CoAP Diagnostic Message Notation

draft-jimenez-core-coap-diag-01

Jaime Jiménez, **Carsten Bormann**

IETF 127, San Francisco, 18 November 2026

The Problem

- CoAP drafts/RFCs notate exchanges in many ways: split vs composed URIs, Header:/Payload: vs Req:/Res: vs nothing
- No clear convention, more like variations on a scheme
- No accepted **machine-checkable** format
  ➳ malformed CBOR/JSON examples happen

The Notation

- **First line**: (request) method + composed URI, or response code + reason phrase
- **Options**: `Name:` value pairs (RFC 9203 style)
- **Content-Format**: code point **and** media type
  ➳ tells tooling which validator to use
- **Payload**: CDN for CBOR, JSON, CoRE Link Format, or hex
- Blank line separates options from payload (HTTP style)

Examples in the draft

Example		Example	
GET, text payload	*00*	Broker discovery	*00*
POST creating a topic	*00*	ACE token request	*00*
FETCH, CBOR filter	*00*	Error response	*00*
iPATCH partial update	*00*	DNS over CoAP	**01**
Observe subscription	*00*	Same query, dns+cbor	**01**
Transport annotations	*00*	Survey Variants 1-7, 8	**01**

Payload representations (-01)

Request

```
FETCH coaps://[2001:db8::1]/
Content-Format: 553 (application/dns-message)
Accept: 553 (application/dns-message)

h'0000 0100 0001 0000 0000 0000
  07 6578616d706c65 03 6f7267 00
  001c 0001'

Payload (text):
;; QUESTION SECTION:
;example.org.        IN      AAAA
```

Response

```
2.05 Content
Content-Format: 553 (application/dns-message)
Max-Age: 58719

Payload (text):
;; QUESTION SECTION:
;example.org.        IN      AAAA
;; ANSWER SECTION:
;example.org. 79689  IN      AAAA
  2001:db8:1:0:1:2:3:4
```

Observe notifications (-02)

Registration

```
GET /ps/data/1bd0d6d (Token=0x4a)
Observe: 0

2.05 Content (Token=0x4a)
Observe: 1234
Content-Format: 60 (application/cbor)

{0: "living-room-sensor", 2: 22.5}
```

Later notifications

```
2.05 Content (Token=0x4a)
Observe: 1235
Content-Format: 60 (application/cbor)

{0: "living-room-sensor", 2: 22.7}

2.05 Content (Token=0x4a)
Observe: 1236
Content-Format: 60 (application/cbor)

{0: "living-room-sensor", 2: 23.1}
```

multipart-core (-02)

Opaque parts

```
2.05 Content
Content-Format: 62 (application/multipart-core)

[60, h'a164 74656d70 f94d60',
  0, h'6c6976696e6720726f6f6d']
```

Parts in Concise Diagnostic Notation

```
2.05 Content
Content-Format: 62 (application/multipart-core)

[60, <<{"temp": 21.5}>>,   # application/cbor
  0, 'living room']        # text/plain
```

OSCORE (RFC 8613) (-02)

Application view

```
GET /alarm_status (OSCORE, kid=0x5f)

2.05 Content (OSCORE)
Content-Format: 0 (text/plain)

0
```

RFC 8613 Figure 12

```
Code: 0.02 (POST)
Token: 0x8c
OSCORE: [kid:5f, Partial IV:42]
Payload: {Code:0.01,
          Uri-Path:"alarm_status"}
```

🦆 Review by Christian Amsüss

https://mailarchive.ietf.org/arch/msg/core/vWb3ekSMAJknuKrGXTkK9BsCp0o
2026-11-18T20:16 mail on core@, which see

- Use Location pseudo-option to combine Location-* in response
- Be explicit about Uri-* vs. Proxy-URI vs. Proxy-Scheme vs. Proxy-Scheme-Number vs. Proxy-CRI (using CBOR DN?)
- Work around long-standing problem showing bytes in hex
  - Maybe use option registration to distinguish:
    • empty, • uint, • string (text), • opaque (bytes)
  - Doesn't work with payload

How to show data in hex

- Bad RFC 8949 convention for encoded data items: `0x123456`
- vs. CBOR DN for an actual CBOR byte string ("wrapped"): `h'123456'`
- Might simply use hex: `123456` or `12 34 56`

vs. JSON, link-format, text/plain, other actual text

Make sure we can do automated checking (what is »12«?)

Open for -02

- Are -00 and -01 examples sufficient?
- Should we add the Multipart and OSCORE examples?
- Proxy-Scheme with Uri-*, signaling 7.xx, Block2 TBD
- Any other thoughts/feedback?

Next steps: -02 onward

- **Adopt** as a WG draft; no rush
- Obtain more **implementation experience**
  - diagnostic output by CoAP implementations
  - kramdown-rfc checker, other CI tools
- **Validate by use** in drafts
  - Make sure notation is **usable**
  - Don't be too **inventive**, though
- Work with a **living I-D** for a while
- When good **coverage**: go Standards-Track (like notation specs do)