Authors: A. Rundgren, Ed.
        Independent

CBOR Object Type Extension (COTX)

Abstract

   This document describes a CBOR tag for augmenting CBOR data items
   with type identifiers in the form of arbitrary CBOR text strings.
   This design enables type identifiers to optionally be expressed as
   URLs, potentially pointing to Web pages holding related descriptions
   in human readable form, as well as being compatible with established
   methods for adding type information to JSON and XML data.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six
   months and may be updated, replaced, or obsoleted by other documents
   at any time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on 6 March 2024.

Table of Contents

1. Introduction

This specification introduces a method for augmenting data expressed
in the CBOR [RFC8949] notation, with a type identifier mechanism
based on CBOR text strings.

The primary purposes of the text based type identifier tag described
in this document are:

   *Enabling developers defining application specific type
    identifiers without necessarily having to go through an external
    registration process.

   *By supporting URLs [URL] as type identifiers, related human
    readable information may (through dereferencing), be provided for
    usage with Web browsers. Since URLs are compatible with firmly
    established methods for adding type information to JSON and XML
    data, this design may simplify a switch to CBOR. See also
    Appendix A.

This specification is also intended to provide a path for ISO using
CBOR as a possible alternative to XML by supporting their current
URN [RFC8141] based type identifier naming scheme. See also
Appendix B.

By applying the typing scheme to top level CBOR objects, additional
functionality is enabled including:

  *Support for embedding CBOR objects in other CBOR and non-CBOR
   constructs, as well as storage in databases, without being forced
   adding information about the object.

  *Remove the need for application specific media types. In many
   cases "application/cbor" would suffice.

1.1. Terminology

In this document the term CBOR "object" is used interchangeably with
the CBOR [RFC8949] "data item".

The key words **"MUST"**, **"MUST NOT"**, **"REQUIRED"**, **"SHALL"**, **"SHALL NOT"**,
"SHOULD", **"SHOULD NOT"**, **"RECOMMENDED"**, **"NOT RECOMMENDED"**, **"MAY"**, and
"OPTIONAL" in this document are to be interpreted as described in
BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all
capitals, as shown here.

2. Specification

This specification builds on the CBOR [RFC8949] tag feature (major
type 6), by defining a fixed tag with the preliminary decimal value
of 1010. See also Section 3.

This tag MUST in turn enclose a CBOR array (major type 4) with two
elements, where the first element MUST contain a type identifier in
textual format indicating the definition of a CBOR object, while the
second element MUST hold an instance of the associated object
itself. The type identifier MUST be a valid CBOR text string (major
type 3), while the only constraint on the targeted object is that it
MUST be a valid CBOR object.

The syntax expressed in CBOR diagnostic notation (section 8 of
[RFC8949]) would read as:

    1010([*Object Type Identifier*, *Object Instance Data*])

Note that real-world usages will typically impose constraints like
requiring type identifiers to be expressed as HTTPS URLs etc.

2.1. Sample

Consider the following sample:

```
1010(["https://example.com/myobject", {
  1: "data",
  2: "more data"
}])
```

Converting the sample above to CBOR expressed in hexadecimal
notation (here shown with embedded comments as well), should result
in the following output:

```
D9 03F2                             # tag(1010)
   82                               # array(2)
      78 1C                         # text(28)
         68747470733A2F2F6578616D706C652E636F6D2F6D796F626A656374
                                    # "https://example.com/myobject"
      A2                            # map(2)
         01                         # unsigned(1)
         64                         # text(4)
            64617461                # "data"
         02                         # unsigned(2)
         69                         # text(9)
            6D6F72652064617461      # "more data"
```

In a typical implementation "https://example.com/myobject" would
also serve as a hyper-link to human readable information about the
identifier, accessed through a Web browser.

3. IANA Considerations

In the registry [IANA.cbor-tags], IANA is requested to allocate the
tag defined in Table 1.

Tag	Data Item	Semantics	Reference
1010	array: [id: text string, obj: any]	Object type identifier	draft-rundgren-cotx-04

Table 1: Values for Tag Numbers

This request has been granted.

4. Security Considerations

This specification inherits all the security considerations of CBOR
[RFC8949].

URL-based type identifiers MUST NOT be used for automatically
downloading CBOR schema data like CDDL [RFC8610] to CBOR processors,
since this introduces potential vulnerabilities.

The availability of type information does in no way limit the need
for input data validation.

For signed CBOR objects, it is RECOMMENDED to include the object type identifier extension in the signature calculation as well. The same considerations apply to encryption using AEAD algorithms.

5. References

5.1. Normative References

[IANA.cbor-tags] IANA, "Concise Binary Object Representation (CBOR) Tags", 19 September 2013, <https://www.iana.org/assignments/cbor-tags>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/info/rfc8949>.

5.2. Informative References

[RFC4151] Kindberg, T. and S. Hawke, "The 'tag' URI Scheme", RFC 4151, DOI 10.17487/RFC4151, October 2005, <https://www.rfc-editor.org/info/rfc4151>.

[RFC5141] Goodwin, J. and H. Apel, "A Uniform Resource Name (URN) Namespace for the International Organization for Standardization (ISO)", RFC 5141, DOI 10.17487/RFC5141, March 2008, <https://www.rfc-editor.org/info/rfc5141>.

[RFC8141] Saint-Andre, P. and J. Klensin, "Uniform Resource Names (URNs)", RFC 8141, DOI 10.17487/RFC8141, April 2017, <https://www.rfc-editor.org/info/rfc8141>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/info/rfc8610>.

[URL] What WG, "Living Standard — Last Updated 3 May 2022", <https://url.spec.whatwg.org/>.

[XSD]
> W3C, "XML Schema Definition Language (XSD) 1.1 Part 1: Structures", <http://www.w3.org/TR/xmlschema11-1/>.

Appendix A. URI and URL Identifiers

The primary reason for using URI or URL [URL] based type identifiers is for maintaining a single name-space for the entire specification of a system. Note that the referenced URL specification does not distinguish between URIs and URLs.

This non-normative section describes different methods for dealing with type identifiers expressed as URIs or URLs.

A.1. Registering a Dedicated Domain

A core issue with identifiers depending on domain (DNS) names is that domain names may not necessarily remain valid during the anticipated life-time of an identifier. The owner of a domain name may due to organizational changes, neglect, lack of interest, or even death, lose control over its use, effectively leaving associated identifiers orphaned.

A.2. Using a Sub-domain

An alternative is using a dedicated sub-domain belonging to an entity that is likely to survive for the foreseeable future. With the advent of public repositories like GitHub, this appears to be a simpler, cheaper, and more robust solution than maintaining dedicated domain names.

A.3. The 'tag' URI Scheme

For applications where strict control over the name-space is hard to achieve, the 'tag' URI scheme [RFC4151] may be used.

Appendix B. URN Identifiers

ISO currently use URN [RFC8141] [RFC5141] based type identifiers like "urn:iso:std:iso:20022:tech:xsd:pain.001.001.10" for data definitions using XML schema [XSD]. This method could be applied to CBOR and CDDL [RFC8610] as well.

Acknowledgements

People who have contributed with valuable feedback to this specification include Christian Amsüss, Carsten Bormann, and Joe Hildebrand.

Document History

Version 00:

  *Initial publication.

Version 01:

  *IANA reference update.

Version 02:

  *Made type identifier a CBOR text string.

  *Wordsmithing.

Version 03:

  *Cleaner abstract and intro.

Version 04:

  *Type identifier throughout the spec.

Version 05:

  *IANA have granted the regitration.

Author's Address

Anders Rundgren (editor)
Independent
Montpellier
France

Email: anders.rundgren.net@gmail.com
URI: https://www.linkedin.com/in/andersrundgren/