

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 21, 2007

K. Kim, Ed.
picosNet Corp/Ajou Univ.
S. Daniel Park, Ed.
SAMSUNG Electronics
G. Montenegro
Microsoft Corporation
S. Yoo
Ajou University
N. Kushalnagar
Intel Corp
June 19, 2007

6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD)
[draft-daniel-6lowpan-load-adhoc-routing-03.txt](#)

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on December 21, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD) is intended for use by IEEE 802.15.4 devices in a 6LoWPAN. It is a simplified on-demand routing protocol based on AODV.

Table of Contents

1.	Introduction	3
2.	Requirements notation	3
3.	Overview	3
4.	Terminology	5
5.	Data Structures	6
5.1	Routing Table Entry	6
5.2	Route Request Table Entry	6
5.3	Message Format	7
5.3.1	Route Request (RREQ)	7
5.3.2	Route Reply (RREP)	8
5.3.3	Route Error (RERR)	10
6.	Operation	11
6.1	Generating Route Request	11
6.2	Processing and Forwarding Route Request	11
6.3	Generating Route Reply	11
6.4	Receiving and Forwarding Route Reply	12
6.5	Local Repair and RERR	12
7.	Configuration Parameters	13
8.	IANA Consideration	13
9.	Security Considerations	13
10.	Acknowledgments	13
11.	References	14
11.1	Normative Reference	14
11.2	Informative Reference	14
	Authors' Addresses	16
	Intellectual Property and Copyright Statements	17

1. Introduction

The IEEE 802.15.4 standard [[ieee802.15.4](#)] targets low power personal area networks. The "IPv6 over IEEE 802.15.4" document [I-D.montenegro-lowpan-ipv6-over-802.15.4] defines basic functionality required to carry IPv6 packets over IEEE 802.15.4 networks (including an adaptation layer, header compression, etc). Likewise, the functionality required for packet delivery in IEEE 802.15.4 meshes is defined, as mesh topologies are expected to be common in LowPAN networks. However, neither the IEEE 802.15.4 standard nor the "IPv6 over IEEE 802.15.4" specification provide any information as to how such a mesh topology could be obtained and maintained.

The 6LowPAN Ad hoc Routing Protocol (LOAD) is a simplified on-demand routing protocol based on AODV[RFC3561] for 6LowPAN. Besides the main AODV specification [[RFC3561](#)], several efforts aim at simplifications of the protocol, as in the AODVjr proposal [[AODVjr](#)] or the TinyAODV implementation [[TinyAODV](#)]. Similarly, DyMO allows for minimalist implementation leaving non-essential functionality as optional [[I-D.ietf-manet-dymo](#)]. LOAD enables multihop routing between IEEE 802.15.4 devices to establish and maintain routing routes in 6LowPAN.

This document defines the message formats, the data structures, and the operations of LOAD.

2. Requirements notation

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 [[RFC2119](#)].

3. Overview

This section describes the distinctive features of LOAD compared to AODV. LOAD is defined to be operating on top of the adaptation layer instead of the transport layer. That is, it creates a mesh network topology underneath and unbeknownst to IPv6 network layer. IPv6 sees a 6LowPAN as a single link. This is similar to how other technologies regularly create complex structures underneath IP (e.g., ethernet spanning tree bridges, token ring source routing, ATM, etc). LOAD control packets use the encapsulation defined in [I-D.montenegro-lowpan-ipv6-over-802.15.4]. All LOAD control packets shall use the prot_type value TBD (suggested value of 4).

LOAD assumes the use of either one of the two different addresses for routing: the EUI-64 address and the 16 bit short address of the 6LowPAN device.

LOAD makes use of broadcast in its route discovery. It does so in order to propagate the Route Request (RREQ) messages. In this specification, such broadcast packets are obtained by setting the PAN id to the broadcast PAN (0xffff) and by setting the destination address to the broadcast short address (0xffff).

LOAD doesn't use the destination sequence number in order to reduce the size of the control messages and simplify the route discovery process. For ensuring loop freedom, only the destination of a route SHOULD generate a RREP in reply. The intermediate nodes SHOULD not respond with a RREP. By the same reason, LOAD does not use the "Gratuitous RREP".

LOAD MAY use the local repair for a link break during a data delivery. In a local repair, only the destination generates a RREP in reply because of no use of the destination sequence number.

If a local repair fails, LOAD MAY generate a Route Error (RERR) message toward the originator of the data delivery to notify that the destination is no longer reachable by way of the broken link. The format of RERR is simplified to include only one unreachable destination while the RERR of AODV MAY include multiple ones.

LOAD does not use the "precursor list" of AODV to simplify the routing table structure. Notice that AODV uses the precursors for forwarding RERR messages in the event of detection of the loss of the next hop link. In LOAD, RERR is forwarded only to the originator of the failed data delivery, thus no requiring to use the precursor list.

LOAD MAY use the route cost, which is the accumulated link cost from the originator to the destination, as a metric of routing. For this, LOAD utilizes the Link Quality Indicator (LQI) of the 6LoWPAN PHY layer in the routing decision in addition to the hop distance. There are many ways to include LQI in the routing metric. The approach taken by LOAD avoids a route which contains weak links whose LQI is below certain threshold value (i.e. WEAK_LQI_VALUE).

LOAD SHOULD utilize the acknowledged transmission option at the 6LoWPAN MAC layer for keeping track of the connectivity of a route. LOAD uses neither the passive acknowledgements nor the HELLO messages of AODV.

The basic operations of LOAD are route discovery, managing data structures and maintaining local connections. For these operations, LOAD maintains the following two tables: the routing table and the route request table. The routing table stores route information such as destination, next hop node, and status. The route request table stores the temporary route information used in the route discovery process.

There are two different types of 6LoWPAN devices: the reduced function device(RFD) and the full function device (FFD). LOAD SHOULD utilize only FFD for mesh routing. Thus, A FFD SHOULD implement the operations of LOAD and maintain the data structures of LOAD.

4. Terminology

This section defines the terminology of LOAD that is not defined in [\[RFC3753\]](#) and [\[RFC3561\]](#).

destination

A node to which data packets are to be transmitted. Same as "destination node".

forward route

A route set up to send data packets from the originator to its destination.

link cost

The link Quality (LQ) between a node and its neighbor node.

link quality indicator (LQI)

A mechanism to measure the Link Quality (LQ) in IEEE 802.15.4 PHY layer [\[ieee802.15.4\]](#). It measures LQ by receiving the signal energy level. A high LQ value implies the good quality of communication (i.e. low link cost).

weak link

A link of which the LQI falls below WEAK_LQI_VALUE.

originator

A node that initiates a route discovery process. Same as "originating node"

route cost

An accumulated link cost as a LOAD control message (RREQ or RREP) passes through the nodes on the route.

reverse route

A route set up to forward a RREP back to the originator from the destination. Same as "reverse route" in [[RFC3561](#)].

5. Data Structures

A FFD in 6LoWPAN SHOULD maintain a routing table and a route request table. This section describes the tables and the message formats.

5.1 Routing Table Entry

The routing table of LOAD includes the following fields:

destination address

The 16 bit short or EUI-64 link layer address of the final destination of a route

next hop address

The 16 bit short or EUI-64 link layer addresses of the next hop node to the destination.

status

The status of a route. It includes the following states: VALID, INVALID, ROUTE_DISCOVERY, etc.

life time

The valid time in milliseconds before the expiration or the deletion of a route.

5.2 Route Request Table Entry

Route request table is used for discovering routes. It stores the following route request information until a route is discovered.

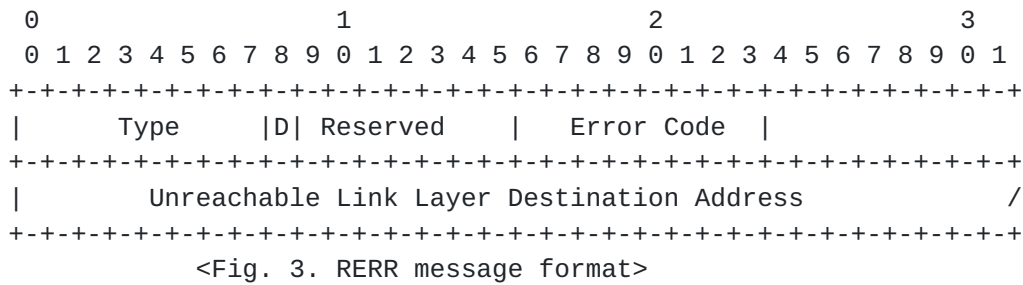
route request ID

a sequence number uniquely identifying the particular RREQ when taken in conjunction with the originator

The RREP message format is shown in Fig. 2 and contains the following fields:

Type	2 for indicating a RREP message.
CT	Type of route cost. The followings are the current route cost types known: 0 : Hop count while avoiding weak links 1-0xf: TBD
WL	The total number of weak links on the routing path from the originator of the RREP to the sender of the RREP.
R	1 Local Repair.
D	1 for the 16 bit address of the destination, 0 for the EUI-64 address of the destination.
O	1 for the 16 bit address of the originator, 0 for the EUI-64 address of the originator.
Reserved	0; ignored on reception.
RC(Route cost)	The accumulated link cost of the route from the originator of the RREP to the sender of the RREP. The type of link cost is specified by CT.
RREQ ID	A sequence number uniquely identifying the particular RREQ when taken in conjunction with the originator.
Link layer Destination Address	The 16 bit short or EUI-64 link layer address of the destination for which a route is supplied.
Link layer Originator Address	The 16 bit short or EUI-64 link layer address of the node which originated the Route Request.

5.3.3 Route Error (RERR)



The RERR message format is shown in Fig. 3 and contains the following fields:

Type	3 for indicating a RERR message.
D	1 for the 16 bit address of the destination, 0 for the EUI-64 address of the destination.
Reserved	0; ignored on reception.
Error Code	Numeric value for describing error. 0x00 = No available route 0x01 = Low battery 0x02 = routing cost not supported 0x03 - 0xff = reserved (TBD)
Unreachable Link Layer Destination Address	The 16 bit short or EUI-64 link layer address of the final destination that has become unreachable due to a link break.

6. Operation

6.1 Generating Route Request

The basic operations of LOAD include route discovery, managing data structures and maintaining local connections. A node maintains the following two tables for routing: the routing table and the routing request table.

During the discovery period, an originator, a node that requests a route discovery, generates a Route Request (RREQ) message with the RREQ ID which was incremented by one from the previous RREQ ID value.

A node SHOULD NOT originate more than RREQ_RATELIMIT RREQs per second. After brocasting a RREQ, a node waits for a RREP. If a route is not discovered within NET_TRAVERSAL_TIME milliseconds, the node MAY try again the discovery process a maximum of RREQ_RETRIES times.

6.2 Processing and Forwarding Route Request

Upon receiving a RREQ, an intermediate FFD node tries to find the entry of the same originator address and RREQ ID pair in the route request table. If the entry is found, the node just discards the RREQ. Otherwise, the node creates a reverse route to the originator in the routing table and a RREQ entry in the route request table. It then checks whether the link through which the RREQ is received is a weak link or not. If the link is a weak link, the node adds 1 to the WL field of the RREQ. Then, the node forwards the RREQ.

6.3 Generating Route Reply

When the destination receives a RREQ, it tries to find the entry of the same originator address and RREQ ID pair in the route request table. If the entry is found, the destination compares the route cost of the RREQ with the forward route cost of the entry. If the cost of the RREQ is better than (i.e. less than) that of the entry, the destination updates the reverse route to the originator in the routing table and generates a RREP in reply. If the cost of the RREQ is not less than that of the entry, the destination just discards the RREQ.

If the CT field of the RREQ is 0 (i.e. hop count while avoiding weak links), the route cost becomes a tuple of (WL, RC) and is ordered lexicographically. That is, the route cost (WL,RC) is said to be better (or smaller) than or equal to (WL',RC') if the following condition holds.

$(WL, RC) \leq (WL', RC')$ if and only if $WL < WL'$, or $WL == WL'$ and $RC \leq RC'$

6.4 Receiving and Forwarding Route Reply

Upon receiving a RREP, an intermediate node checks whether the link through which the RREP is received is a weak link or not. If the link is a weak link, the node add 1 to the WL field of the RREP.

The node then checks whether it has a route entry for the destination of the RREP (i.e. the originator of the corresponding RREQ). If it does not have the route entry, it just discards the RREP. Otherwise, it also checks for the existence of the corresponding RREQ entry (which has the same RREQ ID and originator address pair as that of the RREP) in the route request table. If there is no such entry, then it just discards the RREP.

If there is such an entry and the entry has worse reverse route cost (i.e. higher value) than the route cost of the RREP, the node updates the entry with the information of the RREP and forwards it to the previous hop node toward the destination of the RREP. If the entry has better reverse route cost (i.e. lower value) than that of the RREP, the node just discards the RREP.

If the CT field of the RREP is 0 (i.e. hop count while avoiding weak links), the route cost becomes a tuple of (WL,RC) and is ordered lexicographically.

During the delivery of the RREP to the originator, the route cost value of the RREP is accumulated on the reverse route from the destination to the originator.

6.5 Local Repair and Route Error (RERR) Messages

If a link break occurs or a device fails during the delivery of data packets, the upstream node of the link break MAY repair the route locally. To repair a route, the node disseminates a RREQ with the originator address set to its own address and the destination address set to the data packet's destination address. In this case, the 'R flag' of the RREQ is set to 1. The data packet is buffered during the route discovery period. If the destination node receives the RREQ for a route repair, it responds with a RREP of which the 'R flag' is also set to 1.

If the repairing node cannot receive a RREP from the final destination until the end of the route discovery period, it unicasts a RERR with an error code that indicates the reason of the repair failure to the originator. A repairing node SHOULD NOT generate more than RERR_RATELIMIT RERRs per second. Then, the buffered data packet is discarded. If the originator that sends a data packet receives the RERR, it MAY try to reinitiate route discovery.

When the repairing node receives a RREP from the destination during the route discovery period, it updates the routing table entry information from the RREP. Then the node transmits the buffered data packet to the destination through the new route.

7. Configuration Parameters

This section describes the default values for some important parameters associated with LOAD operations.

Parameter Name	Value
-----	-----
NET_TRAVERSAL_TIME	TBD
RREQ_RETRIES	3
RREQ_RATELIMIT	2
RERR_RATELIMIT	2
WEAK_LQI_VALUE	8

8. IANA Consideration

This document needs an additional IANA registry for the prot_type value that indicates the LOAD format.

9. Security Considerations

The security considerations of the [[RFC3561](#)] are applicable to this document. As described in the charter of the 6lowpan w.g., LOAD will also try to reuse existing security considerations related to Ad hoc routing protocols. Further considerations will be studied in the next version.

10. Acknowledgments

Thanks to the authors of [RFC 3753](#) and [RFC 3561](#), as parts of this document are patterned after theirs. Thanks to Nandakishore Kushalnagar, Byeong-Hee Roh, Myung-ho Jung, Dae-hong Son, and Minhoo Lee for their useful discussions and supports for writing this document.

11. References

11.1 Normative Reference

- [EUI64] "GUIDELINES FOR 64-BIT GLOBAL IDENTIFIER (EUI-64) REGISTRATION AUTHORITY", IEEE <http://standards.ieee.org/regauth/oui/tutorials/EUI64.html>.
- [I-D.ietf-6lowpan-format]
N., Kushalnagar., Montenegro, G., Hui, J., and D. Culler,
"6LoWPAN: Transmission of IPv6 Packets over IEEE 802.15.4 Networks", [draft-ietf-6lowpan-format](#) (work in progress), April 2007.
- [I-D.ietf-ipv6-rfc2462bis]
Thomson, S., "IPv6 Stateless Address Autoconfiguration", [draft-ietf-ipv6-rfc2462bis-08](#) (work in progress), May 2005.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [BCP 26](#), [RFC 2434](#), October 1998.
- [ieee802.15.4]
IEEE Compute Society, "IEEE Std. 802.15.4-2003", IEEE Std. 802.15.4-2003, October 2003.

11.2 Informative Reference

- [RFC1884] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", [RFC 1884](#), December 1995.
- [I-D.ietf-6lowpan-problem]
N., Kushalnagar., Montenegro, G., and C. Schumacher,
"6LoWPAN: Overview, Assumptions, Problem Statement and Goals", [draft-ietf-6lowpan-problem](#) (work in progress), February 2007.
- [RFC3561] Perkins, C., Belding-Royer, E., and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", [RFC 3561](#), July 2003.
- [RFC3753] Manner, J. and M. Kojo, "Mobility Related Terminology", [RFC 3753](#), June 2004.

[AODVjr] Chakeres, Ian and Klein-Berndt, Luke, "AODVjr, AODV Simplified", ACM SIGMOBILE Mobile Computing and Communications Review pp. 100-101, July 2002.

[I-D.ietf-manet-dymo]
Chakeres, I., "Dynamic MANET On-demand (DYMO) Routing",
[draft-ietf-manet-dymo](#) (work in progress), May 2007.

[TinyAODV]
"TinyAODV Implementation", TinyOS Source Code Repository h
[ttp://cvs.sourceforge.net/viewcvs.py/tinyos/tinyos-1.x/](http://cvs.sourceforge.net/viewcvs.py/tinyos/tinyos-1.x/contrib/hsn/)
[contrib/hsn/](http://cvs.sourceforge.net/viewcvs.py/tinyos/tinyos-1.x/contrib/hsn/).

Authors' Addresses

Kim, Ki Hyung (editor)
picosNet Corp/Ajou Univ.
San 5 Wonchun-dong, Yeongtong-gu
Suwon-si, Gyeonggi-do 442-749
KOREA

Phone: +82 31 219 2433
EMail: kkim86@picosnet.com

Soohong Daniel Park, Editor
Mobile Platform Laboratory, SAMSUNG Electronics
416 Maetan-3dong, Yeongtong-gu
Suwon-si, Gyeonggi-do 442-742
KOREA

Phone: +82 31 200 4508
Email: soohong.park@samsung.com

Gabriel Montenegro
Microsoft Corporation
US

Email: gabriel_montenegro_2000@yahoo.com

Seung Wha Yoo
Ajou University
San 5 Wonchun-dong, Yeongtong-gu
Suwon-si, Gyeonggi-do 442-749
KOREA

Phone: +82 31 219 1603
Email: swyoo@ajou.ac.kr

Nandakishore Kushalnagar
Intel Corp
US

Email: nandakishore.kushalnagar@intel.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

