6LoWPAN
Simple Fragment Recovery

(draft-thubert-6lo-forwarding-fragments-00)

Pascal Thubert/ Jonathan Hui

6LoWPAN WG Meeting
88th IETF Meeting
Vancouver

What's new

- Fragment forwarding
  - Using the datagram tag as a switchable label
  - Acks are used to clean intermediate states
  - ECN echo restored
- Removed Compressed ack bitmap
- Needed for 6TiSCH when operating on 15.4 2006 PHY
- Quite stable draft

Need for fragment recovery

- Considering
  - that 6LoWPAN packets can be as large as 1280 bytes
  - that Source routing requires space for routing headers
  - that a 802.15.4 frame with security will carry in the order of 80 bytes of effective payload,

=> An IPv6 packet might be fragmented into > 16 fragments at the 6LoWPAN shim layer.

- This level of fragmentation is much higher than that traditionally experienced over the Internet with IPv4 fragments, already known as harmful.
- At the same time, the use of radios increases the probability of transmission loss but retry only 1 hop
- Mesh-Under and fragment routing techniques compound that risk over multiple hops with no ack

Other problems related to frags

- Hop by Hop recomposition
  - Should be avoided: latency and memory hit
- Multipath
  - Forwarding fragments over multipath multiplies the impact of an anomaly
- Recovery buffers Lifetime
  - Terminating device with limited capacity may have trouble maintaining buffers. How long?
  - Intermediate routers congestion

Fragment Recovery proposal

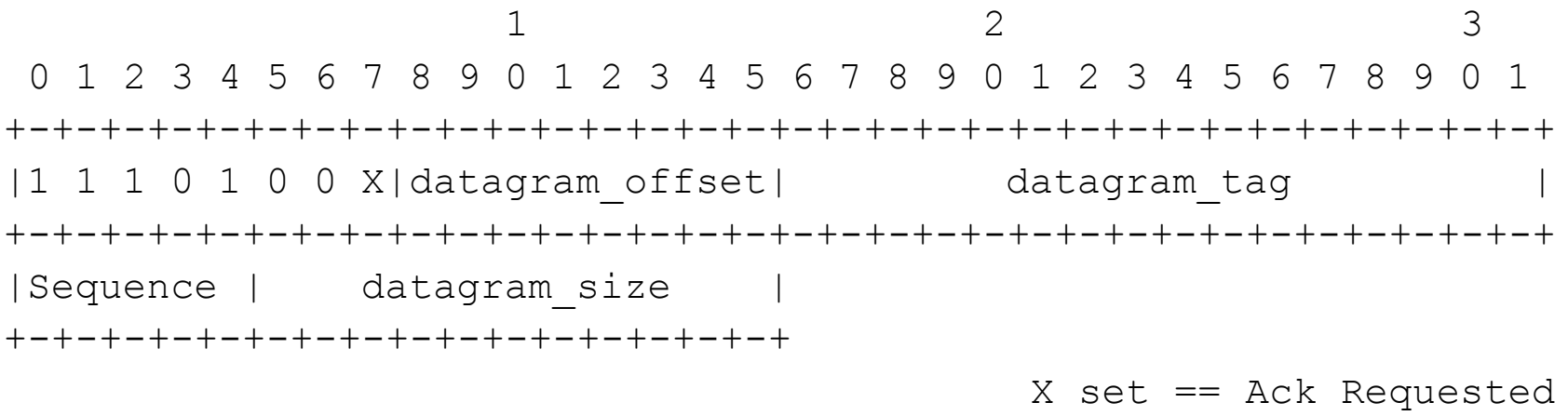
- 32 bits SAck Bitmap
- Variable window size for congestion control
- Round Robin for multipath
- 4 new dispatch types

Pattern	Header Type	
11 101000	RFRAG	- Recoverable Fragment
11 101001	RFRAG-AR	- RFRAG with Ack Request
11 101010	RFRAG-ACK	- RFRAG Acknowledgment
11 101011	RFRAG-AEC	- RFRAG Ack with ECN Echo

Fragment Forwarding proposal

- Frags & Acks have a datagram tag
- Unique for the source if the tag
- Proposal uses the datagram tag as a label
- First fragment sets up a bidir label path
- Final ack & errors clean it up
- Next fragments are label swapped along the same path

Recoverable Fragment Dispatch type and Header



X (check) bit

When set, the sender requires an Acknowledgement from the receiver

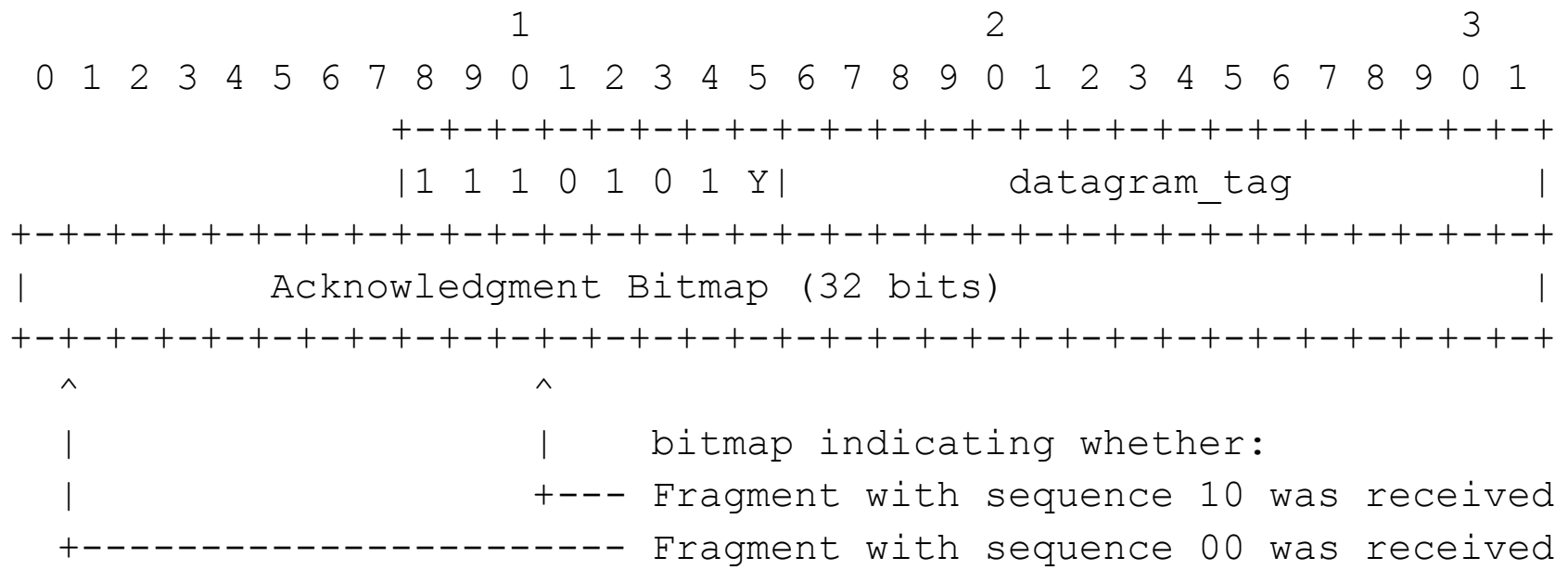
Sequence

The sequence number of the fragment.

Fragments are numbered [0..N] where N is in [0..31].

Fragment Acknowledgement Dispatch type and Header

The ack now has ECN echo:



Y: 1 bit; Explicit Congestion Notification (ECN) signalling

ECN use

- Indicate Congestion in the LoWPAN
  - End to End effect on Transport
  - Potential use at ISA100.11a and 6TiSCH
  - Local Effect on Fragment flow control
- Early detection
  - Avoid Wasteful discard of packets
  - Conditions equivalent to RED
  - Setting ECN is out of scope (just echo)

Explicit Congestion Notification

- ECN in IPv6: Traffic Class bits 6-7

Binary	Keyword	References
00	Not-ECT (Not ECN-Capable Transport)	[RFC 3168]
01	ECT(1) (ECN-Capable Transport(1))	[RFC 3168]
10	ECT(0) (ECN-Capable Transport(0))	[RFC 3168]
11	CE (Congestion Experienced)	[RFC 3168]

  - Not compressed separately by 4944
  - Isolated in RFC 6282 section **3.1.1.** Base Format

- ECN Echo
  - Not an IP function (usually transport)
  - Thus provided by this draft between fragmentation endpoints

????? Questions ?????