Transmission of SCHC-compressed packets over IEEE 802.15.4 networks

`draft-ietf-6lo-schc-15dot4-02`

Carles Gomez
Universitat Politècnica de Catalunya (UPC)
carles.gomez@upc.edu

Ana Minaburo
Consultant
anaminaburo@gmail.com

IETF 117 – San Francisco, July 2023

Introduction

This document

```
+------------+
| CoAP, other|
+------------+
| UDP, other |
+------------+
|    IPv6    |
+------------+
| 6LoWPAN HC |
+------------+
|6LoWPAN Frag|
+------------+
|  802.15.4  |
+------------+
```

Traditional

```
+------------+
| CoAP, other|
+------------+
| UDP, other |
+------------+
|    IPv6    |
+------------+
|  SCHC HC   |  <-- NEW
+------------+
|6LoWPAN Frag|
+------------+
|  802.15.4  |
+------------+
```

Main goal

```
+------------+
|    CoAP    |
+------------+
|     UDP    |
+------------+
|   SCHC HC  |  <-- NEW
+------------+
|    IPv6    |
+------------+
| 6LoWPAN HC |
+------------+
|6LoWPAN Frag|
+------------+
|  802.15.4  |
+------------+
```

Transition

SCHC exploits a priori knowledge of header field values

Status

- WG adoption
  - draft-ietf-6lo-schc-15dot4-00
    - Same content as draft-gomez-6lo-schc-15dot4-05
  - In January 2023

- Version -02
  - Several additions and updates
  - One coauthor, now a contributor
    - Flavien Moullec (had joined in -01)

Table of Contents

Main technical updates

Main new additions

4.3. Pointer-based, R.O. frame format

- SCHC Pointer Dispatch

```
<------------------ IEEE 802.15.4 frame payload ------------------->

                                 <----- SCHC Packet ----->
+-----------------+--------------+-------------+---------+ - - - - +
| SCHC Ptr Dsptch | SCHC Pointer | SCHC Header | Payload | Padding |
+-----------------+--------------+-------------+---------+ - - - - +
                       v                <->
                       |                 |
                       +-----------------+
                compr. dest. addr. residue
```

NEW

- Additional 6LoWPAN Dispatch Type (page 0):
  - Bit pattern 01000101 (to be confirmed by IANA)
  - Indicate that this dispatch is followed by the SCHC Pointer
    - Allows all RuleIDs (starting by 1 or by 0) after the SCHC Dispatch

4.3. Pointer-based, R.O. frame format

- SCHC Pointer format:
  - OLD:

```
 0 1 2 3 4 5 6 0 1 2 3 4 5 6
+-+-----------+-+-----------+
|P|  Bit      |0|  Address  |
| |  pointer  | |  length   |
+-+-----------+-+-----------+
```

  - NEW:

```
 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
+-+-------------+-+-------------+
|R|    Bit      |0|   Address   |
| |   pointer   | |   length    |
+-+-------------+-+-------------+
```

    – Byte-aligned
    – Allows to represent a full 128-bit IPv6 destination address (if needed)

Appendix A. Header compr. examples (I/II)

- A.1. Single-hop or Straightforward Route-Over
  - IPv6/UDP uplink packet
- A.2. Tunneled, RPL-based Route-Over
  - TO-DO
- A.3. Pointer-based Route-Over
  - IPv6/UDP uplink packet
- A.4. Mesh-Under
  - TO-DO
- A.5. Enabling the transition protocol stack
  - IPv6/UDP/CoAP uplink packet

Appendix A. Header compr. examples (II/II)

- A.1. Single-hop or Straightforward Route-Over

Uncompressed IPv6/UDP packet:

```
60 00 00 00 00 17 00 40    FD 00 00 00 00 00 00 00
02 02 00 02 00 02 00 02    20 01 00 00 00 00 00 00
00 00 00 00 00 00 00 01    22 3D 16 2E 00 0F 33 68
68 65 6C 6C 6F 20 31
```

55 bytes

Compression (RuleID 0x20)

FID	FL	FP	DI	TV	MO	CDA	Sent [bits]
IPv6 Version	4	1	Bi	6	ignore	not-sent	
IPv6 Diffserv	8	1	Bi	0	equal	not-sent	
IPv6 Flow Label	20	1	Bi	0	equal	not-sent	
IPv6 Length	16	1	Bi		ignore	compute-*	
IPv6 Next Header	8	1	Bi	17	equal	not-sent	
IPv6 Hop Limit	8	1	Bi	64	ignore	not-sent	
IPv6 DevPrefix	64	1	Bi	FD00::/64	equal	not-sent	
IPv6 DevIID	64	1	Bi		ignore	value-sent	64
IPv6 AppPrefix	64	1	Bi	2001::/64	equal	not-sent	
IPv6 AppIID	64	1	Bi	::1	equal	not-sent	
UDP DevPort	16	1	Bi	8765	equal	not-sent	
UDP AppPort	16	1	Bi	5678	equal	not-sent	
UDP Length	16	1	Bi		ignore	compute-*	
UDP checksum	16	1	Bi		ignore	compute-*	

SCHC Dispatch

SCHC-compressed packet:

```
44 20 02 02 00 02 00 02
00 02 68 65 6C 6C 6F 20
31
```

17 bytes

Appendix B. Analysis of RO multihop approaches (I/III)

- Straightforward RO approach:
  - Header overhead: 1 byte
  - All nodes (incl. intermediate nodes) must store all the Rules in use in the whole network
  - Suitable for rather small and static networks

- Tunneled, RPL-based RO approach:
  - Header overhead: 2 bytes + variable part
    - Variable part: ≥ 6 bytes (uplink); 12 bytes, 16 bytes… (downlink)
  - A node only stores the Rules for the communications it is involved in as an endpoint
    - Reduces memory requirements and the impact of context updates (if any)
  - Scalable with network size
  - Requires RPL non-storing mode
  - Intranetwork communication requires traversing the root node (might not be necessarily optimal)

Appendix B. Analysis of RO multihop approaches (II/III)

- Pointer-based RO approach:
  - Header overhead: 3 bytes + variable part
  - Variable part is the IPv6 destination address compression residue:
    - Could be 0 bytes in special cases (full address known beforehand)
    - Could be 2-8 bytes in intranetwork communications (prefix known)
    - Could be 16 bytes in communications with external nodes (if several possible destination prefixes)
  - A node only stores the Rules for the communications it is involved in as an endpoint
    - Reduces memory requirements and the impact of context updates (if any)
  - Scalable with network size
  - Does not require RPL
  - Intranetwork communication: not constrained to traversing a root node

Appendix B. Analysis of RO multihop approaches (III/III)

- Best fit:
  - Small networks
    - Straightforward
  - Larger networks
    - Tunneled, RPL-based
      - Communication with (several) external networks
    - Pointer-based
      - Intranetwork communication + special cases of external comm.

Question 1

- **Keep or reduce the number of multihop RO approaches?**
  - Currently, 3 existing approaches
- Authors' opinion:
  - Enable all of them:
    - Relatively complementary
    - The most suitable one can be chosen for each deployment
- Thoughts?

Question 2

- **IEEE 802.15.4-specific document or generic document?**
  - Currently, IEEE 802.15.4-specific
- Authors' opinion:
  - IEEE 802.15.4-specific is more straightforward
    - Well defined scope
- Observations:
  - IEEE 802.15.4-specific doc may be the basis for other similar documents focusing on other similar technologies
  - If generic approach, then one technology-specific document is needed for each technology of interest
    - In addition to the base document
- Thoughts?

Comments/Questions?

draft-ietf-6lo-schc-15dot4-02

Carles Gomez
Universitat Politècnica de Catalunya (UPC)
carles.gomez@upc.edu

Ana Minaburo
Consultant
anaminaburo@gmail.com

Introduction

Assumptions:
- Best case, global addr.
- CoAP
  a) No header options
  b) Table 6, RFC 8824

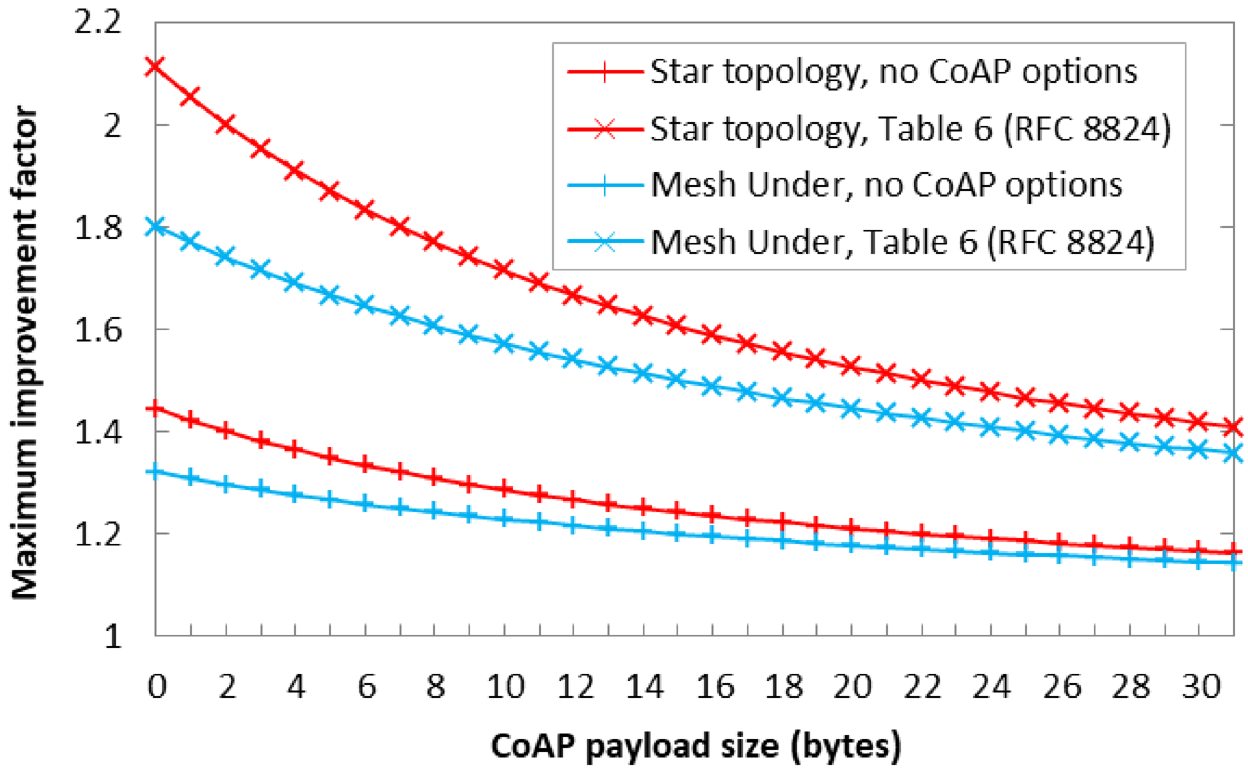
- IPv6/UDP/CoAP header size

	IPv6/UDP (bytes)	**CoAP (bytes)**		**TOTAL (bytes)**	
		a)	b)	a)	b)
No compression	48	4	16	52	64
6Lo(WPAN) - RFC 6282	7	4	16	11	23
SCHC - RFC 8724, 8824	1	1	1	2	2

  - SCHC: static context, a priori knowledge of header field values
- Theoretical battery lifetime improvement over IEEE 802.15.4 by a factor up to >2
  - Actual improvement will be lower, depending on device HW, MAC/adaptation/application layer settings, payload size, network topology, etc.

Introduction (II)

- Maximum battery lifetime improvement factor
  - Short MAC addresses, intra-PAN
  - E.g. a battery-operated sensor that periodically sends a message over IEEE 802.15.4



NOTE: actual improvement will be lower