CoRE: CoRECONF

- RFC 9254: YANG-CBOR
- RFC-Editor (EDIT, 9 weeks): CORE-SID
- WGLC passed CORE-COMI
- WGLC passed CORE-YANG-LIBRARY

- (submission to CBOR WG):
  draft-bormann-cbor-yang-standin-00
core-sid: Implementation Status

— core-sid -24 2023-12-22, approved 2024-01-17
— Remaining PYANG work started at IETF 118 Hackathon
— Message (in-flight, not at-rest) data items in YANG?
— Little support in implementations
— More discussion about status at IETF 119
Base:
RFC 9254: YANG-CBOR
draft-ietf-core-sid: Management of SID space

CoRECONF = YANG/CBOR over CoAP
RESTCONF = YANG/* over HTTP
NETCONF = YANG/XML over SSH
COMI: Status

- Comi -17 2024-03-04
  - Clarification: addresses Unified Datastore only now
    - explicitly mention all-or-none semantics
    - potential discussion about future "candidates" feature
  - Fix RPC/Action examples: no redundant nesting
  - Editorial fixes
    - May need to clarify FETCH request/response pairing further
Koen Zandberg: Recent Implementation effort went well

Can simplify CoRECONF further

— Get rid of "datastore resource" GET/PUT
can do FETCH/iPATCH of "SID 0"
— [x] Semantics of multiple RPC/Actions in one payload?
— Get rid of term "data node resource" and § 5.2.2
Andy Bierman (coauthor)

— [#14] (ed.) Add examples for each media type
— [x] Clarify that the spec is for a unified data store (can't use NMDA as is)
— Doubts about simplification of instance identifiers in response
— [#15] Possibly allow a filter parameter like "depth" in RESTCONF
— [x] Should provide all-or-none semantics
— [x], [#16] Editorial comments on examples
— [x] Remove extra layer of 0 in RPC/action responses
— [x] (ed.) clarify that appendices are normative
~May 2024:
Get remaining comments addressed
(and further examples made)

— Probably another WGLC then
CoRE: CRIs (HREF)

-14 (2024-01-09): address reviews mostly
  - Added section about CoAP integration (complement 7252)
  - Added EDN cri'...' notation

To do:
  - #77 more test vectors. More test vectors. (#52, #53)
  - Make URL scheme registry non-negative (for CoAP uint)
  - #82 Clarify determinism objective (CRI: yes, CRI reference: no)
8. Using CRIs with CoAP (new in -14)

8.1. Converting CoAP CRIs ↔ Sets of CoAP Options
- Analogue to Sections 6.4 and 6.5 of [RFC7252]

8.2. CoAP Options for Forward-Proxies
- Proxy-Uri ➔ Proxy-Cri
- Proxy-Scheme ➔ Proxy-Scheme-Number
### 8.2.1. Proxy-CRI

<table>
<thead>
<tr>
<th>No.</th>
<th>C</th>
<th>U</th>
<th>N</th>
<th>R</th>
<th>Name</th>
<th>Format</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD235</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Proxy-Cri</td>
<td>opaque</td>
<td>1-1023</td>
<td>(none)</td>
</tr>
</tbody>
</table>

*Table 1: Proxy-Cri CoAP Option*

- CoAP opaque ➔ CRI as encoded CBOR item
- Proxy-Cri overrides Proxy-Uri
8.2.2. Proxy-Scheme-Number

<table>
<thead>
<tr>
<th>No.</th>
<th>C</th>
<th>U</th>
<th>N</th>
<th>R</th>
<th>Name</th>
<th>Format</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD239</td>
<td>x</td>
<td>x</td>
<td>-</td>
<td></td>
<td>Proxy-Scheme-Number</td>
<td>uint</td>
<td>0-3</td>
<td>(none)</td>
</tr>
</tbody>
</table>

Table 2: Proxy-Scheme-Number CoAP Option

— uint ➔ Need unsigned integer (no CBOR encoding)

➔ go for unsigned integer (uint) URL scheme numbers
  • use uint for CoAP directly
  • 1's complement (-1 - x) for CRI scheme-id (nint)
HREF: Plan

Get those test vectors in place
• (edit them in CSV: PR#79)

Do the todos

Complete I-D in ~May 2024