

LPWAN WG

WG Chairs:

Alexander Pelov a@ackl.io

Pascal Thubert pthubert@cisco.com

AD: Eric Vyncke

<evyncke@cisco.com>

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (<https://www.ietf.org/contact/ombudsteam/>) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

[BCP 9](#) (Internet Standards Process)

[BCP 25](#) (Working Group processes)

[BCP 25](#) (Anti-Harassment Procedures)

[BCP 54](#) (Code of Conduct)

[BCP 78](#) (Copyright)

[BCP 79](#) (Patents, Participation)

<https://www.ietf.org/privacy-policy/> (Privacy Policy)



Reminder:

Minutes are taken *

This meeting is recorded **

Presence is logged ***

- * All: Please contribute online to the minutes at: <https://codimd.ietf.org/notes-ietf-110-lpwan>
- ** Recordings and Minutes are public and may be subject to discovery in the event of litigation
- *** Based on Meetecho Attendance Record. You can log once and only once to each session.

Minute takers, jabber scribes



- Minutes
 - CodiMD: <https://codimd.ietf.org/notes-ietf-110-lpwan>
 - Minute takers volunteers?
- Remote participation
 - Meetecho: <https://meetings.conf.meetecho.com/ietf110/?group=lpwan&short=&item=1>
 - Jabber: lpwan@jabber.ietf.org
 - Jabber scribe volunteers?
- Mailing list: lp-wan@ietf.org
 - To subscribe: <https://www.ietf.org/mailman/listinfo/lp-wan>
- Meeting materials: <https://datatracker.ietf.org/meeting/110/session/lpwan>

Agenda bashing

15:30 (UTC)	Opening, agenda bashing (Chairs) <ul style="list-style-type: none">• Note-Well, Scribes, Agenda Bashing• Status of drafts• Presenters: The Chairs	5mn
15:35	LPWAN Architecture and general newcomer presentation <ul style="list-style-type: none">• Presenter: Alexander Pelov• Associated drafts: draft-pelov-lpwan-architecture	15mn
15:50	Yang Data Model for SCHC <ul style="list-style-type: none">• Presenter: Laurent Toutain• Associated drafts: draft-ietf-lpwan-schc-yang-data-model	10mn
16:00	New ideas for SCHC-over-SigFox <ul style="list-style-type: none">• Presenters: Juan Carlos Zuniga / Sergio Aguilar• Associated drafts: draft-ietf-lpwan-schc-over-sigfox	15mn

Agenda bashing (cont.)

- | | | |
|-------|--|-----|
| 16:15 | CoAP-over-SCHC | 5mn |
| | <ul style="list-style-type: none">• Presenter: Ana Minaburo• Associated drafts: draft-ietf-lpwan-coap-static-context-hc | |
| 16:20 | SCHC-over-PPP | 5mn |
| | <ul style="list-style-type: none">• Presenters: Pascal Thubert• Associated drafts: draft-thubert-intarea-schc-over-ppp | |
| 16:25 | AOB | 5mn |
| 16:30 | Meetecho ends | |

WG Status

Date	Milestone	↕
Feb 2022	Produce a Standards Track document for SCHC over NB-IoT draft-ietf-lpwan-schc-over-nbiot	
Oct 2021	Produce a Standards Track document for SCHC over SigFox draft-ietf-lpwan-schc-over-sigfox	
Jul 2021	Produce a Standards Track document to enable operations, administration and maintenance (OAM) to the LPWAN device, including support for delayed or proxied liveness verification (Ping)	
Feb 2021	Produce a Standards Track document to define the generic data models to formalize the compression and fragmentation contexts for LPWANs	
Dec 2020	Produce Standard Track documents to apply SCHC IPv6/UDP over the baseline technologies	
May 2020	Perform SCHC Maintenance, including enabling SCHC mechanisms for Upper layer Protocols	

Document's advancement

draft-ietf-lpwan-coap-static-context-hc-19 LPWAN Static Context Header Compression (SCHC) for CoAP	2021-05-08 34 pages	New RFC Ed Queue : EDIT for 2 days Submitted to IESG for Publication: Proposed Standard Reviews: genart, iotdir, opsdir, secdir, tsvalt	Éric Vyncke Pascal Thubert
draft-ietf-lpwan-schc-over-lorawan-14 Static Context Header Compression (SCHC) over LoRaWAN	2021-01-25 28 pages	RFC Ed Queue RFC-EDITOR for 44 days Submitted to IESG for Publication: Proposed Standard Reviews: genart, iotdir, opsdir, secdir, tsvalt	1 Éric Vyncke Dominique Barthel
draft-ietf-lpwan-schc-over-nbiot-04 SCHC over NB-IoT	2021-01-19 22 pages	EDIT = Awaiting editing or being edited RFC-EDITOR = Undergoing final internal review before AUTH48	
draft-ietf-lpwan-schc-over-sigfox-05 SCHC over Sigfox LPWAN	2021-02-22 23 pages	I-D Exists WG Document Oct 2021	Éric Vyncke
draft-ietf-lpwan-schc-yang-data-model-04 Data Model for Static Context Header Compression (SCHC)	2021-02-02 42 pages	I-D Exists WG Document Reviews: yangdoctors	Éric Vyncke

RFCs (2 hits)

RFC 8376 (was draft-ietf-lpwan-overview) Low-Power Wide Area Network (LPWAN) Overview	2018-05 43 pages	Informational RFC	Suresh Krishnan Alexander Pelov
RFC 8724 (was draft-ietf-lpwan-ipv6-static-context-hc) SCHC: Generic Framework for Static Context Header Compression and Fragmentation	2020-04 71 pages	Proposed Standard RFC	Suresh Krishnan Pascal Thubert

Document

↕ Date

↕ Status

↕ IPR

↕ AD / Shepherd

Related Internet-Drafts (2 hits)

draft-barthel-lpwan-oam-schc-02 OAM for LPWAN using Static Context Header Compression (SCHC)	2020-11-02 14 pages	I-D Exists	
draft-pelov-lpwan-architecture-00 Static Context Header Compression (SCHC) Architecture	2021-01-19 6 pages	I-D Exists	

LPWAN Architecture and general newcomer presentation

Alexander Pelov (a@ackl.io)

Pascal Thubert (pthubert@cisco.com)

Ana Minaburo (ana@ackl.io)

Low-Power Wide-Area Networks

25 mW transmission power

Low-Power Wide-Area Networks

20 years on simple battery

15-50 km rural outdoor

Low-Power **Wide-Area** Networks

2-3 km urban indoor

No scheduling

Star topology
ALOHA

Low-Power Wide-Area **Networks**

Device-initiated com

Huge densities

Asymmetric links Low throughput

Collisions

Duty cycling

Acknowledgements

Data-over-NAS In-band

Guard-bands

License free

In licensed spectrum

Low-Power Wide-Area Networks

Collisions

Data-over-NAS In-band

Duty cycling

Acknowledgements

Guard-bands

License free

In licensed spectrum

No scheduling

Star topology

ALOHA

25 mW transmission power

15-50 km rural outdoor

Low-Power Wide-Area Networks

20 years on simple battery

2-3 km urban indoor

Device-initiated com

Huge densities

Asymmetric links Low throughput

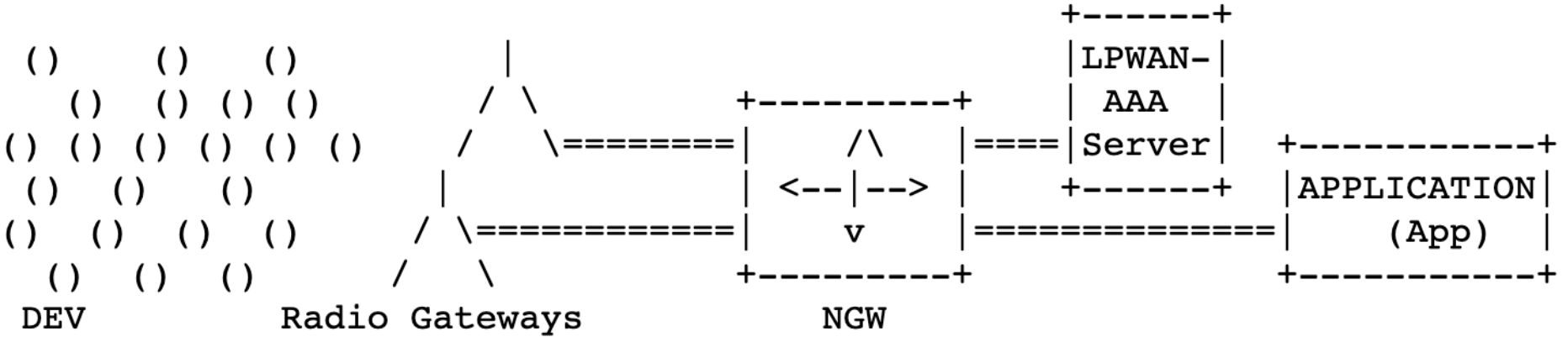
x 10%

100 bps
(50 kbps max)

12 byte payload
(typically 50 bytes)

140 messages
uplink

4 messages
downlink



RFC 8376 : LPWAN Architecture

Function/ Technology	LoRaWAN	NB-IoT	Sigfox	Wi-SUN	IETF
Sensor, Actuator, device, object	End Device	User Equipment	End Point	Leaf Node	Device (DEV)
Transceiver Antenna	Gateway	Evolved Node B	Base Station	Router Node	Radio Gateway
Server	Network Server	PDN GW/ SCEF*	Service Center	Border Router	Network Gateway (NGW)
Security Server	Join Server	Home Subscriber Server	Registration Authority	Authent. Server	LPWAN- AAA Server
Application	Application Server	Application Server	Network Application	Appli- cation	Application (App)

SCHC Architecture

- Provide the reference architecture
 - Modes:
 - SCHC Device/SCHC Gateway
 - SCHC Peers
- RFC 8724 and Rules
 - Introduces Yang data model
 - Discusses rule creation and update
 - Discusses rule installation and discovery

SCHC Architecture

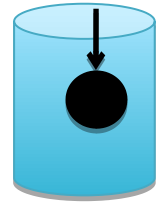


Device
Application



End-Device

Network
Application

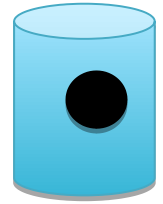
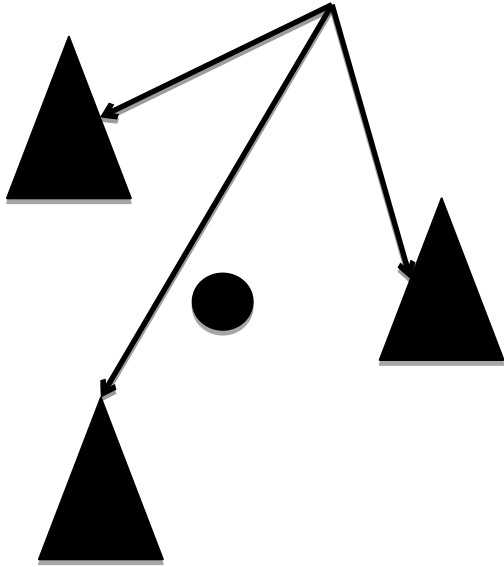


Application
Server

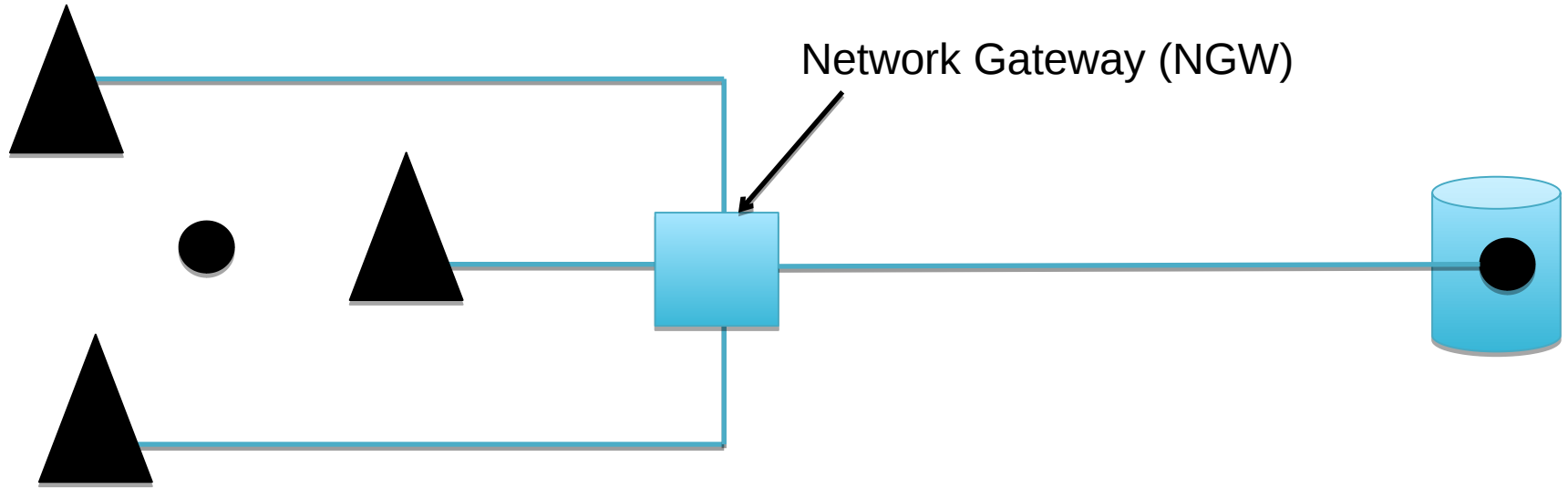
SCHC Architecture

LPWAN

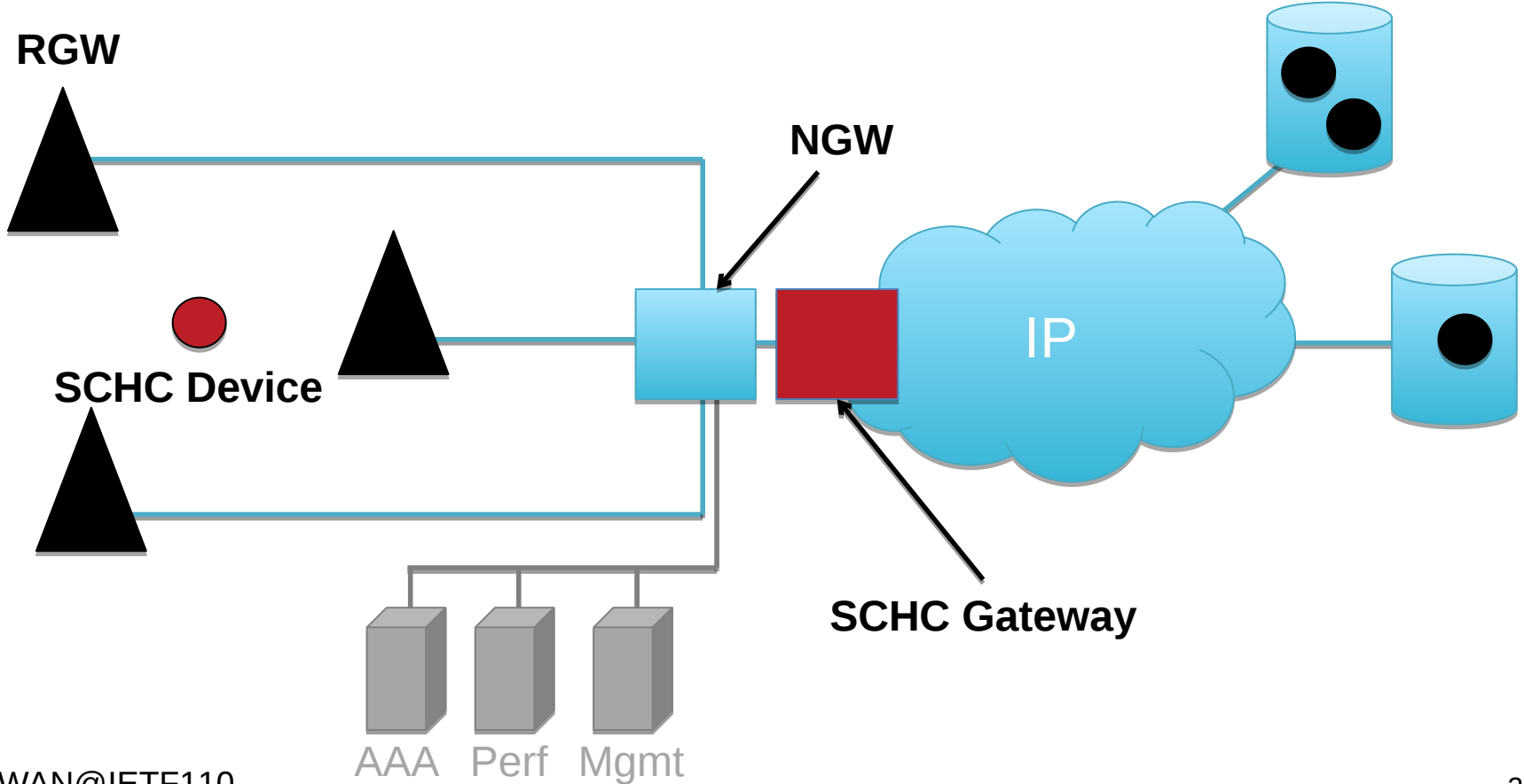
Radio Gateways (RGW)



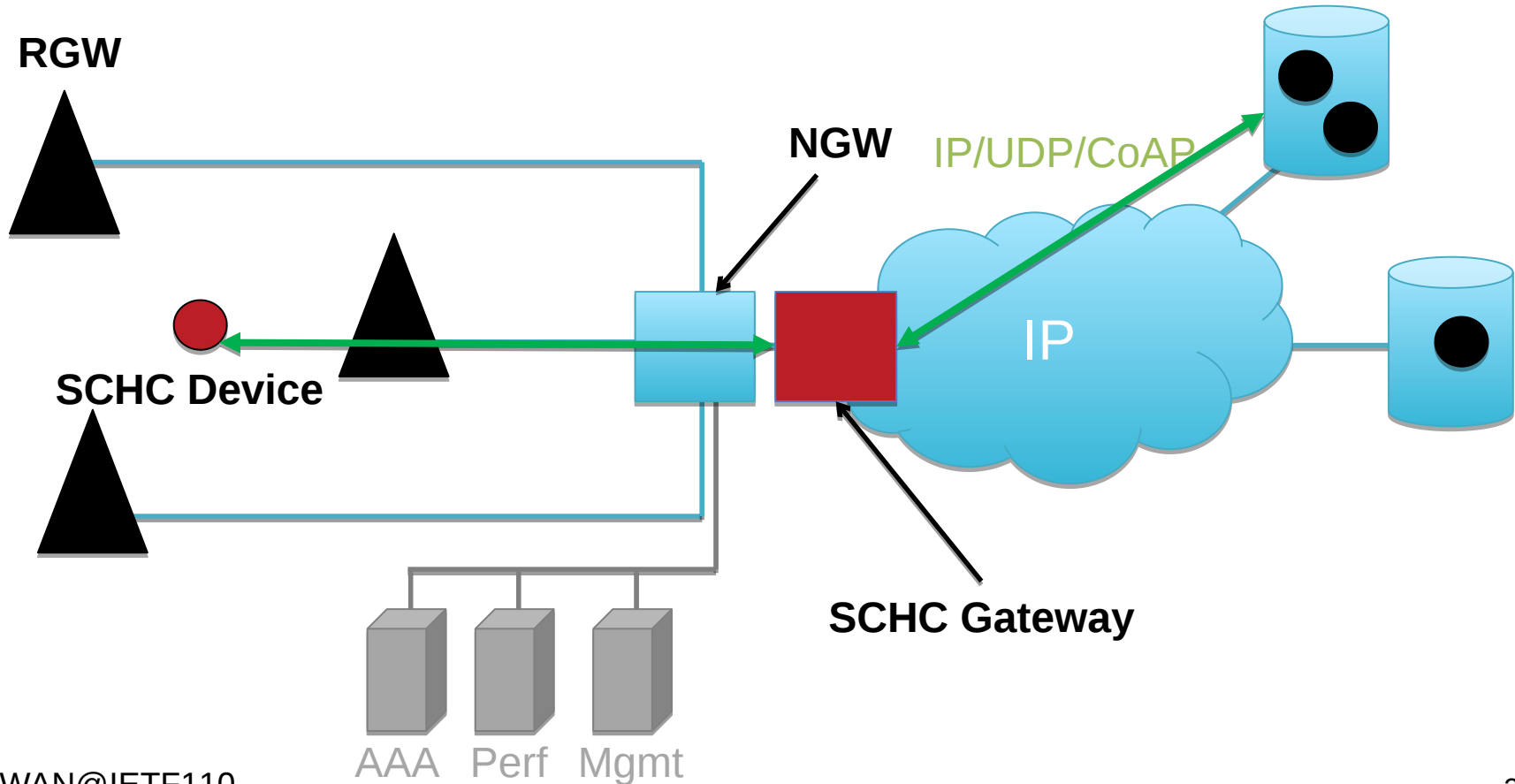
SCHC Architecture



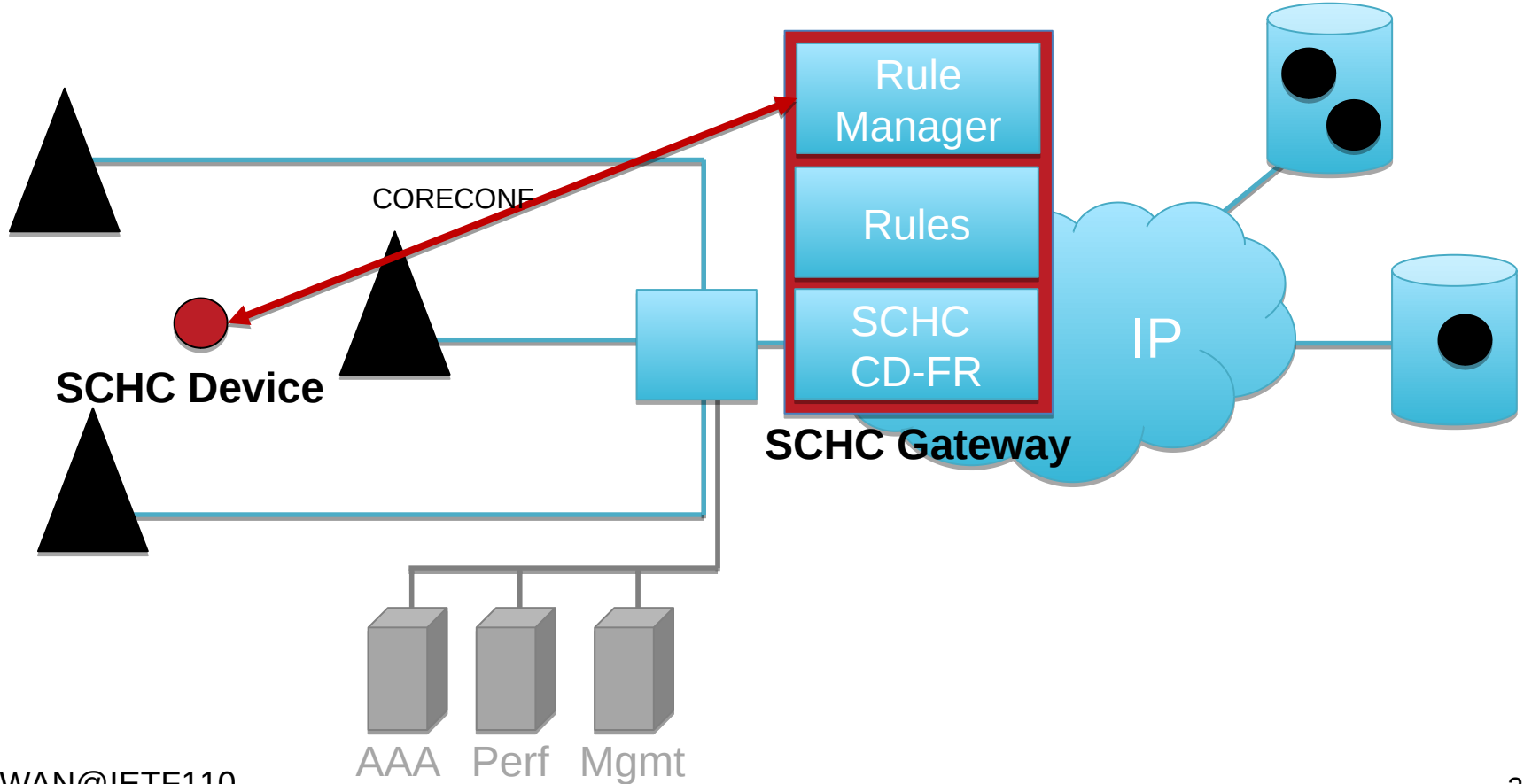
SCHC Architecture



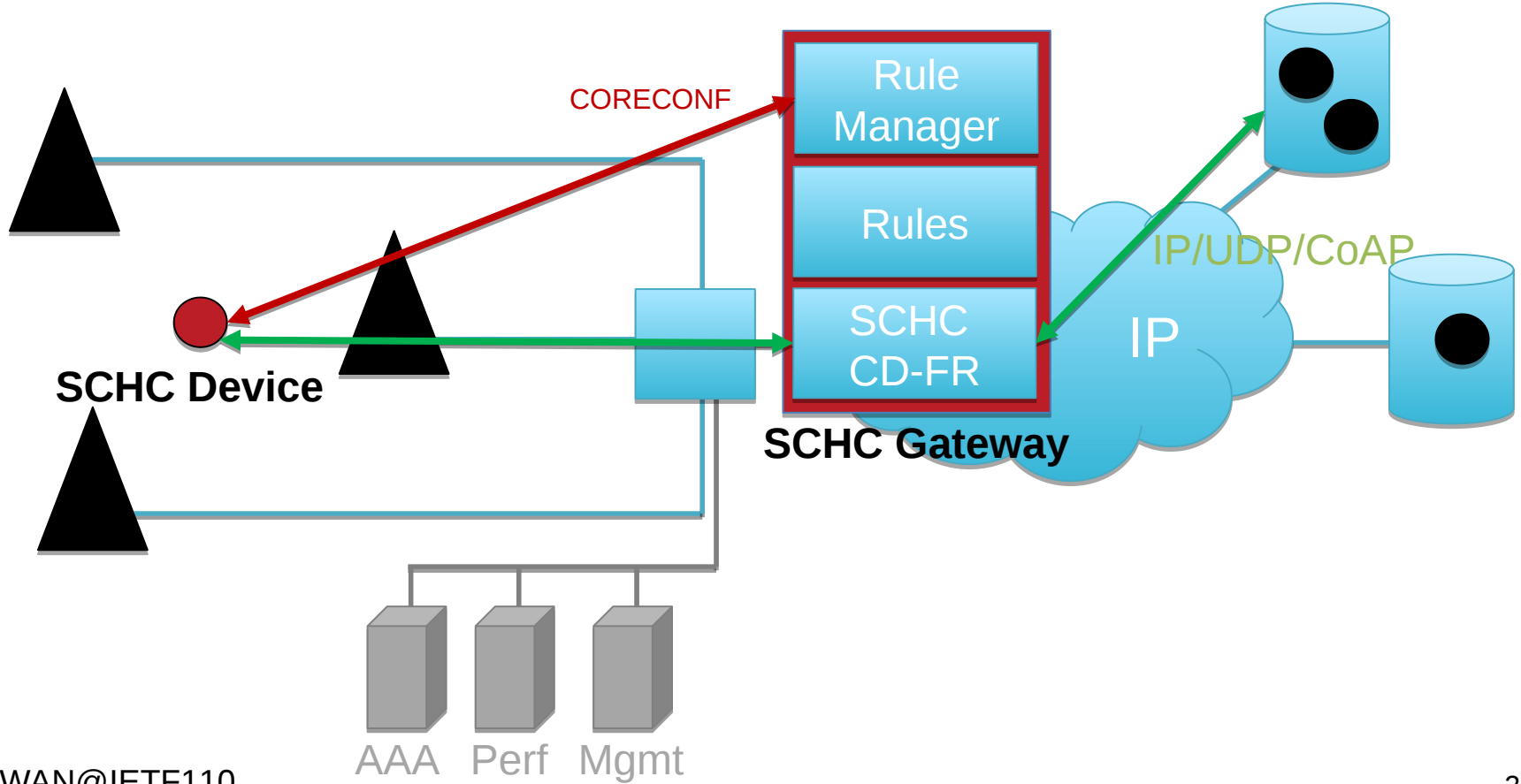
SCHC Architecture



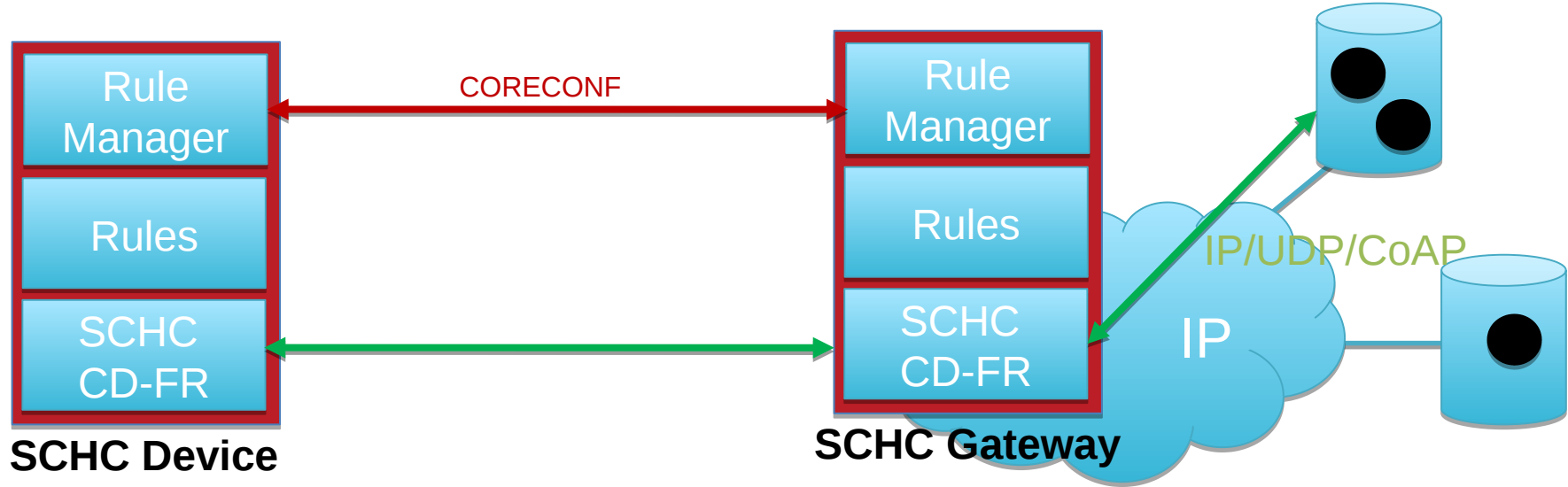
SCHC Architecture



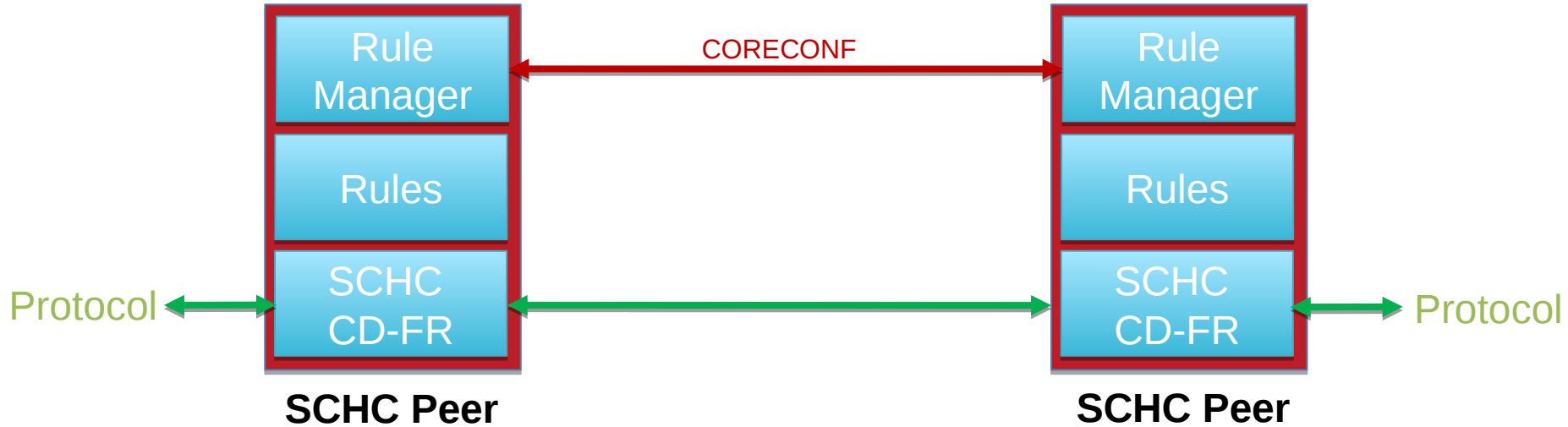
SCHC Architecture



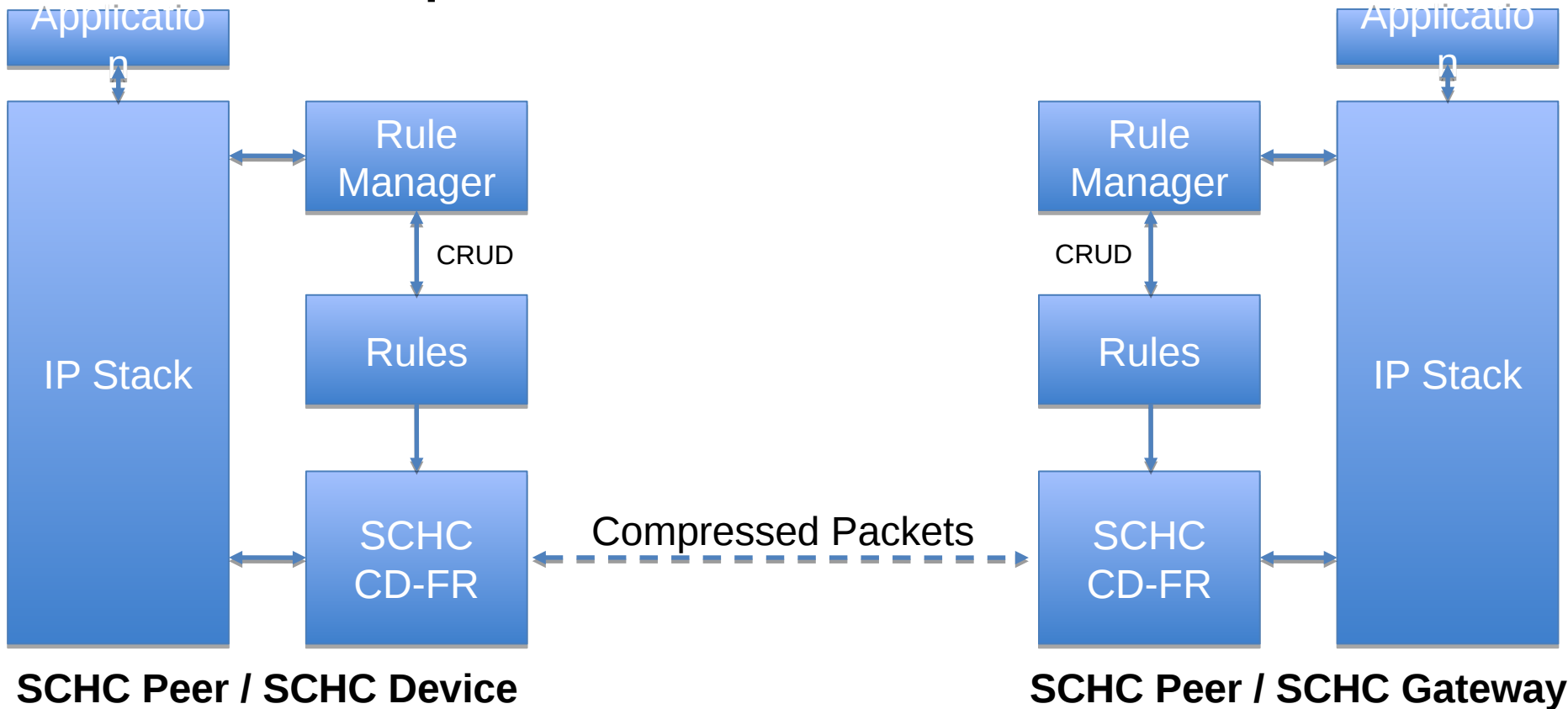
SCHC Architecture



SCHC Architecture



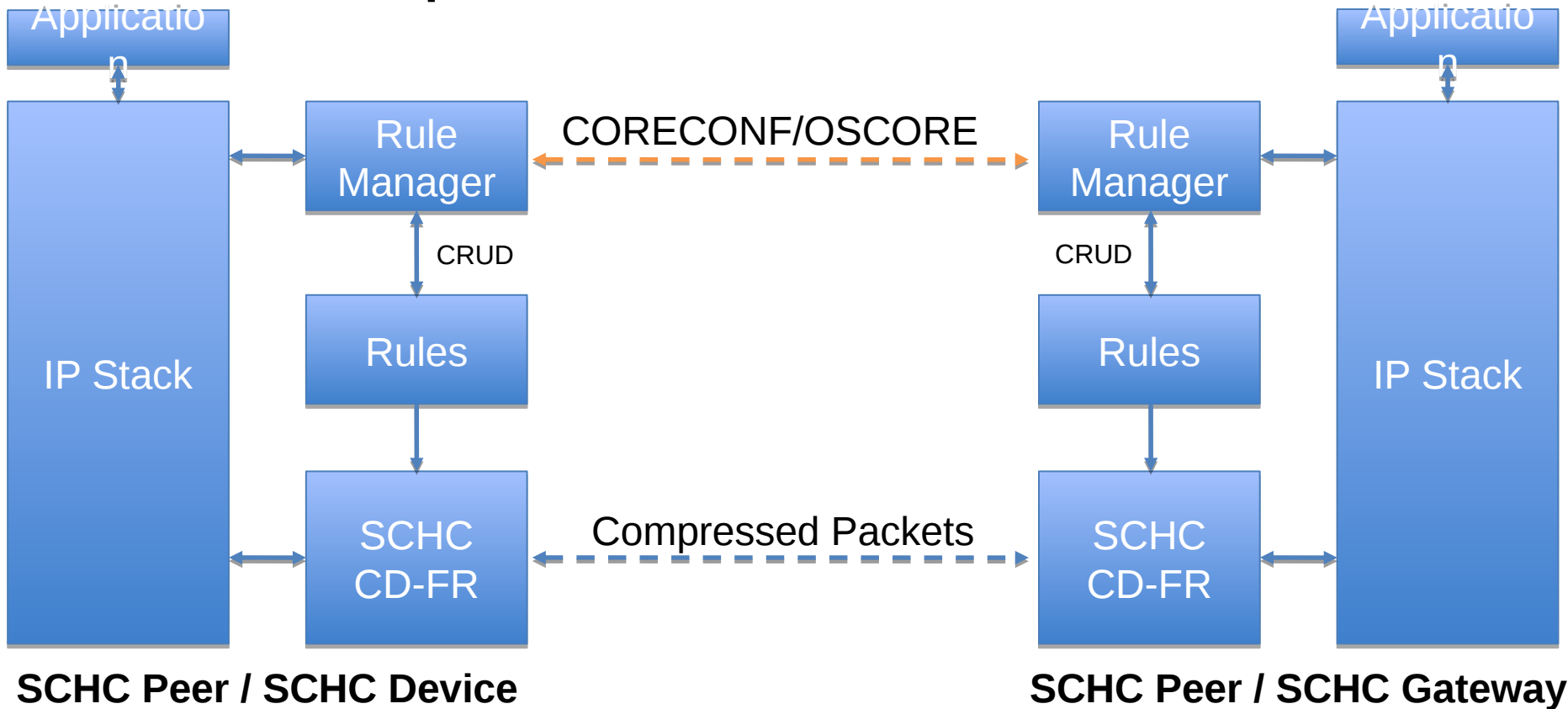
Example of Network-level SCHC



SCHC Peer / SCHC Device

SCHC Peer / SCHC Gateway

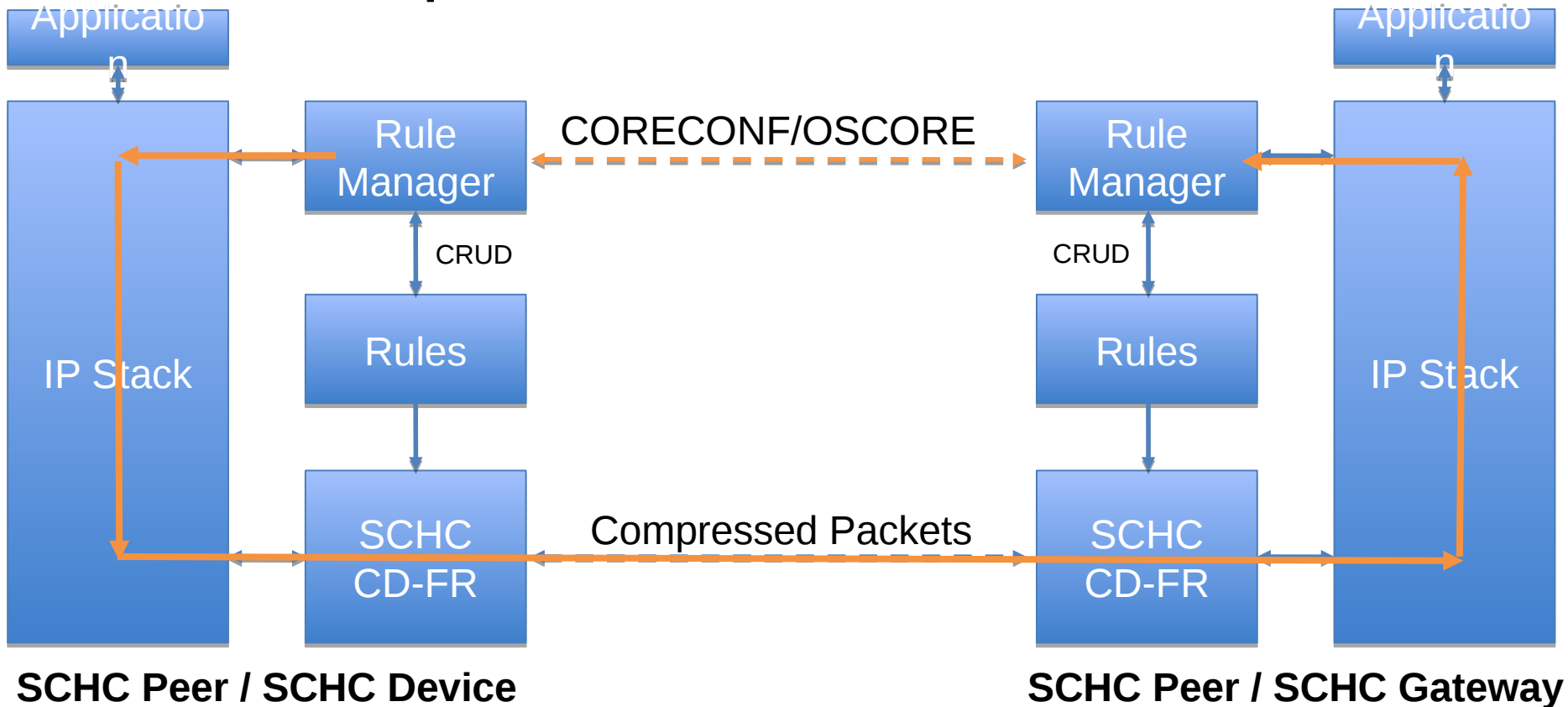
Example of Network-level SCHC



SCHC Peer / SCHC Device

SCHC Peer / SCHC Gateway

Example of Network-level SCHC



SCHC Peer / SCHC Device

SCHC Peer / SCHC Gateway

draft-ietf-lpwan-schc-yang-data-model-04

Authors:

Laurent Toutain (laurent.toutain@imt-atlantique.fr)

Ana Minaburo (ana@ackl.io)

What's new from -03

- Clarifications
- Goal :
 - Define a way to uniquely refers to an ID
 - Field ID
 - MO, CDA
 - Fragmentation parameter
 - Define a structure for rules
 - Inter exchange (see architecture draft)
 - Not internal representation
- Don't focus on size, this is an abstract representation:
 - CORECONF will help to reduce this size

Identifiers

Identifier definition logic

```
// -- RCS algorithm types

identity RCS-algorithm-base-type {
    description "identify which algorithm is used to compute RSC.
    The algorithm defines also the size of the RSC field.";
}

identity RFC8724-RCS {
    description "CRC 32 defined as default RCS in RFC8724.";
    base RCS-algorithm-base-type;
}

typedef RCS-algorithm-type {
    type identityref {
        base RCS-algorithm-base-type;
    }
}
```

Id types:

- **field-id-type**: IPv6, UDP, CoAP, OSCORE, ICMPv6 (56 values)
- **field-length-type**: union int64 and identityref (2 values)
- **direction-indicator-type**: 3 values
- **matching-operator-type**: 4 values
- **comp-decomp-action-type**: 8 values
- **RCS-algorithm-type**: 1 value
- **fragmentation-mode-type**: 3 values
- **ack-behavior-type**: 3 values
- **all1-data-type**: 3 values

Field id

```
identity /field-id-base-type/fid-icmpv6-checksum
identity /field-id-base-type/fid-icmpv6-code
identity /field-id-base-type/fid-icmpv6-identifier
identity /field-id-base-type/fid-icmpv6-sequence
identity /field-id-base-type/fid-icmpv6-type
identity /field-id-base-type/fid-ipv6-appiid
identity /field-id-base-type/fid-ipv6-appprefix
identity /field-id-base-type/fid-ipv6-deviid
identity /field-id-base-type/fid-ipv6-devprefix
identity /field-id-base-type/fid-ipv6-flowlabel
identity /field-id-base-type/fid-ipv6-hoplimit
identity /field-id-base-type/fid-ipv6-nextheader
identity /field-id-base-type/fid-ipv6-payloadlength
identity /field-id-base-type/fid-ipv6-trafficclass
identity /field-id-base-type/fid-ipv6-trafficclass-ds
identity /field-id-base-type/fid-ipv6-trafficclass-ecn
identity /field-id-base-type/fid-ipv6-version
identity /field-id-base-type/fid-udp-app-port
identity /field-id-base-type/fid-udp-checksum
identity /field-id-base-type/fid-udp-dev-port
identity /field-id-base-type/fid-udp-length
```

Field id: coap

```
identity /field-id-base-type/fid-coap-code
identity /field-id-base-type/fid-coap-code-class
identity /field-id-base-type/fid-coap-code-detail
identity /field-id-base-type/fid-coap-mid
identity /field-id-base-type/fid-coap-option-accept
identity /field-id-base-type/fid-coap-option-block1
identity /field-id-base-type/fid-coap-option-block2
identity /field-id-base-type/fid-coap-option-content-format
identity /field-id-base-type/fid-coap-option-etag
identity /field-id-base-type/fid-coap-option-if-match
identity /field-id-base-type/fid-coap-option-if-none-match
identity /field-id-base-type/fid-coap-option-location-path
identity /field-id-base-type/fid-coap-option-location-query
identity /field-id-base-type/fid-coap-option-max-age
identity /field-id-base-type/fid-coap-option-no-response
identity /field-id-base-type/fid-coap-option-observe
identity /field-id-base-type/fid-coap-option-oscore-flags
identity /field-id-base-type/fid-coap-option-oscore-kid
identity /field-id-base-type/fid-coap-option-oscore-kidctx
identity /field-id-base-type/fid-coap-option-oscore-piv
identity /field-id-base-type/fid-coap-option-proxy-scheme
identity /field-id-base-type/fid-coap-option-proxy-uri
identity /field-id-base-type/fid-coap-option-size1
identity /field-id-base-type/fid-coap-option-size2
identity /field-id-base-type/fid-coap-option-uri-host
identity /field-id-base-type/fid-coap-option-uri-path
identity /field-id-base-type/fid-coap-option-uri-port
identity /field-id-base-type/fid-coap-option-uri-query
identity /field-id-base-type/fid-coap-tkl
identity /field-id-base-type/fid-coap-token
identity /field-id-base-type/fid-coap-type
identity /field-id-base-type/fid-coap-version
```

- Removed « payload marker »
- Added OSCORE
- Added ICMPv6

Structures

SCHC Data Model - Frag

```

module: schc
+--rw schc
+--rw version? uint64
+--rw rule* [rule-id rule-length]
+--rw rule-id uint32
+--rw rule-length uint8
+--rw (nature)?
+--:(fragmentation)
+--rw direction schc:direction-indicator-type
+--rw dtag-size? uint8
+--rw wsize? uint8
+--rw fc-size uint8
+--rw RCS-algorithm? RCS-algorithm-type
+--rw maximum-window-size? uint16
+--rw retransmission-timer? uint64
+--rw inactivity-timer? uint64
+--rw max-ack-requests? uint8
+--rw maximum-packet-size? uint16
+--rw fragmentation-mode schc:fragmentation-mode-type
+--rw (mode)?
+--:(no-ack)
+--:(ack-always)
+--:(ack-on-error)
+--rw tile-size? uint8
+--rw tile-in-All1? schc:all1-data-type
+--rw ack-behavior? schc:ack-behavior-type
+--:(compression)

```

Not in RFC 8724

Up or down, bi forbidden

Frag header, only FCN is mandatory

Mandatory: 1..max

Optional : 0..max

noAck, AA, AoE

No, Yes, Sender Choice

All0, All1, Always

SCHC Data Model - Compression

```

+--:(compression)
  +--rw entry* [field-id field-position direction-indicator]
    +--rw field-id          schc:field-id-type
    +-rw field-length      schc:field-length-type
    +--rw field-position    uint8
    +--rw direction-indicator schc:direction-indicator-type
    +-rw target-values* [position]
      | +-rw value?      union
      | +-rw position    uint16
    +--rw matching-operator schc:matching-operator-type
    +--rw matching-operator-value* [position]
      | +-rw value?      union
      | +-rw position    uint16
    +--rw comp-decomp-action schc:comp-decomp-action-type
    +--rw comp-decomp-action-value* [position]
      +-rw value?      union
      +-rw position    uint16
  
```

Optional ?

What's next

- Draft is stable and has been discussed in interim meetings.
- Ready for Working Group Last Call

draft-ietf-lpwan-schc-over-sigfox-05 & PySCHC Implementation

Juan Carlos Zúñiga (Sigfox), Carles Gómez, Sergio Aguilar (UPC),
Laurent Toutain (IMT-Atlantique),
Sandra Céspedes, Diego Wistuba (U Chile)

Updates

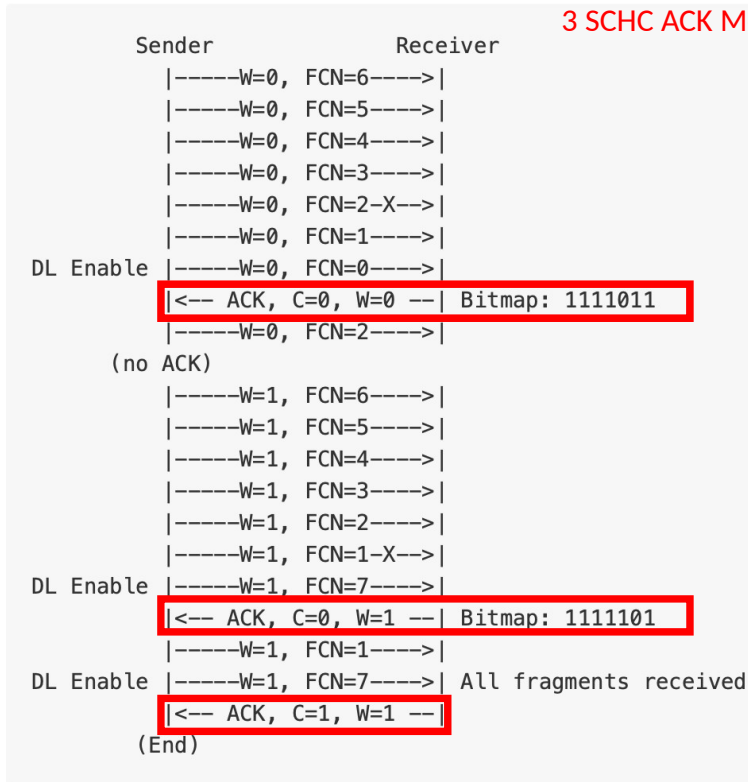
- Hackathon: Off-line coding between UChile and UPC
 - ACK-on-Error and No-ACK SCHC/Sigfox parameter optimizations
 - Error conditions tested
- Last draft updates (rev 05)
 - Added message sequence examples to explain different ACK-on-Error and No-ACK SCHC/Sigfox scenarios
 - Update co-authors' list
- Compound SCHC ACK message proposal (see next slides)

Compound SCHC ACK - Introduction

- ACK-on-Error over Sigfox:
 - Errors in intermediate windows “may” generate at least one ACK
 - Errors in the last window generate at least 2 ACKs
 - Sigfox DL payload is fixed to 64 bits
- When errors occur over multiple windows, the number of ACKs can be reduced by reporting losses from several windows with a single ACK

SCHC Packet: 14 tiles
 Window size: 7 tiles
 3 SCHC ACK Messages

Example

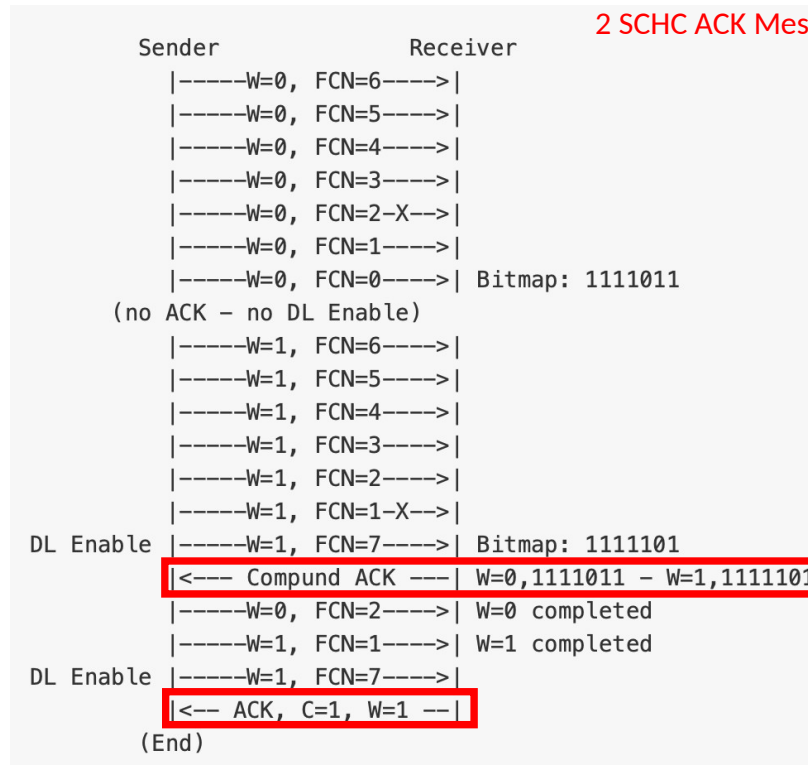


Compound SCHC ACK – Principles and Advantages

- The Compound ACK:
 - Only reports windows with fragment losses
 - Includes W field for each bitmap
 - May not fit all bitmaps of all windows for a SCHC packet
 - Has variable size
 - Compatible with SCHC Receiver Abort message format and ACK Failure message format (RFC8724)
- ACK Reduction when using Compound ACK:
 - Compound ACK messages = Regular SCHC ACKs - (# of windows - 1)

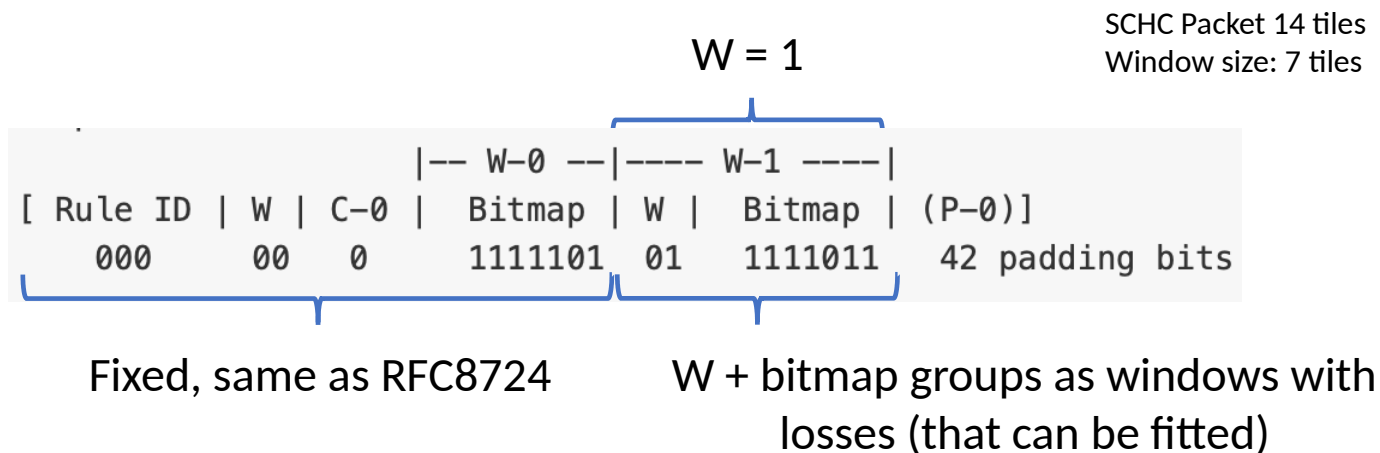
SCHC Packet: 14 tiles
 Window size: 7 tiles
 2 SCHC ACK Messages

Example

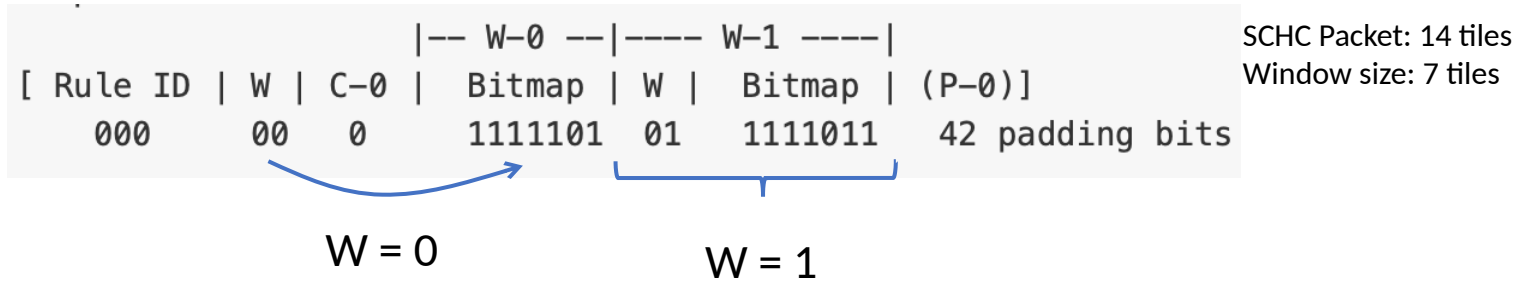


Compound SCHC ACK – Message Format

- Compound ACK message format (only for ACK Failure messages) with SCHC Fragment losses in all windows:



Compound ACK – Message Format



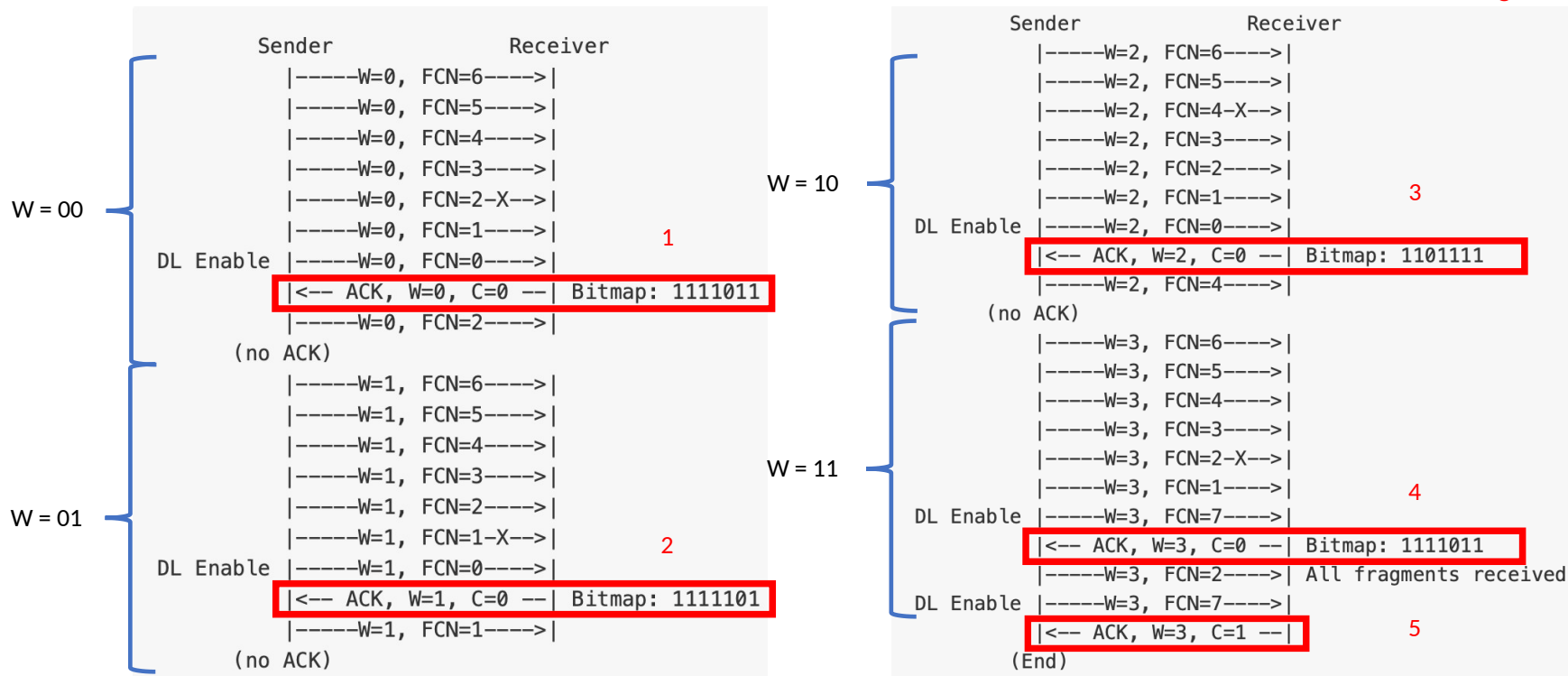
- $W + \text{Bitmap}$ groups are ordered from the smallest window number to the largest
- The window numbered 00 (if present) must always be between the Rule ID and C bit to avoid confusion with padding bits

Thanks!
Questions? Comments?

Backup slides

Example – SCHC Packet 28 tiles

Window size: 7 tiles
5 SCHC ACK Messages

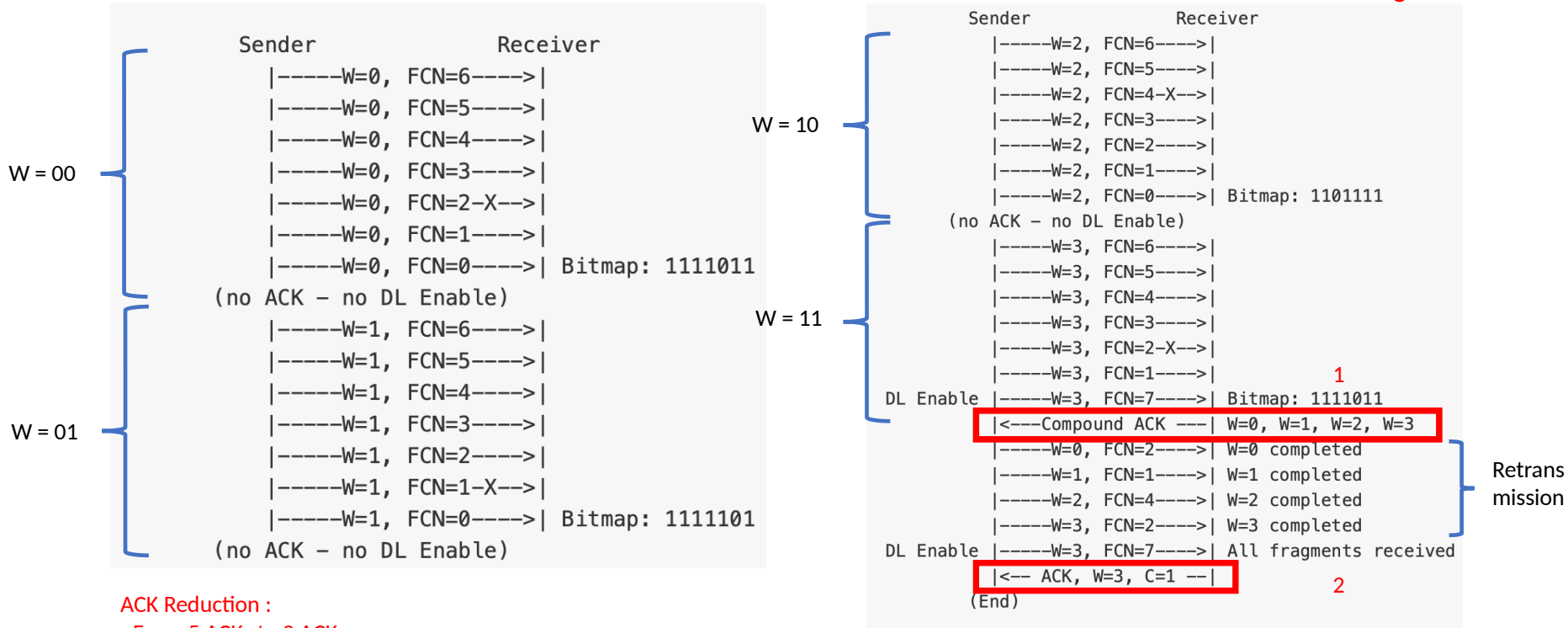


Example – SCHC Packet 28 tiles

SCHC Packet 28 tiles

Window size: 7 tiles

2 SCHC ACK Messages

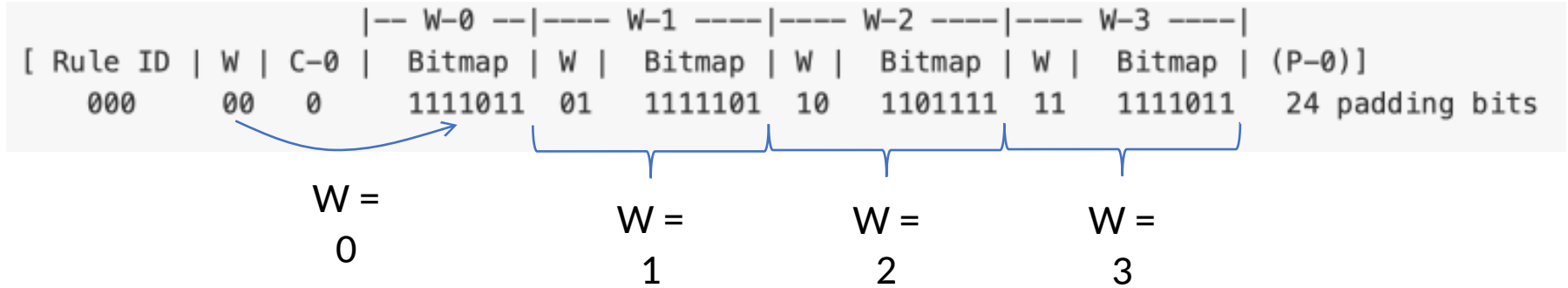


ACK Reduction :

- From 5 ACKs to 2 ACKs

Compound ACK – SCHC Packet 28 tiles

SCHC Packet 28 tiles
Window size: 7 tiles

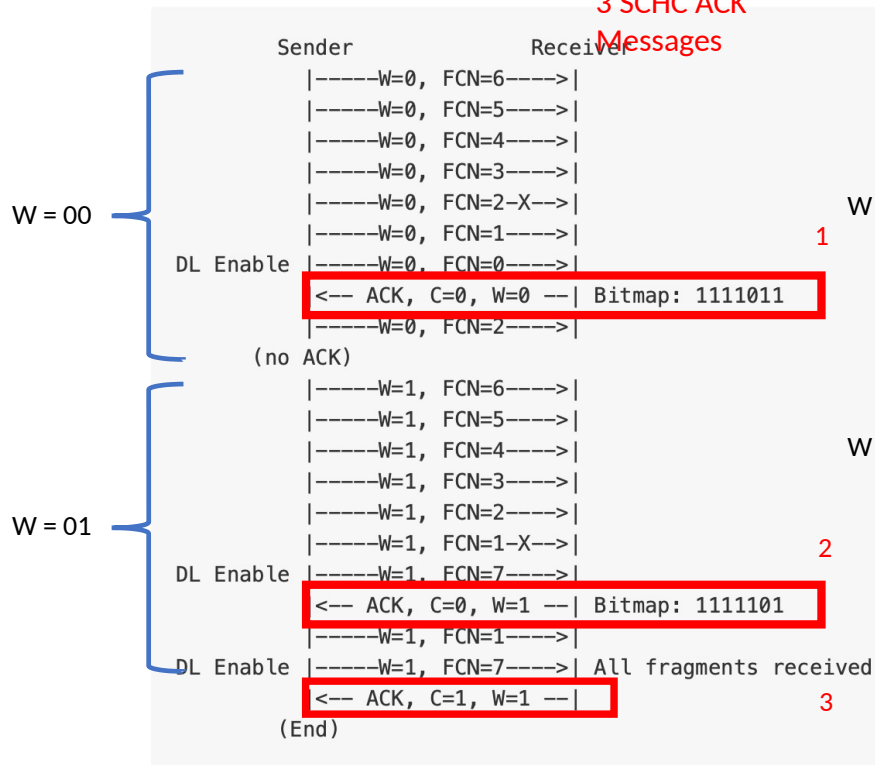


- W + Bitmap groups for all windows with losses
- ACK Reduction
 - Number of windows of the SCHC packet: 4
 - ACK reduction: 3 ACKs

Example – SCHC Packet: 14 tiles

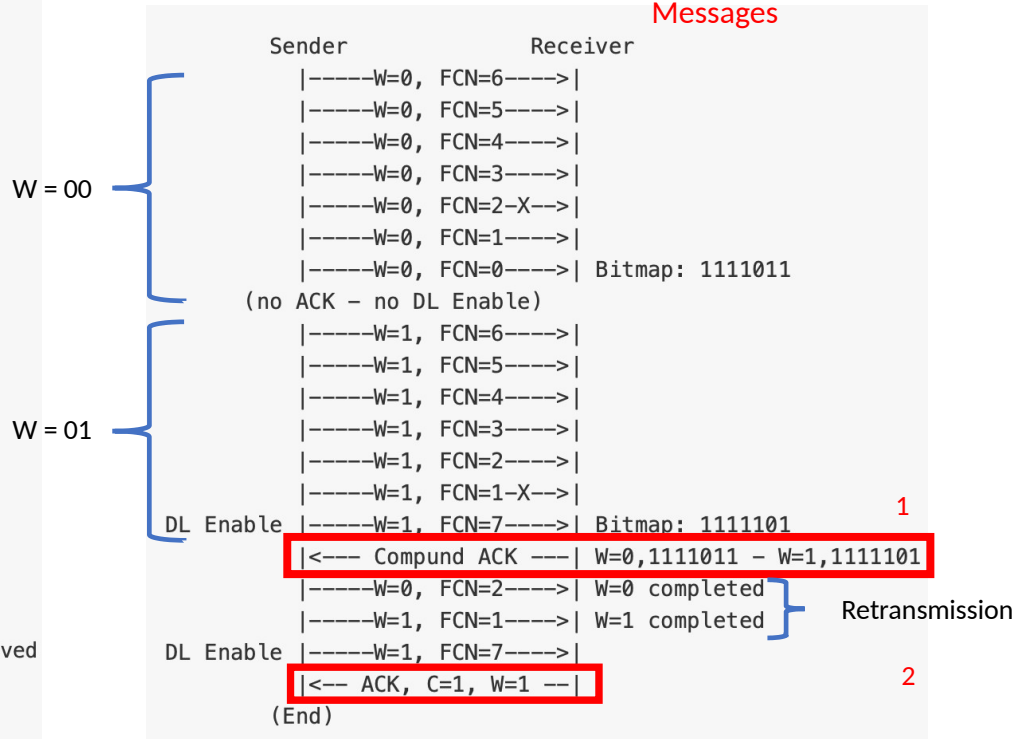
Window size: 7 tiles

3 SCHC ACK
Messages



Window size: 7 tiles

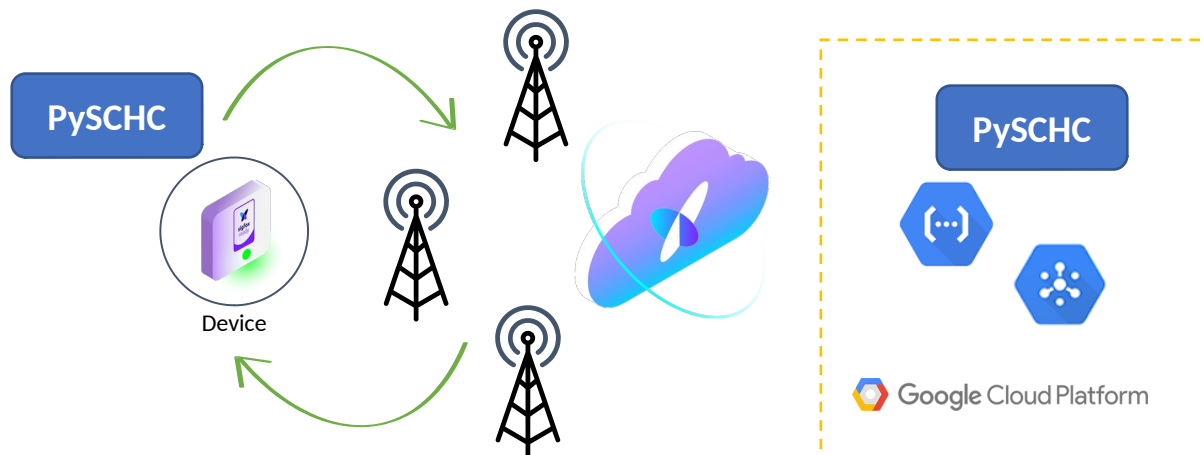
2 SCHC ACK
Messages



Retransmission

Testing Network Architecture

- PySCHC SW
- Pycom (LoPy4)
- Sigfox Network
- Google Cloud *



* <https://cloud.google.com/community/tutorials/sigfox-gw>

draft-ietf-lpwan-coap-static-context-hc-19

Ana Minaburo (ana@ackl.io)

Laurent Toutain (laurent@imt-atlantique.fr)

Ricardo Andreasen (randreasen@fi.uba.ar)

Status

- New Ballot Position Update: RFC Ed Queue
 - Status: EDIT
- Thanks to Benjamin Kaduk

From version 17 to version 19

- V-16
 - New Section 2, Uses Cases
 - New Section 6, CoAP extensions
 - New Section 4, CoAP header fields Compression
 - New Section 5, CoAP Options
- V-17 & V-18
 - New Section CoAP Options
 - Uri-Path and Uri-Query variable length unit in bytes
- V-19
 - New Security Section
 - Update the Examples for the variable length

- Many Editorial Nits

Next steps

- AUTH48... (RFC9024)?
- Questions?

draft-thubert-intarea-schc-over-ppp

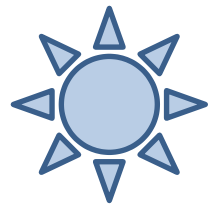
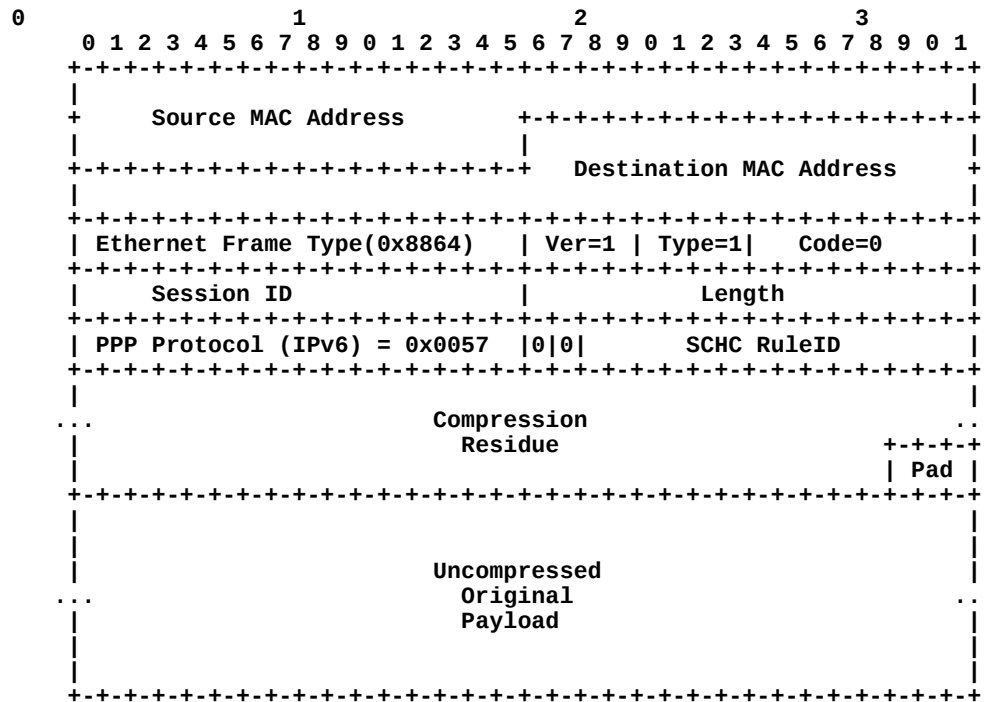
Authors:

Pascal Thubert <pthubert@cisco.com>

draft-thubert-intarea-schc-over-ppp

- SCHC over PPP (and then PPP over foo)
- Enables SCHC over
 - Serial, 3GPP
 - Ethernet with PPPoE, Wi-Fi with Ethernet
- Signals
 - A new compression for PPP (Updates RFC 5172)
 - The URL of the data model for the compression
 - Dependency on draft-ietf-lpwan-schc-yang-data-model

Resulting Packet (no Frag example)



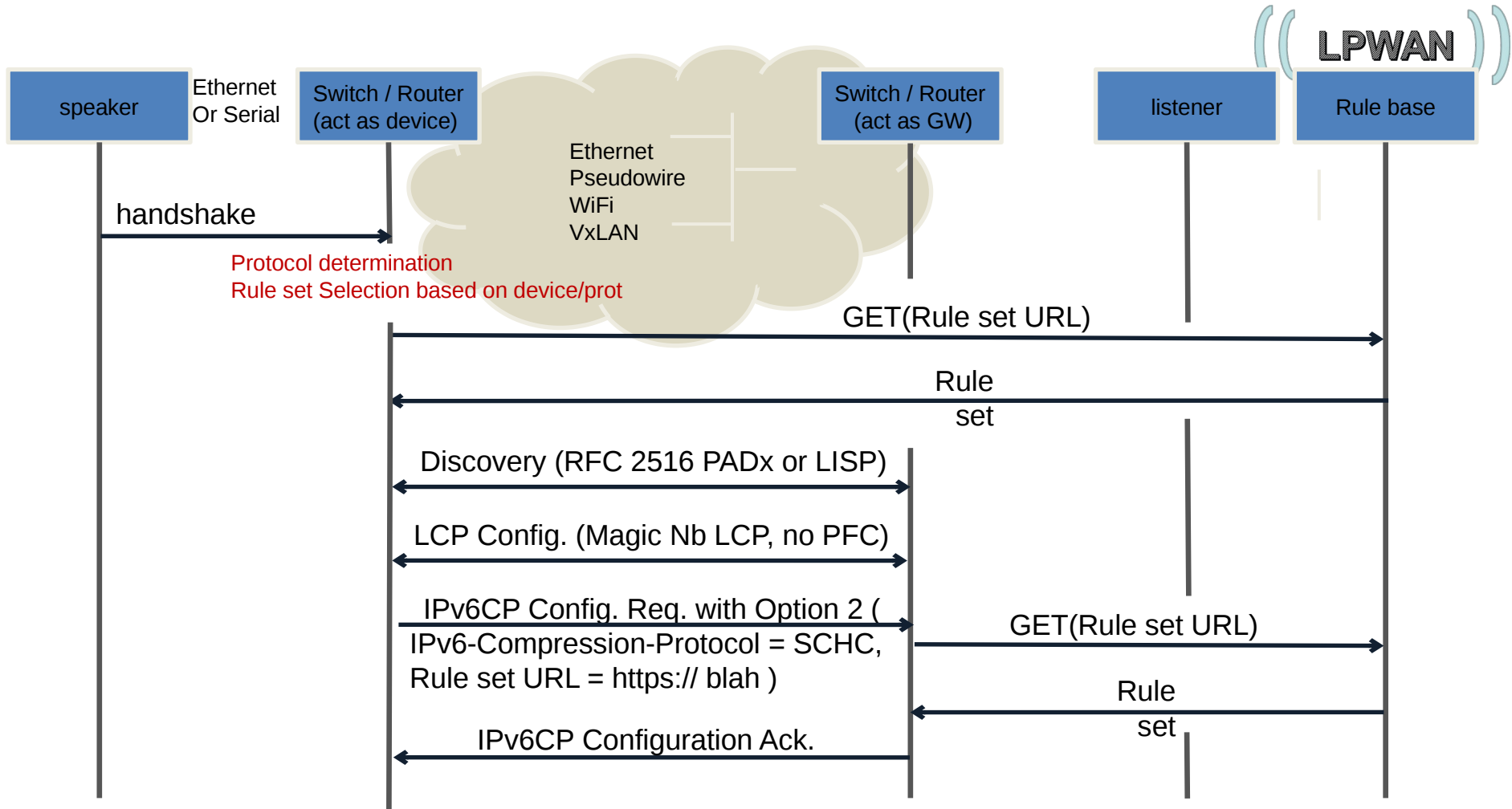
Draft status: Stable

- Need refresh – expires March 26th
- The RuleID for a compression rule is expressed as 2 bytes.
- The first (leftmost) 2 bits of that RuleId MUST be set to 0
- Frag: Only No-Ack mode => no out-of-order packet
 - If used with DetNet => may require PREOF reordering
- The RuleID for a fragmentation rule is expressed as 4 bits
 - Reserved 1111 for NO ACK
- Review from LPWAN cc INT AREA ?

Impact on SCHC Architecture



- PPP creates a peerwise relationship as opposed to device/gateway
- None of the end points own the compression rules
- Both fetch them from the same place through URL



Discussion

- Co Authors?
- Add applicability statement?
- Possible extensions?

AOB ?