multicast forwarders. However, if only one or a few multicast packets are active in the network at a time, a forwarder sending ICMP messages generates unnecessary overhead. As an example, consider a forwarder that stores and needs to rebroadcast a single multicast message m1. According to TMF, it would need to send an ICMP message containing information about m1 (SeedID, sequence number, M bit) and additionally send a Trickle Multicast message with a Trickle Multicast header option which contains exactly the same information (SeedID, sequence number, M bit) plus the useful application data.

In such cases were low latency is required, the extra overhead of sending the ICMP message leads to additional delays, for example in dense network topologies due to increased congestion. In a simulation of a building control installation the operation with and without extra ICMP message was compared for the case that a single multicast message was active. Without ICMP messages an average latency of message delivery to the entire group of 131 ms was observed. The extra overhead generated by ICMP messages led to an average delay of 197 ms, quite close to the Timeliness bound of 200 ms.

The simulation modeled a single IP multicast message active in a 6LoWPAN network, delivery targeted to a group which is a subset of 13 nodes out of 95 nodes total, with a 40-byte data payload, each node acting as a forwarder, with Trickle parameters k=1, Imin=32 ms, Imax=128 ms.

To addresss the latency issue without increasing k (which would lead to increased traffic), we propose that:

- o sending the Trickle ICMP message is made OPTIONAL as part of a transmission event, if a Trickle forwarder has any Trickle Multicast Messages to send in that transmission event. A Trickle Multicast forwarder may decide per transmission event (depending on internal state e.g. number of buffered messages) whether the ICMP message is sent or not.
- o as part of a transmission event, sending the Trickle ICMP message MAY be done after retransmitting Trickle Multicast Messages. Note that the TMF draft does not clearly express a preferred order for Trickle ICMP messages.

These proposed changes are still fully compatible with existing implementations of TMF.

5.2. Ability to configure forwarders as data sinks

The current TMF makes a separation between (IP) hosts and Trickle Multicast forwarders. Nodes that only need to receive IP multicast packets (not wanting to participate in rebroadcasting) therefore can be configured as hosts unaware of the multicast routing protocol. However, this bears the risk that such hosts receive a specific multicast packet very late or never, because they don't have a way to signal missing packets to Trickle forwarders. Implementing a node as a host has the clear advantages that the node does not need to buffer any Trickle Multicast Messages which can considerably reduce memory usage.

A solution that enables the best of both worlds is to allow Trickle Multicast forwarders to act as 'data sinks' only i.e. not acting as a repeater. We propose that:

- o a Trickle Multicast forwarder MAY act as a data sink, which means that it does keep sliding window state for messages it accepts, and sends Trickle ICMP messages, but does not buffer any Trickle Multicast Messages for retransmission.

5.3. Issues in the 'consistency' definition

In the TMF draft the notion of 'consistency' (as we read it) is based on information received in Trickle ICMP messages only, not on information received from incoming Trickle Multicast Messages. This operation can lead to unnecessary delays in certain use cases. Consider the following scenario:

- o Nodes A, B, C are Trickle Multicast forwarders; where A cannot hear C and C cannot hear A
- o A stores messages m1,m2,m3, B stores m1,m2,m3, C stores m2,m3
- o C sends ICMP(m2,m3)
- o B sees an inconsistency based on this and schedules the missing m1 for transmission
- o A sends ICMP (m1,m2,m3) but not any multicast message m_i
- o B sees a consistency and increments c
- o When the Trickle timer at B expires assuming k=1, the scheduled transmission of m1 is cancelled
- o C does not get m1 from B, at least not during this round.

Eventually C will get m2, after more rounds (when B transmits before A does), but later than necessary.

A first approach to improve latency in this scenario is to apply the suppression only to ICMP messages, not to scheduled multicast messages (such as m1 by B in the example above). A refinement of this approach is to maintain a counter c for each SeedID/

Sequence-number combination, in addition to a global Trickle counter c. Then, retransmission of Trickle Multicast Messages is only suppressed for those messages that have been received at least k times. ICMP suppression is still based on the global Trickle counter c as in the current TMF draft.

5.4. Window handling without ICMP

A forwarder that does not support sending ICMP advertizements could advertize its state by retransmitting the multicasat message with the largest number in its window that has no missing messages relative to the lower bound of the window. So if a forwarder has a window containing m1,m2,m4,m5 it retransmits m2, triggering others to send m3 (and maybe higher numbers). If it encounters an inconsistency, i.e. seeing a multicast with a number lower than its own upperbound, it itself would send out the messages that have a higher number than the received multicast message (excluding the ones that it has received at least k tines during the current Trickle interval).

6. Summary of Recommendations for Trickle Multicast Forwarding

From the analyses above emerge a number of recommendations that aim to reduce transmission latency of multicast messages and to reduce the probability of missing a multicast message. In summary, the following adaptations to TMF [I-D.ietf-roll-trickle-mcast] are proposed which can be applied independently of each other:

1. Efficient retransmission: sending the Trickle ICMP message is made OPTIONAL as part of a transmission event if a Trickle forwarder already has any Trickle Multicast Messages to send.
2. Allow data sinks: a Trickle Multicast forwarder MAY refrain from buffering any Trickle Multicast Messages for retransmission.
3. Consistency improvement: When a transmission is suppressed, a forwarder MAY only suppress ICMP but not suppress transmission of a multicast message that was scheduled due to a detected inconsistency. This approach could be refined by keeping in addition to a global Trickle consistency counter c, separate counters c per SeedID/sequence-number combination suppressing only messages seen at least k times.
4. Window handling without ICMP: forwarders without ICMP sending capability can ask for retransmissions by rebroadcasting multicast messages

7. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

8. Security Considerations

TBD

9. Acknowledgments

This I-D has benefited from conversations with and comments from Anders Brandt, Kerry Lynn, Zach Shelby, Emmanuel Frimout, Michael Verschoor, Jamie Mc Cormack, Dee Denteneer, Jerald Martocci, Matthieu Vial, and Nicolas Riou.

10. References

10.1. Normative References

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

[RFC5548] Dohler, M., Watteyne, T., Winter, T., and D. Barthel, "Routing Requirements for Urban Low-Power and Lossy Networks", RFC 5548, May 2009.

[RFC5673] Pister, K., Thubert, P., Dwars, S., and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks", RFC 5673, October 2009.

[RFC5826] Brandt, A., Buron, J., and G. Porcu, "Home Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5826, April 2010.

[RFC5867] Martocci, J., De Mil, P., Riou, N., and W. Vermeylen, "Building Automation Routing Requirements in Low-Power and Lossy Networks", RFC 5867, June 2010.

[RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko, "The Trickle Algorithm", RFC 6206, March 2011.

[RFC6550] Winter, T., Thubert, P., 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.

10.2. Informative References

[I-D.ietf-roll-p2p-rpl]
Goyal, M., Baccelli, E., Philipp, M., Brandt, A., and J. Martocci, "Reactive Discovery of Point-to-Point Routes in Low Power and Lossy Networks", draft-ietf-roll-p2p-rpl-13 (work in progress), June 2012.

[I-D.ietf-roll-trickle-mcast]
Hui, J. and R. Kelsey, "Multicast Forwarding Using Trickle", draft-ietf-roll-trickle-mcast-00 (work in progress), April 2011.

[I-D.ietf-core-coap]
Shelby, Z., Hartke, K., Bormann, C., and B. Frank, "Constrained Application Protocol (CoAP)", draft-ietf-core-coap-10 (work in progress), June 2012.

[I-D.ietf-core-groupcomm]
Rahman, A. and E. Dijk, "Group Communication for CoAP", draft-ietf-core-groupcomm-02 (work in progress), July 2012.

[Zhao] Zhao, J. and R. Govindan, "Understanding Packet Delivery Performance in Dense Wireless Sensor Networks", senSys , 2003.

[Mullender]
Mullender, S., "Distributed Systems, Second Edition", Section 5 , Addison-Wesley Publishing Company, Inc. , ISBN 0-201-62427-3, 1995.

Authors' Addresses

Peter van der Stok (editor)
vanderstok consultancy
Kamperfoelie 8
Helmond, 5708 DM
The Netherlands

Email: consultancy@vanderstok.org

Esko Dijk
Philips Research
High Tech Campus 34-1
Eindhoven, 5656 AA
The Netherlands

Email: esko.dijk@philips.com

Armand Lelkens
Philips Research
High Tech Campus 34-1
Eindhoven, 5656 AA
The Netherlands

Email: armand.lelkens@philips.com