LPWAN WG

WG Chairs:
Alexander Pelov <a@ackl.io>
Pascal Thubert <pthubert@cisco.com>

AD: Suresh Krishnan
<suresh@kaloom.com>

Note Well

Any submission to the IETF intended by the Contributor for publication as all or part of an IETF Internet-Draft or RFC and any statement made within the context of an IETF activity is considered an "IETF Contribution". Such statements include oral statements in IETF sessions, as well as written and electronic communications made at any time or place, which are addressed to:

- The IETF plenary session
- The IESG, or any member thereof on behalf of the IESG
- Any IETF mailing list, including the IETF list itself, any working group or design team list, or any other list functioning under IETF auspices
- Any IETF working group or portion thereof
- Any Birds of a Feather (BOF) session
- The IAB or any member thereof on behalf of the IAB
- The RFC Editor or the Internet-Drafts function

All IETF Contributions are subject to the rules of RFC 5378 and RFC 8179.

Statements made outside of an IETF session, mailing list or other function, that are clearly not intended to be input to an IETF activity, group or function, are not IETF Contributions in the context of this notice. Please consult RFC 5378 and RFC 8179 for details.

A participant in any IETF activity is deemed to accept all IETF rules of process, as documented in Best Current Practices RFCs and IESG Statements.

A participant in any IETF activity acknowledges that written, audio and video records of meetings may be made and may be available to the public.

I E T F

Reminder:

Minutes are taken *
This meeting is recorded **
Presence is logged ***

* Scribe; please contribute online to the minutes at: https://etherpad.tools.ietf.org/p/lpwan
** Recordings and Minutes are public and may be subject to discovery in the event of litigation.
*** From the Webex login

Agenda bashing

17:05   Opening, agenda bashing (Chairs)   5mn
- Note-Well, Scribes, Agenda Bashing, Approval minutes from last meeting
- Review todo
- Status of drafts

17:10   SCHC MIC selection   25mn

17:35   SCHC update and discussion: WGLC ready or not ready   20mn

17:55   AOB   5mn

Last meeting Action items

- ~~Submit rev for SCHC draft~~
- Launch WGLG

What is the best MIC ?

CRC size

Why CRC ?
- Allow the suppression of UDP checksum with something more robust

- 16:
  - Same size as UDP checksum
  - Robust enough ?

- 32:
  - Double the size of UDP checksum
  - Robust

Methodology

- Create a packet:
  - Random values
  - Repeating value
  - uniform
- Compute packet CRC
- Generate all the possible packet loss configuration
  - Compute CRC
  - Compare with packet CRC
    - If equal then failure
- Exponential complexity, stop at 18 frag window

No ACK : CRC 16 – random packets

Max frag = 1
0 200 100 0b1 losses = 0 frag = 1
Max frag = 1
0 200 200 0b1 losses = 0 frag = 1
Max frag = 3
0 200 500 0b111 losses = 0 frag = 3
Max frag = 5
0 200 1000 0b11111 losses = 0 frag = 5
Max frag = 1
0 170 100 0b1 losses = 0 frag = 1
Max frag = 2
0 170 200 0b11 losses = 0 frag = 2
Max frag = 3
0 170 500 0b111 losses = 0 frag = 3
Max frag = 6
0 170 1000 0b111111 losses = 0 frag = 6
Max frag = 1
0 140 100 0b1 losses = 0 frag = 1
Max frag = 2
0 140 200 0b11 losses = 0 frag = 2
Max frag = 4
0 140 500 0b1111 losses = 0 frag = 4

Max frag = 8
0 140 1000 0b11111111 losses = 0 frag = 8
Max frag = 1
0 110 100 0b1 losses = 0 frag = 1
Max frag = 2
0 110 200 0b11 losses = 0 frag = 2
Max frag = 5
0 110 500 0b11111 losses = 0 frag = 5
Max frag = 10
0 110 1000 0b1111111111 losses = 0 frag = 10
Max frag = 2
0 80 100 0b11 losses = 0 frag = 2
Max frag = 3
0 80 200 0b111 losses = 0 frag = 3
Max frag = 7
0 80 500 0b1111111 losses = 0 frag = 7
Max frag = 13
0 80 1000 0b1111111111111 losses = 0 frag = 13
Max frag = 2
0 50 100 0b11 losses = 0 frag = 2
Max frag = 4
0 50 200 0b1111 losses = 0 frag = 4

Max frag = 10
0 50 500 0b1111111111 losses = 0 frag = 10
Max frag = 20
0 50 1000 0b10000100110 losses = 16 frag = 20
1 50 1000 0b100110110001001100 losses = 12 frag = 20
2 50 1000 0b11001011010101010111 losses = 9 frag = 20
3 50 1000 0b1010001111111100000 losses = 10 frag = 20
4 50 1000 0b1010101101000010011 losses = 11 frag = 20
5 50 1000 0b1011001100110110010 losses = 10 frag = 20
6 50 1000 0b1100111000101001000 losses = 12 frag = 20
7 50 1000 0b111001010100110011 losses = 9 frag = 20
8 50 1000 0b1111101001101000010 losses = 10 frag = 20
9 50 1000 0b100000101101111001010 losses = 11 frag = 20
10 50 1000 0b10000110011100010100 losses = 12 frag = 20
11 50 1000 0b1001111001010101010101 losses = 9 frag = 20
12 50 1000 0b10110011010101111001 losses = 8 frag = 20
13 50 1000 0b10111110100001101011 losses = 8 frag = 20
14 50 1000 0b1100010111001001011011 losses = 9 frag = 20
15 50 1000 0b11010010010111111000 losses = 9 frag = 20

2^{20} combinations = 1048576

confusion probability: 0,00002

1 is with only 5 losses.

Different CRC

Number of frame
with the sender
MIC
50 attempts

frag#	a8f4	0xd405	0x8408	md5[0,1]	
5	1	1	1	1	
6	1	1	1	1	
7	1	1	1	1	
8	1	1	1,020408163	1	
9	1,02	1	1	1,020408163	
10	1,02	1,020408163	1,081632653	1	
11	1,02	1	1,020408163	1	
12	1,02	1,142857143	1,081632653	1,040816327	
13	1,16	1,183673469	1,265306122	1,06122449	
14	1,24	1,183673469	1,265306122	1,244897959	
15	1,56	1,367346939	1,448979592	1,612244898	
16	1,82	1,795918367	2,040816327	2	
17	3,02	3,040816327	2,795918367	3,12244898	
18	4,693877551	4,795918367	5,020408163	4,632653061	

Interim, December 19th, 2017

Conclusions

- CRC16 is not protecting correctly for window higher that 8 fragments
- CRC do not react better than as any other mechanism
- MIC = Packet Length
  - But do not protect the information inside,
  - cannot be used to compress the UDP checksum
- CRC 32 is perfect (CRC32c):
  - But 2 bytes are lost
- No exhaustive study for more that 22 fragments
  - Realistic if Ack modes (last window)

Conclusion

Default behavior (LT personal choice):

- - Mandate Length and recommend to not compress UDP checksum
- Mandate CRC32c
- Mandate a CRC regarding the window size
- + Always CRC16

This will not block the standard, this is just a default value that can be over written by SCHC-over-foo

draft-ietf-lpwan-ipv6-static-context-hc-08

Authors:
Ana Minaburo <ana@ackl.io>
Laurent Toutain <laurent.toutain@imt-atlantique.fr>
Carles Gomez <carlesgo@entel.upc.edu>

Carles' modifications ... (1)

3. Terminology

5. Fragmentation
   5.1. Overview
⇄ 5.2. Functionalities
⇄ 5.3. Delivery Reliability options
   5.4. Fragmentation Frames Formats
     5.4.1. Fragment format
     5.4.2. Fragmentation formats
     5.4.3. ACK format
     5.4.4. All-1 and All-0 formats
     5.4.5. Abort formats
   5.5. Baseline mechanism
     5.5.1. No ACK
     5.5.2. The Window modes
     5.5.3. Bitmap Optimization
   5.6. Supporting multiple window sizes
   5.7. Downlink fragment transmission
  6. Padding management

Input the terms used in all the draft, ex: All-0, W, Dev, Dw, etc

Introduction to the Fragmentation

Explains each functions of the fragmentation process, ex: MIC, Retransmission Timer, FCN, Attempts, Bitmap, etc

Introduce the 3 fragmentation modes.

Explains in details the different fragment format frames used for all the fragmentation

Explains in detail each delivery reliability option: No ACK, ACK-Always, ACK-on-error. And Explains in detail the Bitmap optimization

Carles' modifications … (2)

- Abort Frames:

```
<------ byte boundary ------><---1 byte --->
+--- ... ---+- ... -+-+-...-+-+-+-+-+-+-+-+-+
| Rule ID | DTag |W| FCN |     FF     | (no MIC & no payload)
+--- ... ---+- ... -+-+-...-+-+-+-+-+-+-+-+-+
Figure 16: All-1 Abort format
```

```
<------ byte boundary -----><---1 byte --->
+----... --+-... -+-+-+-+-+-+-+-+-+-+-+-+-+
| Rule ID | DTag |W| 1..1|     FF     |
+----... --+-... -+-+-+-+-+-+-+-+-+-+-+-+-+
Figure 17: ACK Abort
```

Is this a confirmation intended to confirm that an Abort message has been received?

No, All-1 Abort is used for the sender and ACK abort is used for the receiver in order to trigger an Abort in the fragmentation transmission. See the FSM in the index C

Carles' modifications … (3)

LPWAN

- Some minor modifications in the Baseline description, to be done before the last call. To clarify the ACK-Always and ACK-on-error descriptions.
- Some Editorial suggestion (minor) to be made in section 3
- Minor Editorial corrections in section 5

This modifications will not take much more time.

Do we go for WG LAST CALL?

THANKS

Questions?

AOB ?