July 23rd, 2025

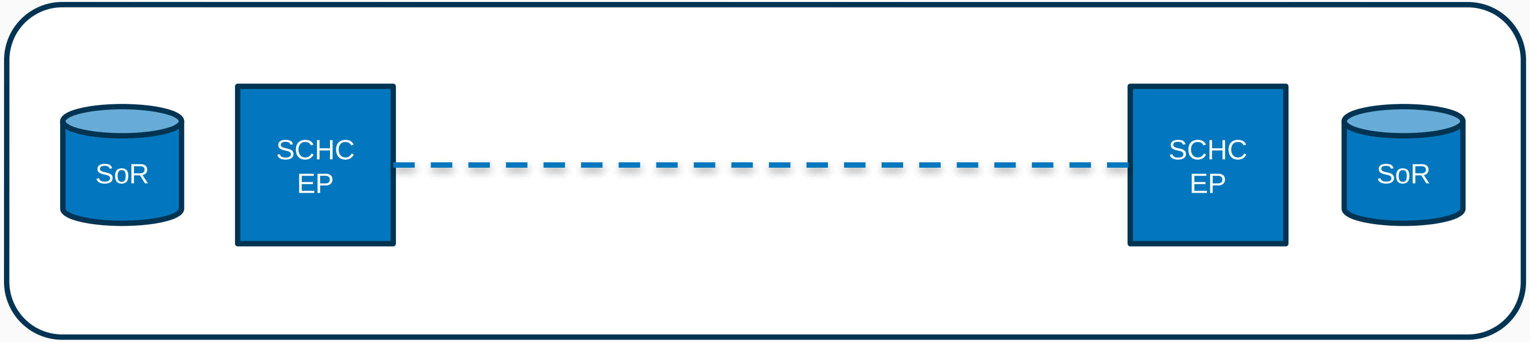
draft-toutain-schc-coreconf-management

Authors: A. Minaburo, L. Toutain, C. Banier, M. Dumay

Presenter: L. Toutain

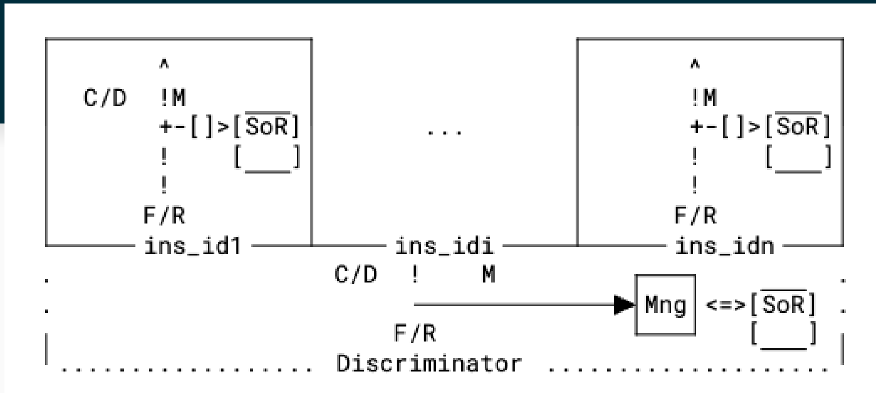
Goal

- Define management procedure inside a point-to-point SCHC instance:
  - Update, add, delete rules.
  - Set of Rules (aka context) must be identical on both end points.



SCHC Instance

Management Rules

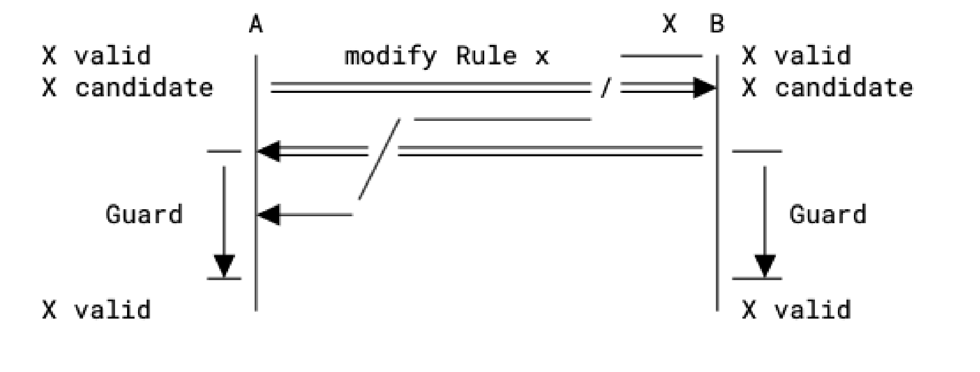


- Identical to Compression rules:
  - Only management rules can modify the SoR
  - Own IPv6/UDP/CoAP stack
    - FE80::1 <-> FE80::2, well-known port numbers
    - 2 compression rules:
      - Residue: match-list(code)+MSB(MID)
    - TBD: include OSCORE or DTLS.

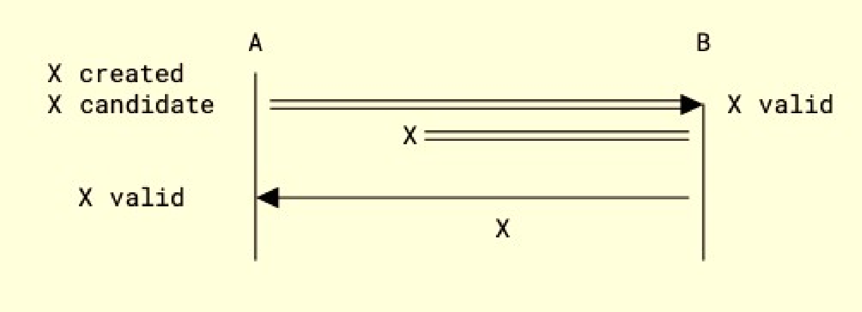
Candidate/Valid status

- The rule ID cannot be used
- SCHC PDU with this rule ID are rejected
- Candidate until Ack + guard period
  - Guard period is function of out of order messages on the link.

Rule change



Rule creation ?



conf-management

Methods Fetch

SID

```
REQ: FETCH /c
     (Content-Format: application/yang-identifiers+cbor-seq)
["target-value",        6, 3, "fid-ipv6-version", 1, "di-bidirectional"],
["matching-operator",   6, 3, "fid-ipv6-version", 1, "di-bidirectional"],
["comp-decomp-action",  6, 3, "fid-ipv6-version", 1, "di-bidirectional"]

RES: 2.05 Content
     (Content-Format: application/yang-instances+cbor-seq)
{
  {"target-value"        : [{"index" : 0, "value" : h"06"}]},
  {"matching-operator"   : "mo-equal"},
  {"comp-decomp-action"  : "cda-not-sent"}
}
```

Methods iPATCH

```
iPATCH /c
{
  ["entry", 7, 8, field, 1, "di-bidirectional"] : {
      delta_target-value        : [{delta_index : 0, delta_value : value}],
      delta_matching-operator   : "mo-equal",
      delta_comp-decomp-action : "cda-not-sent"
  }
}
```

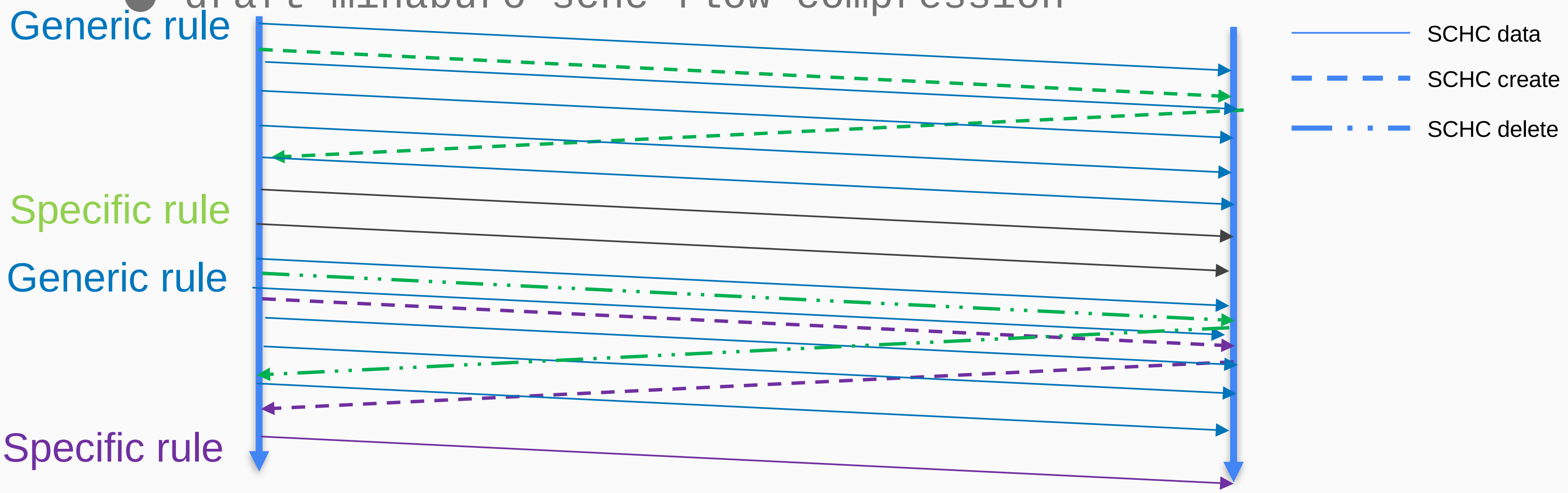
Methods POST (for RPCs)

Limits RTT

```
rpcs:
  +---x duplicate-rule
     +---w input
     |  +---w from
     |  |  +---w rule-id-value?    uint32
     |  |  +---w rule-id-length?   uint8
     |  +---w to
     |  |  +---w rule-id-value?    uint32
     |  |  +---w rule-id-length?   uint8
     |  +---w ipatch-sequence?   binary
     +--ro output
        +--ro status?   string
```

Flow compression

- `draft-minaburo-schc-flow-compression`



Error management

There is different level of error detection:

- COREECONF Errors: these error are directly generated at the CORECONF level. For instance, retrieving a value with a wrong key.
- YANG validation errors: the data model is not conforming with the constrains such as "must" or "mandatory". **This check is optional**, since it may require a lot of resources on a device.
- SCHC errors: Errors on the Data Model that cannot be detected at the YANG level, for example, the rule numbering does not respect a binary tree.

Conclusion

- Work in progress
- Implemented in openSCHC and microSCHC
- Draft contains more examples