6LoWPAN P. Thubert
Internet-Draft Cisco
Intended status: Standards Track March 18, 2008
Expires: September 19, 2008

LoWPAN simple fragment Recovery
draft-thubert-6lowpan-simple-fragment-recovery-00

Status of this Memo

Copyright Notice

Abstract

Considering that 6LoWPAN packets can be as large as 2K bytes and that an 802.15.4 frame with security will carry in the order of 80 bytes of effective payload, a packet might end up fragmented into as many as 25 fragments at the 6LoWPAN shim layer. If a single one of those fragments is lost in transmission, all fragments must be resent, further contributing to the congestion that might have caused the initial packet loss. This draft introduces a simple protocol to recover individual fragments between 6LoWPAN endpoints.

Table of Contents

1. Introduction

Considering that 6LoWPAN packets can be as large as 2K bytes and that a 802.15.4 frame with security will carry in the order of 80 bytes of effective payload, a packet might be fragmented into about 25 fragments at the 6LoWPAN shim layer. This level of fragmentation is much higher than that traditionally experienced over the Internet with IPv4 fragments. At the same time, the use of radios increases the probability of transmission loss and Mesh-Under techniques compound that risk over multiple hops.

Past experience with fragmentation has shown that missassociated or lost fragments can lead to poor network behaviour and, eventually, trouble at application layer. The reader might start his research from [I-D.mathis-frag-harmful] and follow the references. That experience led to the definition of the Path MTU discovery [RFC1191] protocol that avoids fragmentation over the Internet.

An end-to-end fragment recovery mechanism might be a good complement to a hop-by-hop MAC level recovery with a limited number of retries. This draft introduces a simple protocol to recover individual fragments between 6LoWPAN endpoints. Specifically in the case of UDP, valuable additional information can be found in UDP Usage Guidelines for Application Designers [I-D.ietf-tsvwg-udp-guidelines].

2. 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 [RFC2119].

Readers are expected to be familiar with all the terms and concepts that are discussed in "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals" [RFC4919] and "Transmission of IPv6 Packets over IEEE 802.15.4

Networks" [RFC4944].

ERP

Error Recovery Procedure.

LoWPAN endpoints

The LoWPAN nodes in charge of generating or expanding a 6LoWPAN header from/to a full IPv6 packet. The LoWPAN endpoints are the points where fragmentation and reassembly take place.

3. Rationale

There are a number of usages for large packets in Wireless Sensor Networks. Such usages may not be the most typical or represent the largest amount of traffic over the LoWPAN; however, the associated functionality can be critical enough to justify extra care for ensuring effective transport of large packets across the LoWPAN.

The list of those usages includes:

Towards the LoWPAN node:

Packages of Commands: A number of commands or a full configuration can by packaged as a single message to ensure consistency and enable atomic execution or complete roll back. Until such commands are fully received and interpreted, the intended operation will not take effect.

Firmware update: For example, a new version of the LoWPAN node software is downloaded from a system manager over unicast or multicast services. Such a reflashing operation typically involves updating a large number of similar 6LoWPAN nodes over a relatively short period of time.

From the LoWPAN node:

Waveform captures: A number of consecutive samples are measured at a high rate for a short time and then transferred from a sensor to a gateway or an edge server as a single large report.

Large data packets: Rich data types might require more than one fragment.

Uncontrolled firmware download or waveform upload can easily result in a massive increase of the traffic and saturate the network.

When a fragment is lost in transmission, all fragments are resent, further contributing to the congestion that caused the initial loss, and potentially leading to congestion collapse.

This saturation may lead to excessive radio interference, or random early discard (leaky bucket) in relaying nodes. Additional queueing and memory congestion may result while waiting for a low power next hop to emerge from its sleeping state.

4. Requirements

This paper proposes a method to recover individual fragments between LoWPAN endpoints. The method is designed to fit the following requirements of a LoWPAN (with or without a Mesh-Under routing protocol):

Number of fragments

The recovery mechanism must support highly fragmented packets, with a maximum of 32 fragments per packet.

Minimimum acknowledgement overhead

Because the radio is half duplex, and because of silent time spent in the various medium access mechanisms, an acknowledgement consumes roughly as many resources as data fragment.

The recovery mechanism should be able to acknowledge multiple fragments in a single message.

Controlled latency

The recovery mechanism must succeed or give up within the time boundary imposed by the recovery process of the Upper Layer Protocols.

Support for out-of-order fragment delivery

A Mesh-Under load balancing mechanism such as the ISA100 Data Link Layer can introduce out-of-sequence packets. The recovery mechanism must account for packets that appear lost but are actually only delayed over a different path.

Optional flow control

The aggregation of multiple concurrent flows may lead to the saturation of the radio network and congestion collapse.

The recovery mechanism should provide means for controlling the number of fragments in transit over the LoWPAN.

Backward compatibility

A node that implements this draft should be able to communicate with a node that implements [RFC4944]. This draft assumes that compatibility information about the remote LoWPAN endpoint is obtained by external means.

5. Overview

The fragmentation/reassembly of a packet must complete within an acceptable overall latency, otherwise the packet expires and must be dropped. This latency must be smaller than Upper Layer Protocol retry values, and smaller than expiration period of the information transported.

The sender transfers a controlled number of fragments (possibly all of them) and flags the last fragment of a series with an Acknowledgement request.

The sender sets a retry timer for the fragment that carries the Acknowledgement request. That fragment is retransmitted individually upon time out. This is repeated until an Acknowledgement comes back

or the packet expires.

Upon receipt of an Acknowledgement request, the receiver responds with an Acknowledgement containing a bitmap that indicates which fragments were actually received. The bitmap is a 32bit DWORD, which accommodates up to 32 fragments and is sufficient for the 6LoWPAN MTU. For all n in [0..31], bit n is set to 1 in the bitmap to indicate that fragment n was received, otherwise the bit is set to 0.

If any fragment immmediately preceding the acknowledgement request is missing, the receiver MAY intentionally delay its response to allow in-transit fragments to arrive.

The sender has either one or no Acknowledgement pending. An Acknowledgement that is not expected or does not acknowledge the pending sequence in the bitmap is a duplicate and is ignored.

When a valid Acknowledgement is received, the sender resumes sending fragments and the process is repeated until all fragments are acknowledged or the packet expires.

Fragments are sent in a round robin fashion: the sender sends all the fragments for a first time before it retries any lost fragment; lost fragments are retried in sequence, oldest first. This mechanism enables the receiver to acknowledge fragments that were delayed in the network before they are actually retried.

It is up to the sender to decide how many fragments are (re)sent before an acknowledgement is received, and the sender can adapt that number to the network conditions. This way, the number of outstanding fragments can be used as a flow control mechanism to protect the network.

6. New Dispatch types and headers

This specification extends "Transmission of IPv6 Packets over IEEE 802.15.4 Networks" [RFC4944] with 3 new dispatch types, for Recoverable Fragments (RFRAG) headers with or without Acknowledgement Request, and for the Acknowledgement back.

```
  Pattern    Header Type
+------------+------------------------------------------------+
| 11  101000 | RFRAG      - Recoverable Fragment              |
| 11  101001 | RFRAG-AR   - RFRAG with Acknowledgement Req    |
| 11  101010 | RFRAG-ACK  - RFRAG Acknowledgement             |
+------------+------------------------------------------------+
```

Figure 1: Additional Dispatch Value Bit Patterns

In the following sections, the semantics of "datagram_tag," "datagram_offset" and "datagram_size" and the reassembly process are unchanged from [RFC4944] Section 5.3. "Fragmentation Type and Header."

6.1. Recoverable Fragment Dispatch type and Header

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 1 1 0 1 0 0 X|datagram_offset|         datagram_tag          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Sequence |    datagram_size    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     X set == Ack Requested
```

Figure 2: Recoverable Fragment Dispatch type and Header

X bit

When set, the sender requires an Acknowledgement from the receiver

Sequence

The sequence number of the fragment. Fragments are numbered [0..N] where N is in [0..31].

6.2. Fragment Acknowledgement Dispatch type and Header

```
                         1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
                    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                    |1 1 1 0 1 0 1 0|         datagram_tag          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |           Acknowledgement Bitmap                              |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     ^                   ^
     |                   |        bitmap indicating whether
     |                   +-----Fragment with sequence 10 was received
     +-------------------------Fragment with sequence 00 was received
```

Figure 3: Fragment Acknowledgement Dispatch type and Header

Acknowledgement Bitmap

Each bit in the Bitmap refers to a particular fragment: bit n set indicates that fragment with sequence n was received, for n in [0..31].

All zeroes means that the fragment was dropped because it corresponds to an obsolete datagram_tag. This happens if the packet was already reassembled and passed to the network upper layer, or the packet expired and was dropped.

7. Security Considerations

The process of recovering fragments does not appear to create any opening for new threat.

8. IANA Considerations

Need extensions for formats defined in "Transmission of IPv6 Packets over IEEE 802.15.4 Networks" [RFC4944].

9. Acknowledgments

The author wishes to thank Jay Werb, Christos Polyzois, Soumitri Kolavennu and Harry Courtice for their contribution and review.

10. References

10.1. Normative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.

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

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

10.2. Informative References

[I-D.ietf-tsvwg-udp-guidelines] Eggert, L. and G. Fairhurst, "UDP Usage Guidelines for Application Designers", draft-ietf-tsvwg-udp-guidelines-05 (work in progress), February 2008.

[I-D.mathis-frag-harmful] Mathis, M., "Fragmentation Considered Very Harmful", draft-mathis-frag-harmful-00 (work in progress), July 2004.

[RFC4919] Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals", RFC 4919, August 2007.

Author's Address

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

Phone: +33 4 97 23 26 34
Email: pthubert@cisco.com

ietf-ipr@ietf.org.

Acknowledgment

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