
Static Context Header Compression
[sjiek]
Where do we want to go
in RIOT?
RIOT SUMMIT 2020 - BART MOONS

WHERE ARE WE GOING TO?

"an entirely new reality in which **our interface to the Internet** will no longer be predominantly a screen, but rather the **objects of the cyber-physical system** embodied by the Internet of Things."

TOWARDS A WEB OF THINGS

Connecting any *thing* to provide an interface to the real world.

Web
with JSON
over HTTP
using OAuth

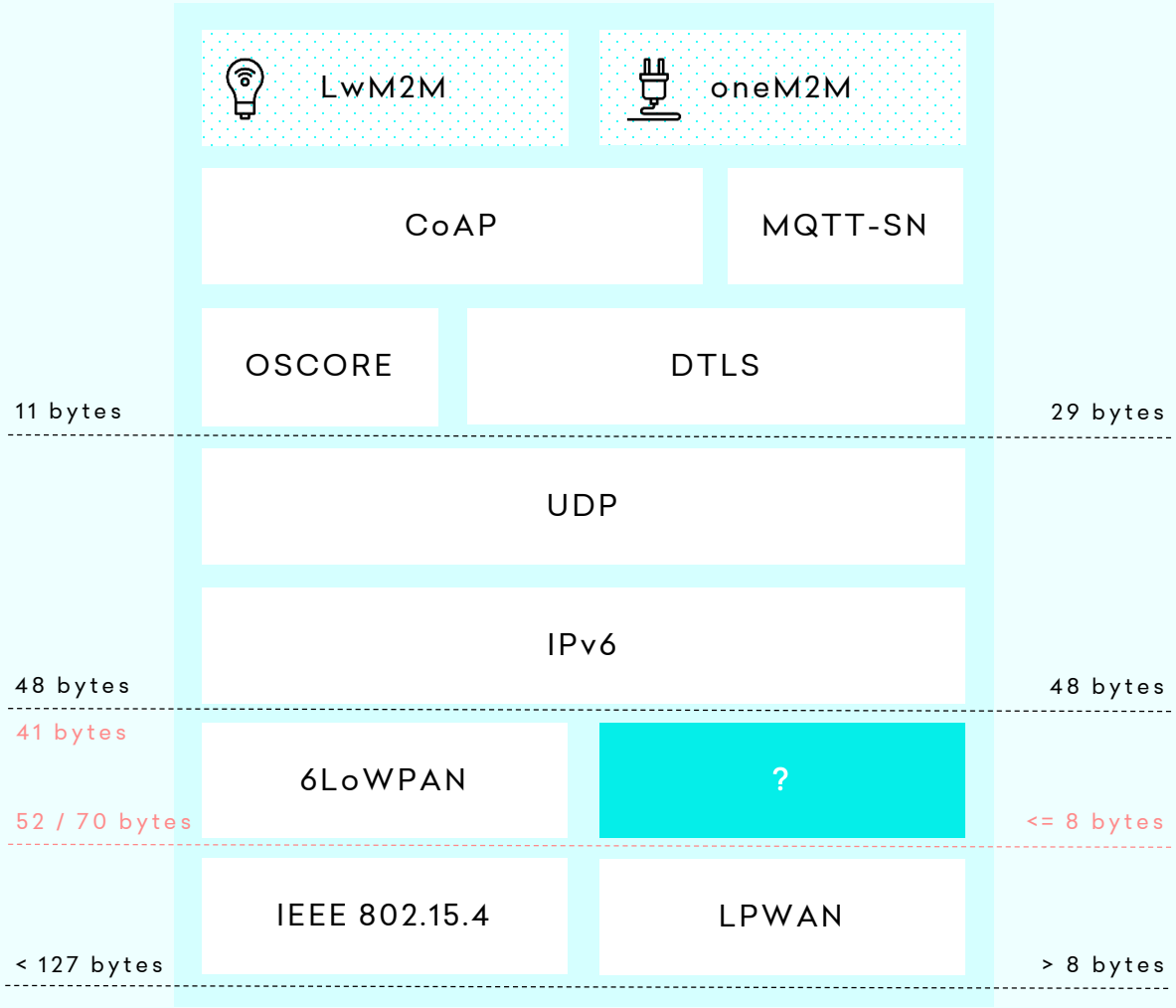
oneM2M
with JSON
over MQTT
using TLS

LwM2M
with SenML
over CoAP
using DTLS

Web
with CBOR
over CoAP
using OSCORE

WHAT DEFINES A THING?

1. Services and semantics
   - What do I measure?
   - What is on/off?
2. Networking
   - How do I send a packet?
3. Communication
   - What frequency do I use?



1) SHORT RANGE AND 2) LONG RANGE IoT STACKS

LOW POWER WIDE AREA NETWORKS

Short range technologies do not suffice.

Sigfox
8 - 12 bytes

< 15 km
0.1 kbps

LoRa
51 - 242 bytes

< 15 km
0.3 – 5.5 kbps

NB-IoT
125 bytes

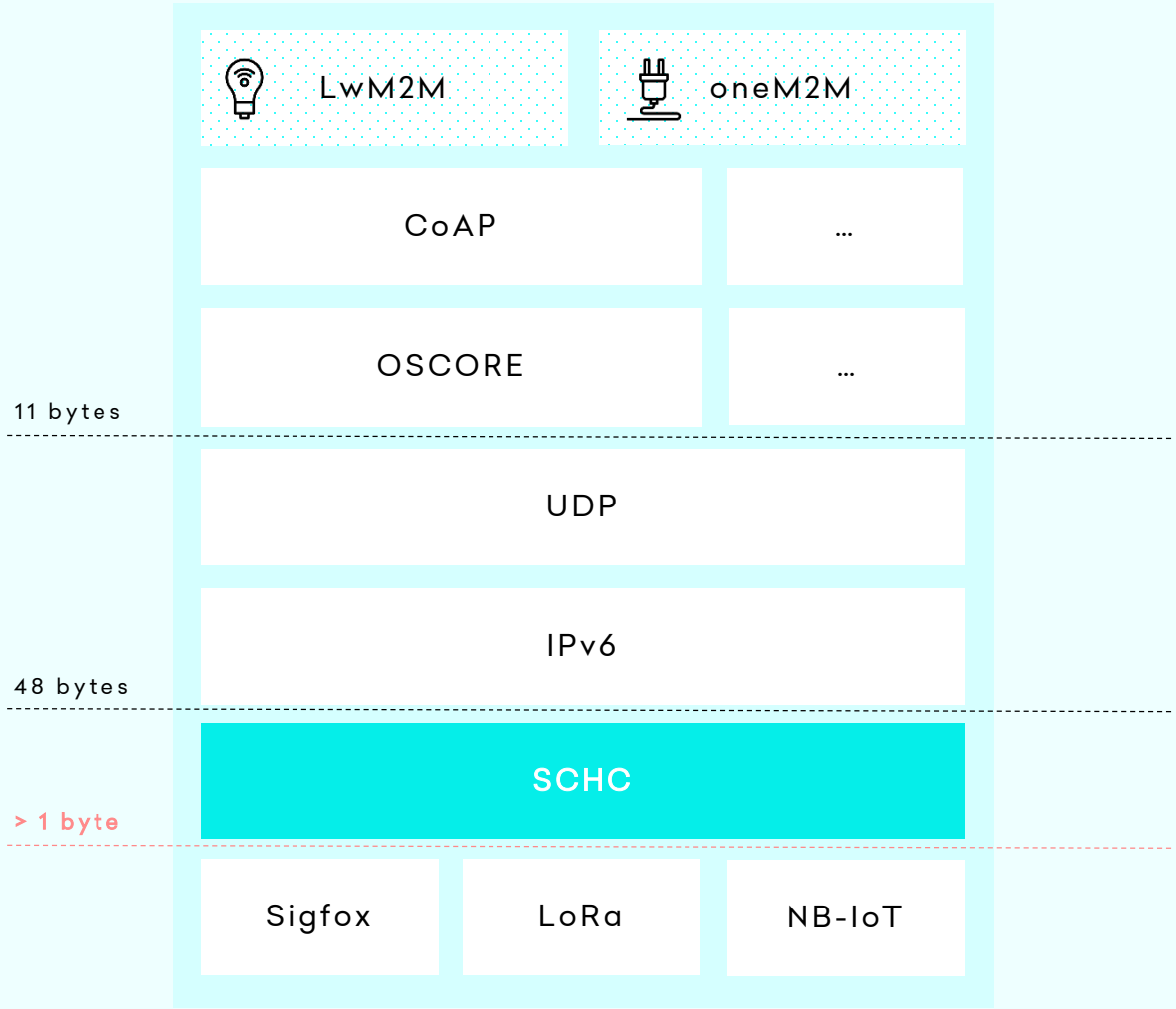
< 10 km
100 kbps

Weightless
> 10 bytes

< 5 km
< 10 Mbps

STATIC CONTEXT HEADER COMPRESSION

- Generic Framework for Header compression in LPWANs: **RFC 8724**



LONG RANGE IoT STACK

STATIC CONTEXT HEADER COMPRESSION

Static Context Header Compression employs the fact that

- UDP/IP flows remain static in LPWANs
- some fields can be calculated from L2

These static flows are stored on both sides

- the id of the flow is sent to the other side

SCHC CONTEXT

The list of flows contains a Target Value (**TV**) for every header field

- Can be matched using the Matching Operator (**MO**)
- Can be (de)compressed using the Compression/Decompression Action (**CDA**)

Send the rule id and possible residue

FIELD	FL	DI	TV	MO	CDA	**ID 1**
src port	16	BI	5683	&equal	not-sent	

SCHC CONTEXT FIELDS

STATIC CONTEXT HEADER COMPRESSION RESIDUE

Matching Operator provides flexibility to match a header field with a rule field:

- `equal` matches the header field with the target value
- `match-map` sends the index of the matched target value
- `MSB` sends the last bits of the original header field

Field	FL	DI	TV	MO	CDA	**ID 1**
Src port	16	BI	5683	&MSB(4)	LSB(12)	
Dst port	16	BI	5683	&equal	not-sent	
Length	16	BI		&ignore	comp-length	
Checksum	16	BI		&ignore	comp-chk	

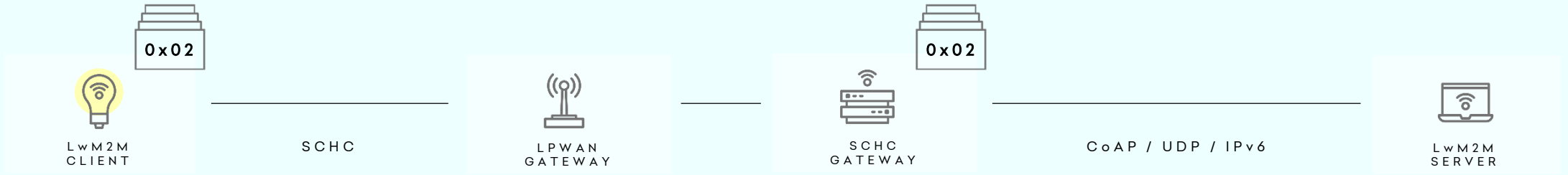
Table 1: UDP RULE 1

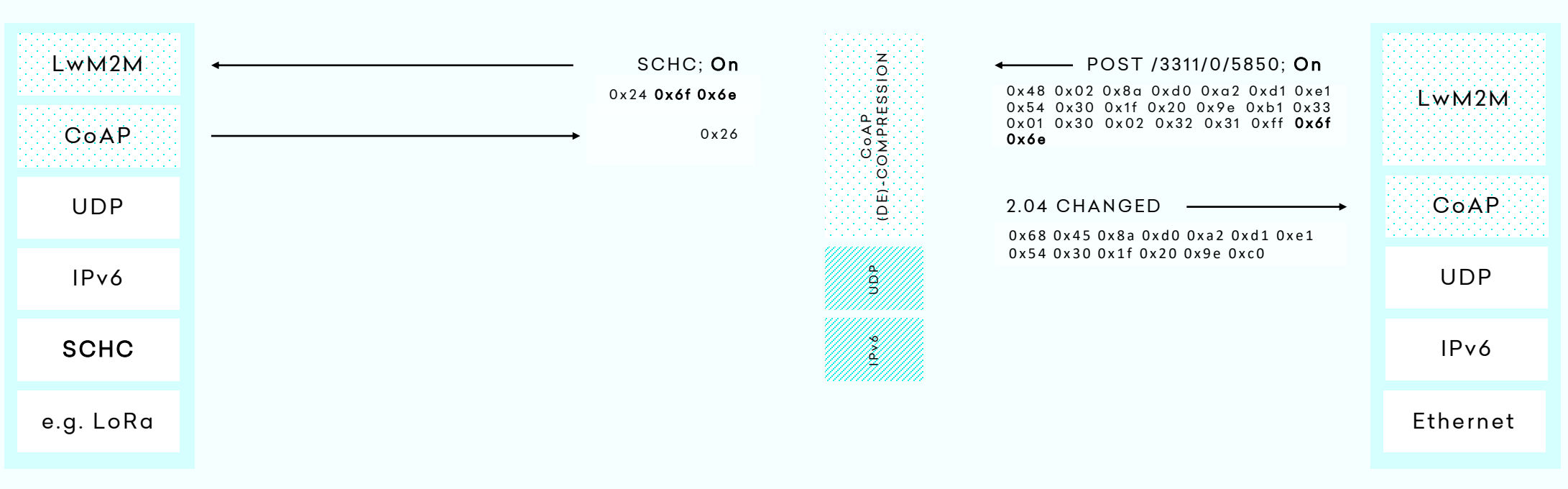
Field	FL	DI	TV	MO	CDA	**ID 2**
Version	2	BI	1	&equal	not-sent	
Type	2	DW	CON	&equal	not-sent	
Type	2	BI	[ACK, RST]	&match-map	mapping-sent	
TKL	4	BI	0x00	&equal	not-sent	
Code	8	UP	GET	&equal	not-sent	
Code	8	DW	CONTENT	&equal	not-sent	
MID	16	BI	0x0000	MSB(4)	LSB(12)	
Uri-Path	88	DW	temperature	&equal	not-sent	

Table 2: CoAP RULE 1

COMPRESSION EXAMPLE

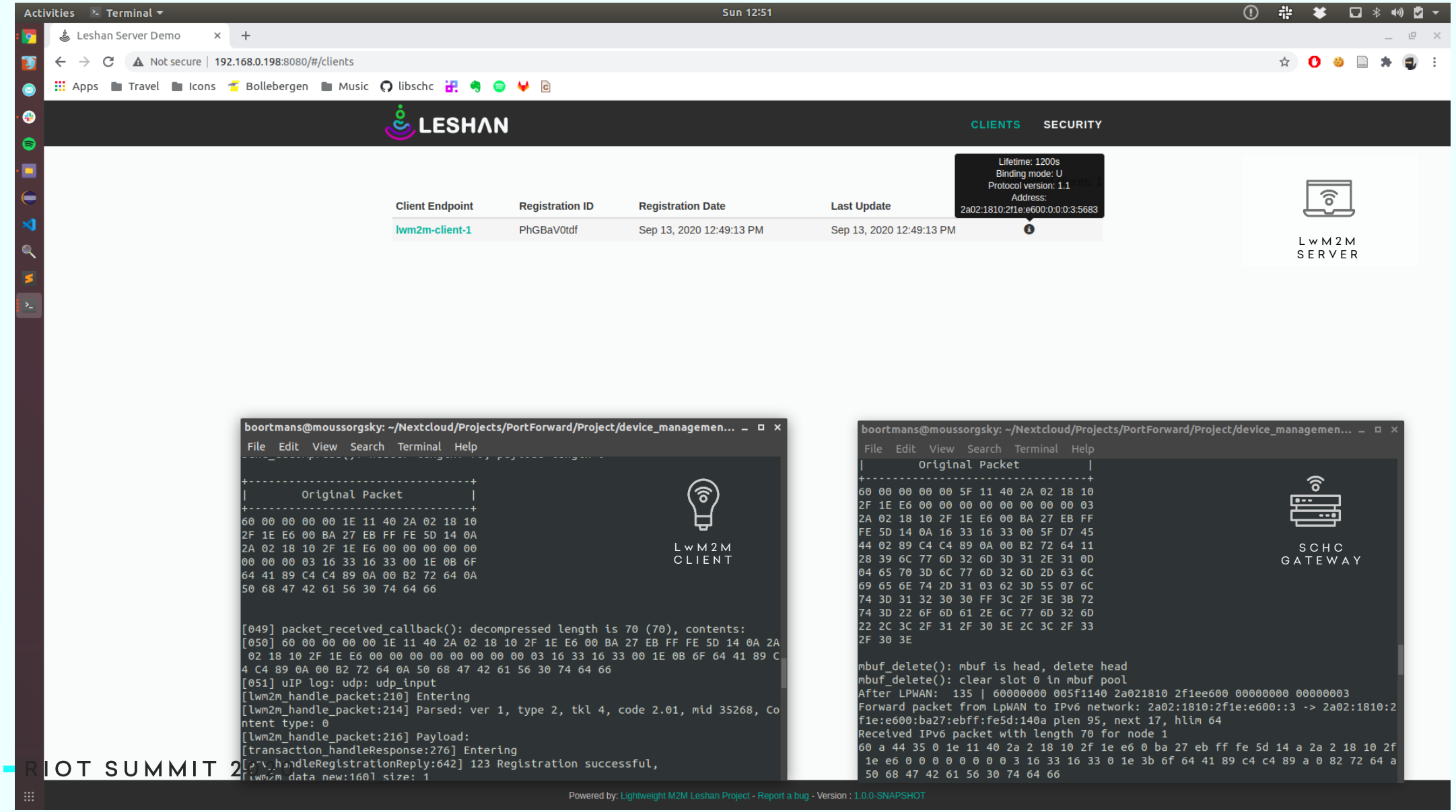
Sensor response to a CoAP POST request.





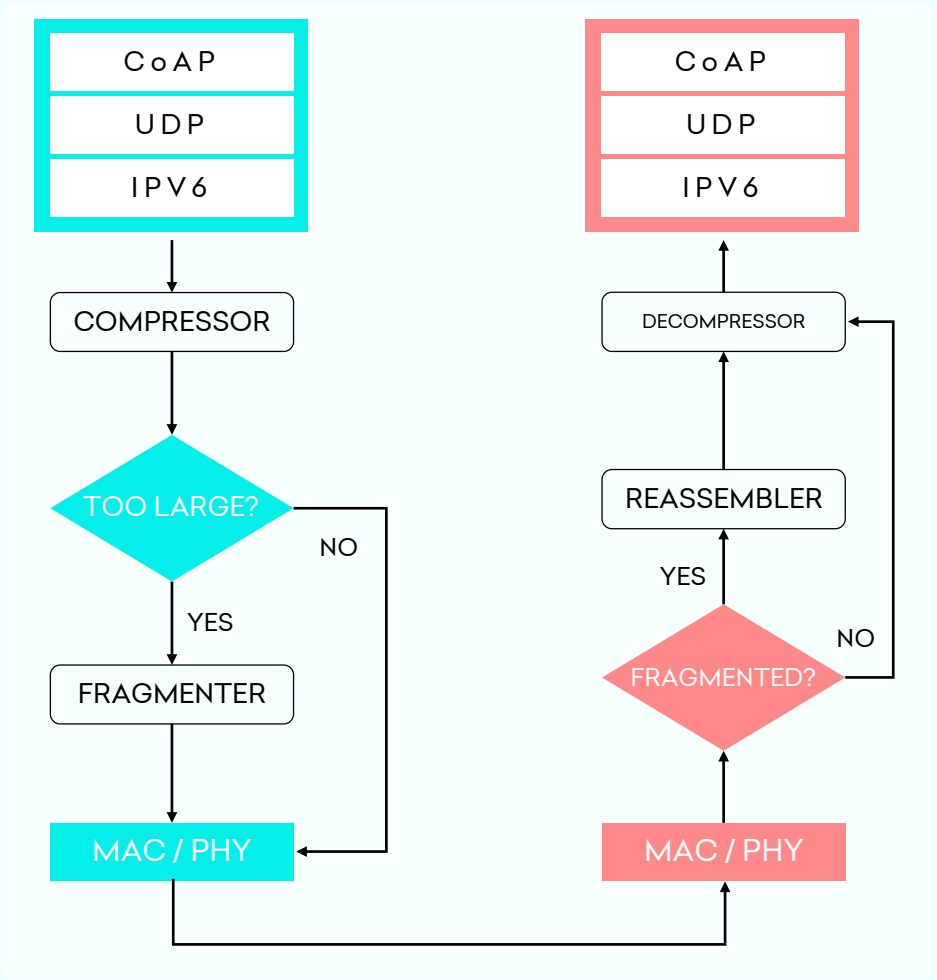
LwM2M* EXAMPLE

* or any IPv6 enabled service



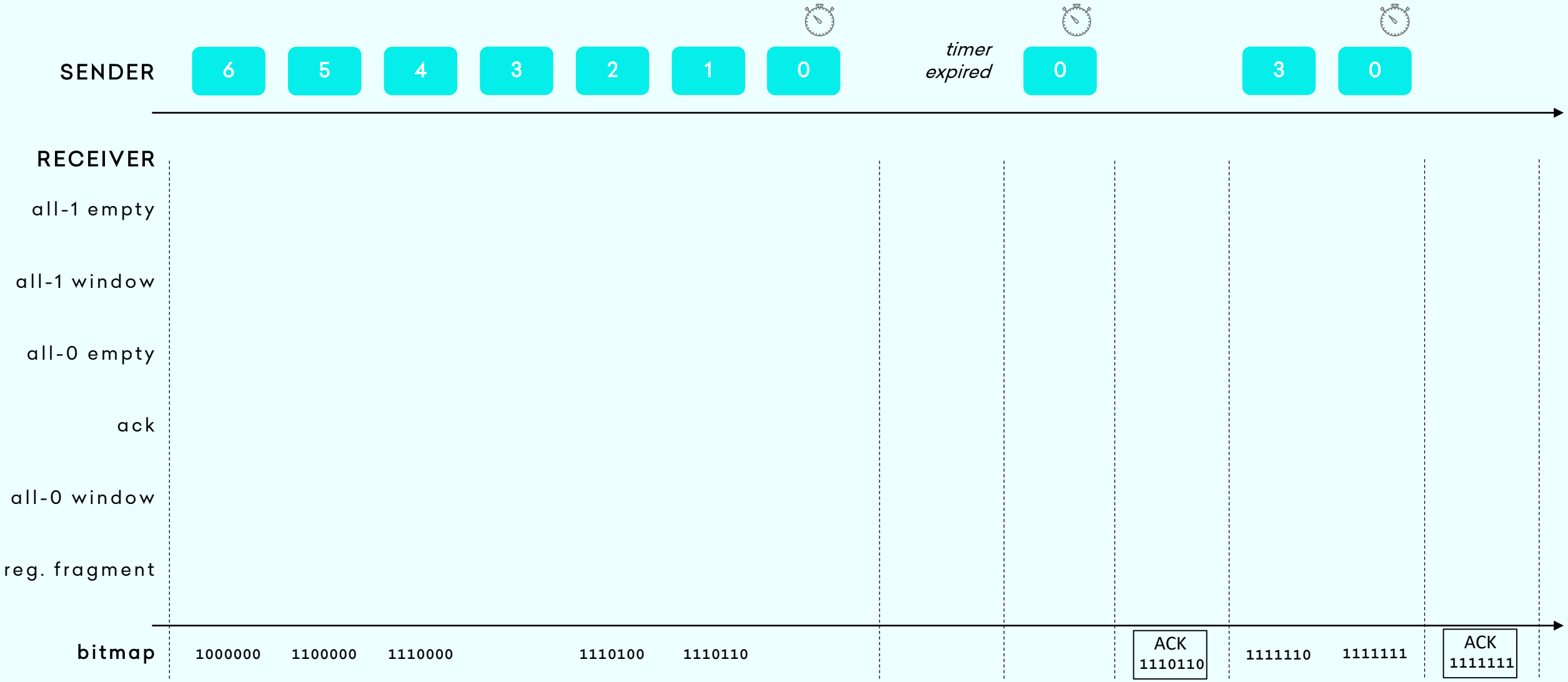
FRAGMENTATION

- Packet remains too large
- Cope with IPv6 MTU value for routers (1280 bytes): **fragmentation**
  - No-ack : no reliability
  - Ack-On-Error : acknowledge an erroneous window
  - Ack-Always : total reliability

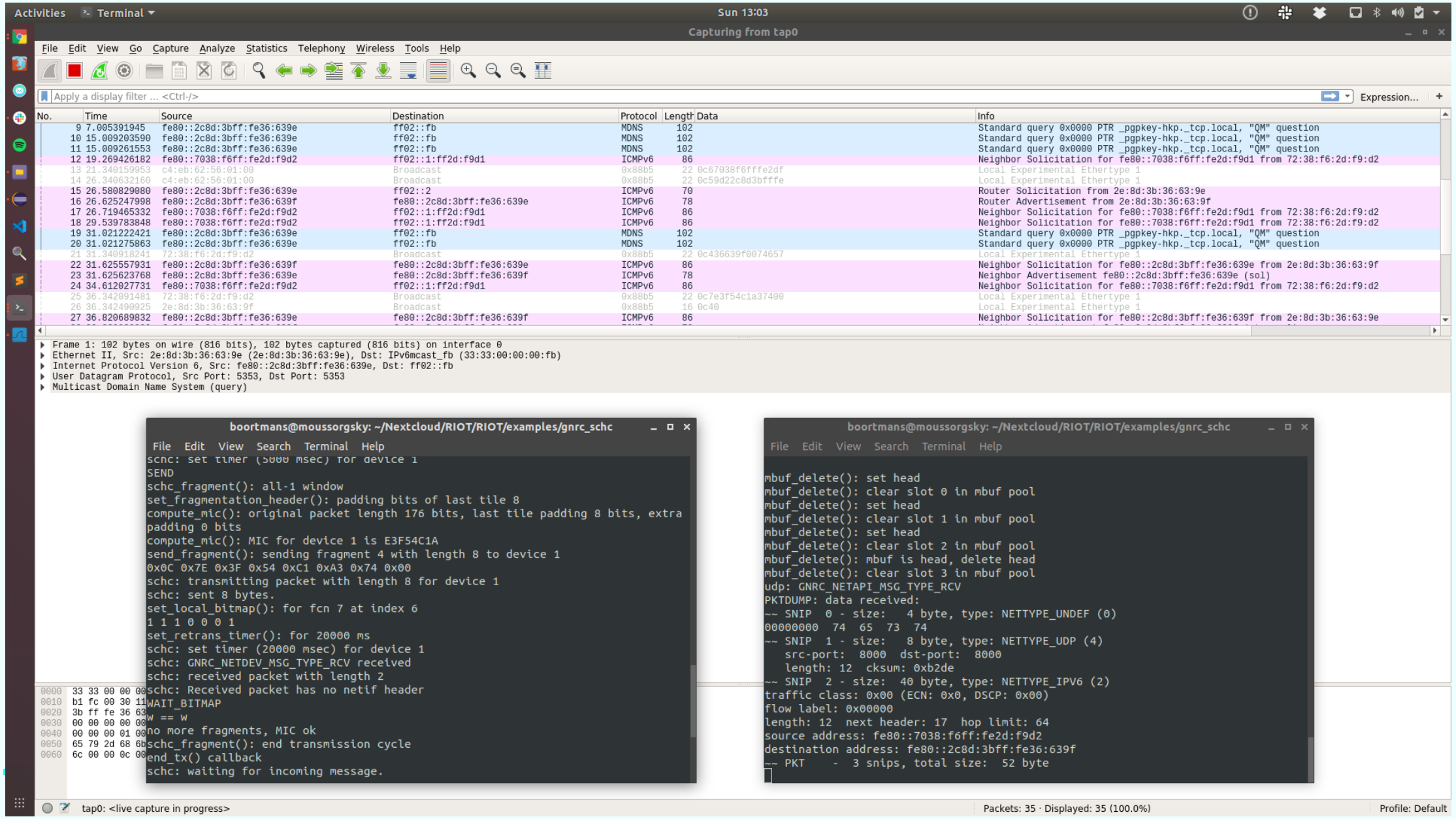


SCHC STATE MACHINE

FRAGMENTATION EXAMPLE



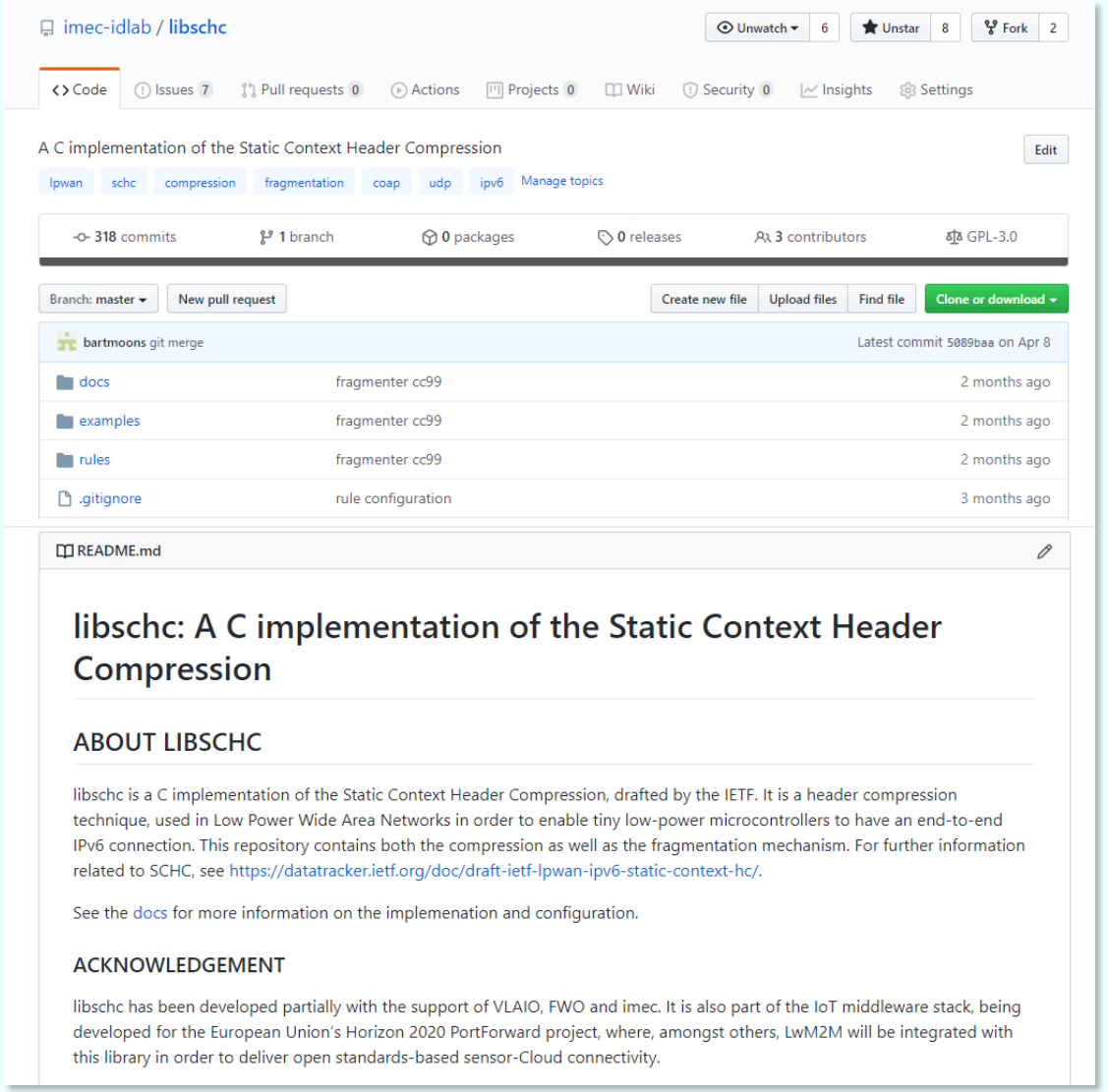
RIOT INTEGRATION



LIBSCHC

Open source implementation of

- Rule context
- Compression
  - Interoperability tested using test vectors of the LPWAN WG
- Fragmentation/reassembly
  - Tiles, windows, bitmaps, timers
  - Integrity checking
  - No-Ack, Ack-Always, Ack-on-Error
- Padding management



https://github.com/imec-idlab/libschc

CURRENT IMPLEMENTATION

Memory management

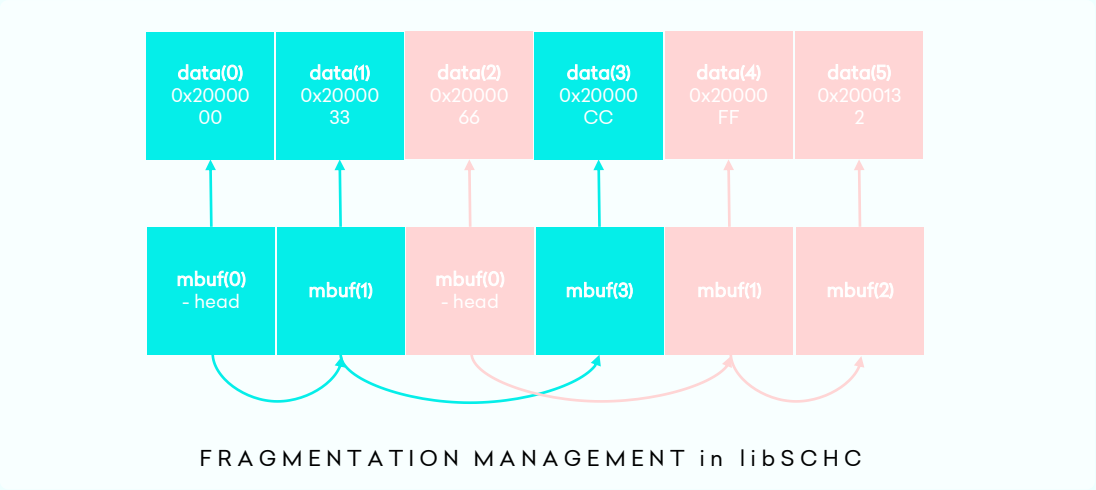
- fragmented packet uses pre-allocated chunk of memory, stored in a **mbuf** (*network memory buffer*) **chain**
- contains a pointer to headers and payload

```
typedef struct schc_mbuf_t {
        ...
        /* the length of the fragment */
        uint16_t len;
        /* pointer to the chunk of memory */
        uint8_t* data;
        /* pointer to the next mbuf */
        struct schc_mbuf_t next;
} schc_mbuf_t;
```

FRAGMENTATION MANAGEMENT

Abstraction required

- differentiate between end-devices
- possibility to
  - reorder linked list (missing fragments)
  - take a single **payload** byte from the chain
    - to calculate MIC

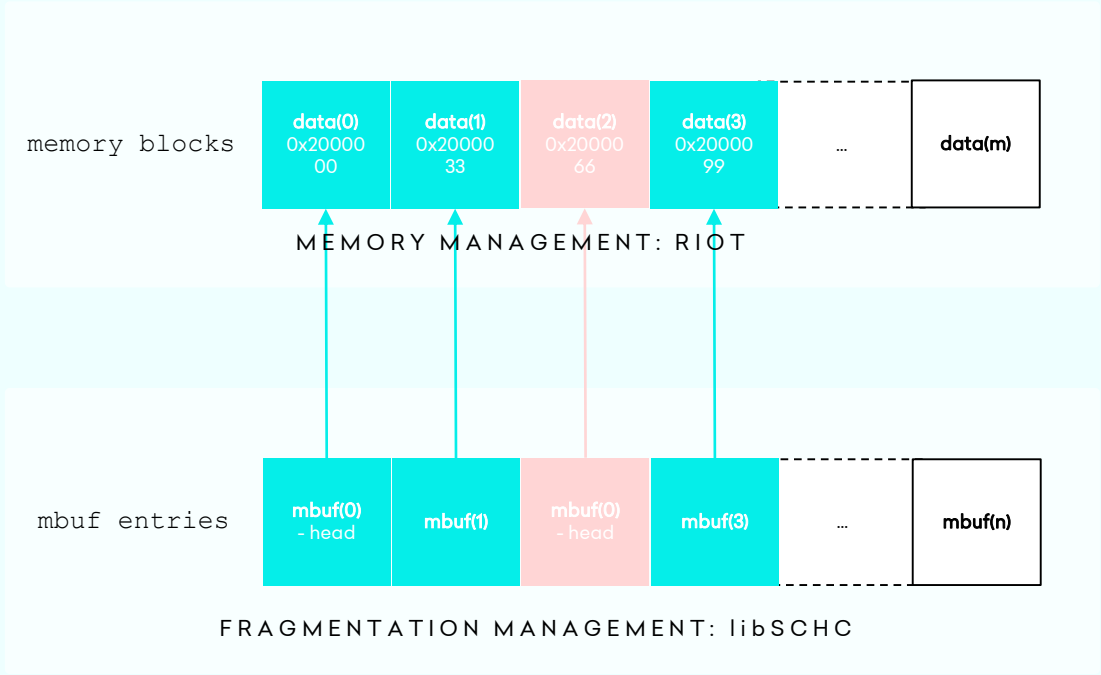


FRAGMENTATION MANAGEMENT in libSCHC

RIOT INTEGRATION

Currently poor memory management in `libSCHC`.

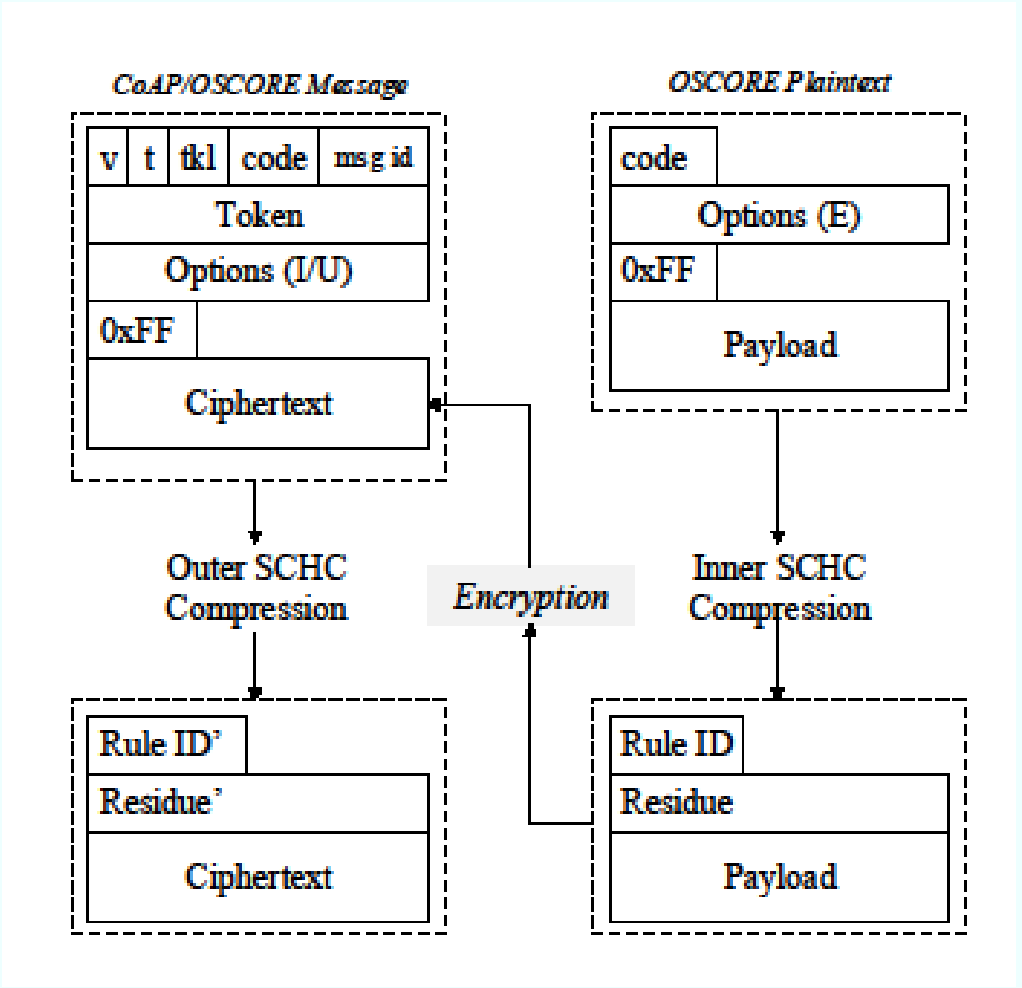
- Separate memory and `mbuf` logic and instead use `pktsnip, pktqueue`
- Pass stripped, concatenated fragments to the IPv6 layer without copying



WHAT'S NEXT?

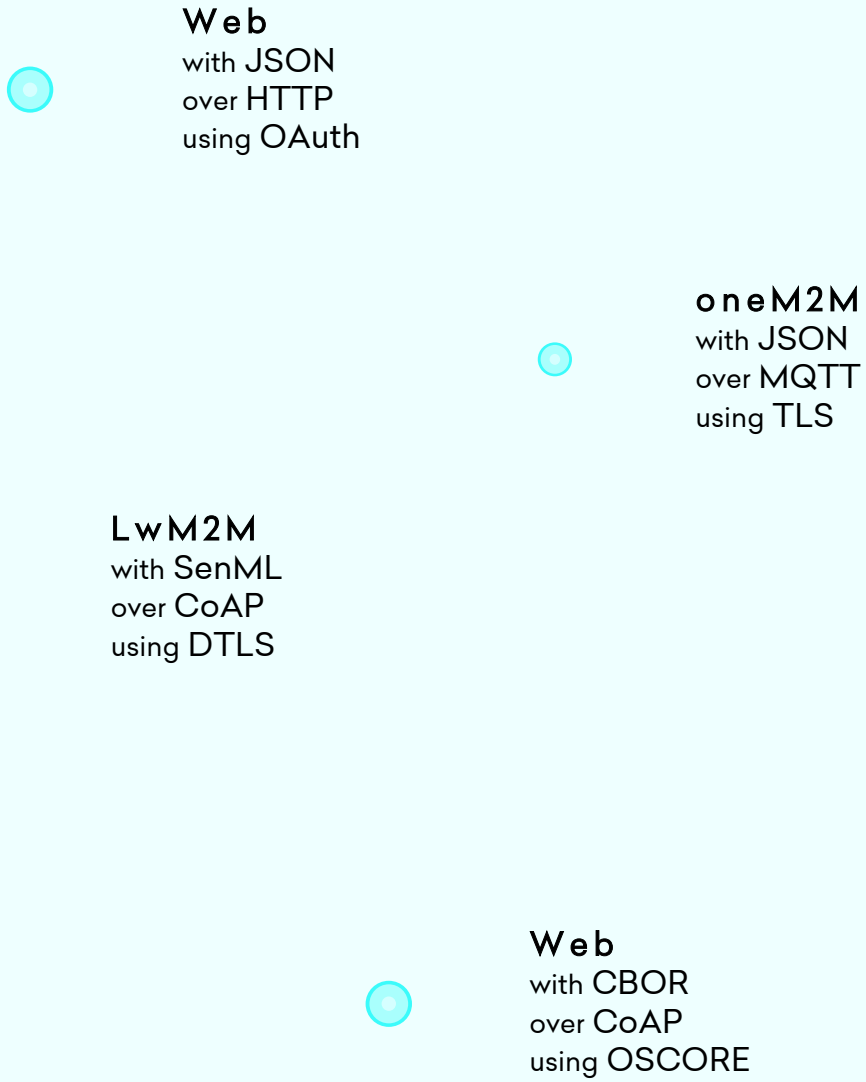
SCHC specifies CoAP w/ OSCORE compression for end-to-end security.

- Integrate libOSCORE and libSCHC for end-to-end security in LPWANs



WHAT'S NEXT?

- SCHC is a generic framework:
  - HTTP
  - CoAP
  - MQTT
  - …
  - NDN?
- RIOT/LoRaWAN gateway w/o MQTT
- Firmware updates for LPWANs

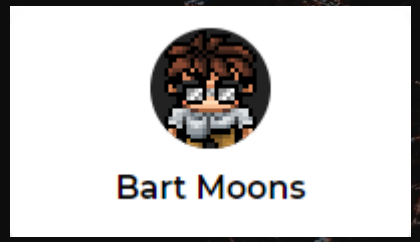


Static Context Header Compression [sjiek] in RIOT

BART MOONS.

Ghent University – IDLab – imec

bamoons.moons@ugent.be
idlab.ugent.be



GHENT UNIVERSITY

IDLab
INTERNET & DATA LAB

imec
embracing a better life