draft-ietf-lpwan-schc-yang-data-model-03

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

Ana Minaburo (ana@ackl.io)

What's new

- Reference document:
  - schc@2020-06-15.yang
  - In draft -03 and github

- Changes:
  - Merged YANG data model in a single file
  - Added new identityref for SCHC parameters
  - Finished fragmentation parameters
  - Added descriptions
  - Reworked mandatory/optional fields

- Size:
  - 123 elements: 88 ids + 34 leaves

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 if 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
```
From draft OAM
```
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

- Exhaustive list of options:
  - Do not reserve space for future options
  - Anticipate upcoming options

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 dtagsize?              uint8
              |  +--rw wsize?                 uint8
              |  +--rw fcnsize                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 forbiden

Frag header, only FCN is mandatory

Mandatory: 1..max

Optional : 0..max

noAck, AA, AoE

No, Yes, Sender Choice

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                              type binary; type string;
      |  +--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
```

What's next

- Review wording and descriptions
- Check YANG Model

- Working Group Last Call