

A Concise Binary Object Representation (CBOR) of DNS Messages

[draft-lenders-dns-cbor-01]

Martine S. Lenders, Thomas C. Schmidt, Matthias Wählisch

IETF 115 CBOR Meeting, 2022-11-10

Motivation

Objectives and Definition

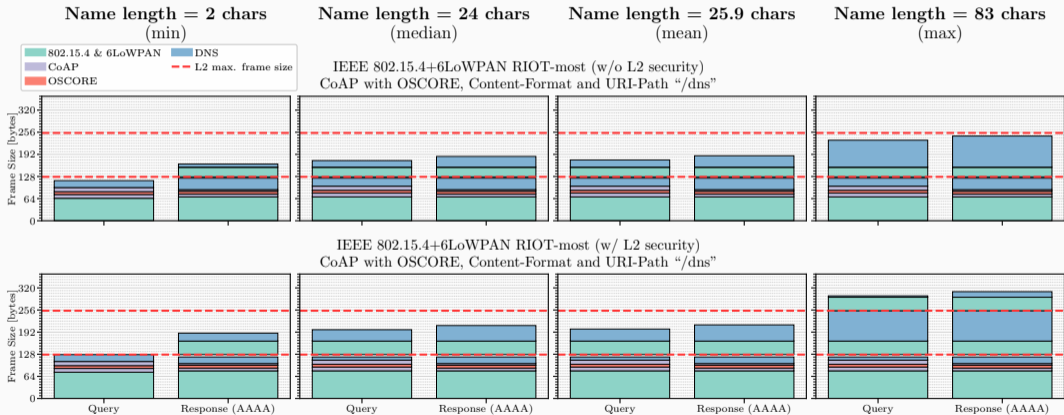
Feedback from DNSOP in CoRE discussions

Next Steps for draft-lenders-dns-over-coap

Motivation: DNS in Constrained Networks

Packet size exceeds 802.15.4 PDU depending on queried name length

⇒ Fragmentation



DNS over CoAP (draft-ietf-core-dns-over-coap) messages for different name lengths

Motivation: DNS in Constrained Networks

Packet size exceeds 802.15.4 PDU depending on queried name length

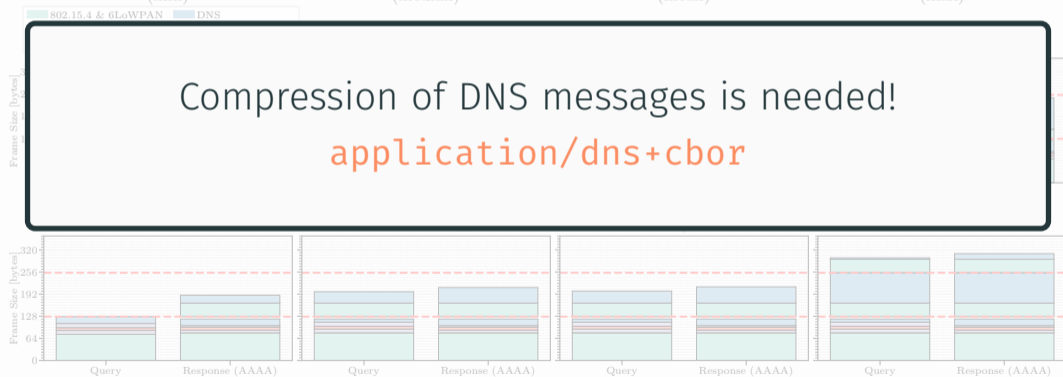
⇒ Fragmentation

Name length = 2 chars
(min)

Name length = 24 chars
(median)

Name length = 25.9 chars
(mean)

Name length = 83 chars
(max)



DNS over CoAP (draft-ietf-core-dns-over-coap) messages for different name lengths

Objectives of draft-lenders-dns-cbor (application/dns+cbor)

Reduce packet sizes of DNS queries and replies:

1. Encoding of DNS messages in CBOR
2. Omit (redundant) DNS fields in DNS queries and responses
3. Address and name compression using packed CBOR (optional)

- Concise format might hamper future DNS extensions
- Possible ways to address:
 - Allow for unstructured resource records as byte string
 - Content negotiation
 - e.g., `application/dns-message` as fallback is not off the table
 - ⟨Your thoughts?⟩

Using CDDL (RFC8610)

```
domain-name = tstr
type-spec = (
  record-type: uint,
  ? record-class: uint,
)
dns-question = (
  ? id: uint,
  name: domain-name,
  ? type-spec,
)
dns-query = [dns-question]
```

CBOR array:

- At minimum containing text string domain name (IDNA encoded)
- Optional ID and record type specification (ID defaults to 0, record-type to AAAA, record-class to IN)

CoRE & DNSOP discussions:

- Does not consider additional section for EDNS(0) pseudo-RRs
- How to express DNS Stateful Operations (DSO)?
 - Opcode = 6
 - All section counts = 0
 - TLVs after traditional sections


```
rr = (  
  ? name: domain-name,  
  ttl: uint,  
  ? type-spec,  
  rdata: bstr / domain-name,  
)  
dns-rr = [rr]
```

CBOR array:

- At minimum containing TTL and resource data
- Optional name and record type specification (both default to question values)

CoRE & DNSOP discussions:

- `dns-rr = [rr] / bstr` for potential future DNS extensions?

```
extra-sections = (  
  ? authority: [+ dns-rr],  
  additional: [+ dns-rr]  
)  
sections = ((  
  ? id: uint,  
  answer: [+ dns-rr]  
) // (  
  ? id: uint,  
  question: dns-query,  
  answer: [+ dns-rr],  
  ? extra-sections,  
)  
)  
dns-response = [sections]
```

CBOR array of arrays:

- At minimum containing answer section (array of DNS resource records)
- **Generally assumes that transport can map query to response!** (original question and ID may be amended optionally)

CBOR array of arrays:

```
extra-sections = (  
  ? authority: [+ dns-rr],  
  ? additional: [+ dns-rr],  
)
```

CoRE & DNSOP discussions:

- DNS Stateful Operations again missing
- Fields completely ignored at the moment:
Opcode, Rcode, Flags

```
section: {  
  ? id: uint,  
  answer: [+ dns-rr],  
  // ( ? id: uint, ? rcode: int, ? flags: int )  
  question: dns-query,  
  answer: [+ dns-rr],  
  ? extra-sections,  
})
```

```
dns-response = [sections]
```

- At minimum containing answer section (array of DNS resource records)
- Sections that transport additional data (e.g. response ID may be amended optionally)

Simple Example

Query IPv6 address for `example.org`

(13 bytes vs. 52 bytes wire-format: compression 400%)

```
["example.org"]
```

Corresponding response (24 bytes vs. 68 bytes wire-format: compression 283.3%):

```
[[[3600, h'20010db8000000000000000000000001']]]
```

A More Complex Example

Query ANY record for `example.org` (cf. DNS-SD)

(17 bytes vs. 52 bytes wire-format: compression 305,9%)

```
["example.org", 255, 255]
```

Corresponding response (200 bytes vs. 195 bytes wire-format: compression 97.5%):

```
[  
  ["example.org", 12, 1],  
  [[3600, "_coap._udp.local"]],  
  [[3600, 2, "ns1.example.org"], [3600, 2, "ns2.example.org"]],  
  [  
    ["_coap._udp.local", 3600, 28, h'20010db8000000000000000000000001'],  
    ["ns1.example.org", 3600, 28, h'20010db8000000000000000000000035'],  
    ["ns2.example.org", 3600, 28, h'20010db800000000000000000000003535']  
  ]  
]
```

A More Complex Example

Query ANY record for `example.org` (cf. DNS-SD)

(17 bytes vs. 52 bytes wire-format: compression 305,9%)

```
["example.org", 255, 255]
```

Corresponding response (200 bytes vs. 195 bytes wire-format: compression 97.5%):

```
[  
  ["example.org", 12, 1],  
  [[3600, "_coap._udp.local"]],  
  [[3600, 2, "ns1.example.org"], [3600, 2, "ns2.example.org"]],  
  [  
    ["_coap._udp.local", 3600, 28, h'20010db8000000000000000000000001'],  
    ["ns1.example.org", 3600, 28, h'20010db8000000000000000000000035'],  
    ["ns2.example.org", 3600, 28, h'20010db800000000000000000000003535']  
  ]  
]
```

⇒ **Larger than wire-format!** Address and name compression needed

A More Complex Example

Query ANY record for `example.org` (cf. DNS-SD)

(17 bytes vs. 52 bytes wire-format: compression 305,9%)

```
["example.org", 255, 255]
```

Corresponding response (200 bytes vs. 195 bytes wire-format: compression 97.5%):

```
[  
  ["example.org", 13, 1],  
  [[3600, "_coap._udp.local"],  
   [[3600, 2, "ns1", 3600, 28, h'20010db8000000000000000000000001'],  
    ["_coap._udp.local", 3600, 28, h'20010db8000000000000000000000001'],  
    ["ns1.example.org", 3600, 28, h'20010db80000000000000000000000035'],  
    ["ns2.example.org", 3600, 28, h'20010db8000000000000000000000003535']  
  ]  
]
```

⇒ **Larger than wire-format!** Address and name compression needed

Our Proposal: Name and Address Compression Using Packed CBOR

- Optional packed CBOR support *for responses* negotiated using media type parameter `packed=1` (own media type in draft -01)
- Make shared value and argument tables one list for that media type

`compr-dns-response = any # TBD; how to express packed CBOR in CDDL?`

`packed-dns-response = [[pack-table], compr-dns-response]`

`pack-table = any`

Response becomes another CBOR array of two arrays:

1. Packing table (combined shared value and argument table)
2. Compressed `dns-response`
(structure as defined before: CBOR array of sections)

Example: ANY Record Response in application/dns+cbor;packed=1

Original CBOR response (200 bytes)

```
[["example.org", 12, 1],  
 [[3600, "_coap._udp.local"]],  
 [[3600, 2, "ns1.example.org"],  
 [3600, 2, "ns2.example.org"]],  
 [["_coap._udp.local", 3600, 28,  
  h'20010db8000000000000000000000001'],  
 ["ns1.example.org", 3600, 28,  
  h'20010db8000000000000000000000035'],  
 ["ns2.example.org", 3600, 28,  
  h'20010db800000000000000000000003535']]
```

Packed CBOR response (119 bytes)

```
[[h'20010db8000000000000000000000000',  
  "_coap._udp.local", "example.org",  
  3600, 28, 2  
],  
 [[simple(2), 12, 1],  
  [[simple(3), simple(1)]],  
  [[simple(2), simple(5), 219("ns1.")],  
   [simple(2), simple(5), 219("ns2.")]],  
  [[simple(1), simple(3), simple(4), 6(h'0001')],  
   [219("ns1."), simple(3), simple(4), 6(h'0035')],  
   [219("ns2."), simple(3), simple(4), 6(h'3535')]  
]]
```

(*cmp.* 195 bytes wire-format: compression 163.9%)

Example: ANY Record Response in application/dns+cbor;packed=1

Original CBOR response (200 bytes)

```
[["example.org", 12, 1],  
  [[3600, "_coap._udp.local"]],  
  [[3600, 2, "ns1.example.org"],  
   [3600, 2, "ns2.example.org"]],  
  ["_coap._udp.local", 3600, 28,  
   h'20010db8000000000000000000000001'],  
  ["ns1.example.org", 3600, 28,  
   h'20010db8000000000000000000000035'],  
  ["ns2.example.org", 3600, 28,  
   h'20010db800000000000000000000003535']]
```

Packed CBOR response (119 bytes)

```
[[h'20010db8000000000000000000000000',  
  "_coap._udp.local", "example.org",  
  3600, 28, 2  
],  
 [[simple(2), 12, 1],  
  [[simple(3), simple(1)]],  
  [[simple(2), simple(5), 219("ns1.")],  
   [simple(2), simple(5), 219("ns2.")]],  
  [[simple(1), simple(3), simple(4), 6(h'0001')],  
   [219("ns1."), simple(3), simple(4), 6(h'0035')],  
   [219("ns2."), simple(3), simple(4), 6(h'3535')]  
]]]
```

(*cmp.* 195 bytes wire-format: compression 163.9%)

Implied DNS-specific table entries for global compression contexts (e.g. TLDs)
enable potential for more elision

Next Steps

- Back to the drawing board?
 - Major format change needed for EDNS(0)/DSO etc.
 - Or keep as is (with minor changes)? Use `application/dns-message` as fallback
- Define more details on using packed CBOR:
 - How to construct packing table?
 - Global compression contexts, DNS-specific implied table entries
 - ⟨Your thoughts.⟩