

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: 24 April 2022

A. Keränen  
Ericsson  
C. Bormann  
Universität Bremen TZI  
21 October 2021

SenML Data Value Content-Format Indication  
draft-ietf-core-senml-data-ct-07

## Abstract

The Sensor Measurement Lists (SenML) media type supports multiple types of values, from numbers to text strings and arbitrary binary data values. In order to facilitate processing of binary data values, this document specifies a pair of new SenML fields for indicating the content format of those binary data values, i.e., their Internet media type including parameters as well as any content codings applied.

## 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 24 April 2022.

## Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

|                      |   |                    |
|----------------------|---|--------------------|
| <a href="#">1.</a>   | Introduction . . . . .                            | <a href="#">2</a>  |
| <a href="#">1.1.</a> | Evolution . . . . .                               | <a href="#">3</a>  |
| <a href="#">2.</a>   | Terminology . . . . .                             | <a href="#">4</a>  |
| <a href="#">3.</a>   | SenML Content-Format ("ct") Field . . . . .       | <a href="#">5</a>  |
| <a href="#">4.</a>   | SenML Base Content-Format ("bct") Field . . . . . | <a href="#">6</a>  |
| <a href="#">5.</a>   | Examples . . . . .                                | <a href="#">6</a>  |
| <a href="#">6.</a>   | ABNF . . . . .                                    | <a href="#">7</a>  |
| <a href="#">7.</a>   | Security Considerations . . . . .                 | <a href="#">8</a>  |
| <a href="#">8.</a>   | IANA Considerations . . . . .                     | <a href="#">8</a>  |
| <a href="#">9.</a>   | References . . . . .                              | <a href="#">9</a>  |
| <a href="#">9.1.</a> | Normative References . . . . .                    | <a href="#">9</a>  |
| <a href="#">9.2.</a> | Informative References . . . . .                  | <a href="#">10</a> |
|                      | Acknowledgements . . . . .                        | <a href="#">11</a> |
|                      | Authors' Addresses . . . . .                      | <a href="#">11</a> |

## [1.](#) Introduction

The Sensor Measurement Lists (SenML) media types [[RFC8428](#)] can be used to send various kinds of data. In the example given in Figure 1, a temperature value, an indication whether a lock is open, and a data value (with SenML field "vd") read from an NFC reader is sent in a single SenML pack. The example is given in SenML JSON representation, so the "vd" (data value) field is encoded as a base64url string (without padding), as per [Section 5 of \[RFC8428\]](#).

```
[
  {"bn":"urn:dev:ow:10e2073a01080063:","n":"temp","u":"Cel","v":7.1},
  {"n":"open","vb":false},
  {"n":"nfc-reader","vd":"aGkgCg"}
]
```

Figure 1: SenML pack with unidentified binary data

The receiver is expected to know how to interpret the data in the "vd" field based on the context, e.g., name of the data source and out-of-band knowledge of the application. However, this context may not always be easily available to entities processing the SenML pack,

especially if the pack is propagated over time and via multiple entities. To facilitate automatic interpretation it is useful to be able to indicate an Internet media type and, optionally, content codings right in the SenML Record.

The CoAP Content-Format ([Section 12.3 of \[RFC7252\]](#)) provides this information in the form of a single unsigned integer; enclosing a Content-Format number (in this case number 60 as defined for content-type application/cbor in [\[RFC8949\]](#)) in the Record is illustrated in Figure 2. All registered CoAP Content-Format numbers are listed in the COAP Content-Formats registry [[IANA.core-parameters](#)] as specified by [Section 12.3 of \[RFC7252\]](#). Note that, at the time of writing, the structure of this registry only provides for zero or one content codings; nothing in the present document needs to change if the registry is extended to allow sequences of content codings.

```
{"n":"nfc-reader", "vd":"gmNmb28YKg", "ct":"60"}
```

Figure 2: SenML Record with binary data identified as CBOR

In this example SenML Record, the data value contains a string "foo" and a number 42 encoded in a CBOR [\[RFC8949\]](#) array. Since the example above uses the JSON format of SenML, the data value containing the binary CBOR value is base64-encoded ([Section 5 of \[RFC4648\]](#)). The data value after base64 decoding is shown with CBOR diagnostic notation in Figure 3.

```
82          # array(2)
  63          # text(3)
    666F6F  # "foo"
  18 2A      # unsigned(42)
```

Figure 3: Example Data Value in CBOR diagnostic notation

### [1.1.](#) Evolution

As with SenML in general, there is no expectation that the creator of a SenML pack knows (or has negotiated with) each consumer of that pack, which may be very remote in space and particularly in time. This means that the SenML creator in general has no way to know whether the consumer knows:

- \* each specific media-type-name used
- \* each parameter and each parameter value used
- \* each content coding in use

- \* each Content-Format number in use for a combination of these

What SenML, as well as the new fields defined here, guarantees is that a recipient implementation *\_knows\_* when it needs to be updated to understand these field values and the values controlled by them; registries are used to evolve these name spaces in a controlled way. SenML packs can be processed by a consumer while not understanding all the information in them, and information can generally be preserved in this processing such that it is useful for further consumers.

## [2.](#) Terminology

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.

**Media Type:** A registered label for representations (byte strings) prepared for interchange, identified by a Media-Type-Name [[RFC1590](#)], [[RFC6838](#)].

**Media-Type-Name:** A combination of a type-name and a subtype-name registered in [[IANA.media-types](#)] as per [[RFC6838](#)], conventionally identified by the two names separated by a slash.

**Content-Type:** A Media-Type-Name, optionally associated with parameters ([Section 5 of](#) [[RFC2045](#)], separated from the Media-Type-

Name and from each other by a semicolon). In HTTP and many other protocols, used in a Content-Type header field.

content coding: A name registered in the HTTP Content Coding registry [[IANA.http-parameters](#)] as specified by Sections [16.6.1](#) and 18.6 of [[I-D.ietf-httpbis-semantics](#)], indicating an encoding transformation with semantics further specified in Section 8.4.1 of [[I-D.ietf-httpbis-semantics](#)]. Confusingly, in HTTP, content coding values are found in a header field called "Content-Encoding", however "content coding" is the correct term for the process and the registered values.

content format: the combination of a Content-Type and zero or more content codings, identified by (1) a numeric identifier defined in the COAP Content-Formats registry [[IANA.core-parameters](#)] as per [Section 12.3 of \[RFC7252\]](#) (referred to as Content-Format number), or (2) a Content-Format-String.

Content-Format-String: the string representation of the combination

of a Content-Type and zero or more content codings.

Content-Format-Spec: the string representation of a content format; either a Content-Format-String or the (decimal) string representation of a Content-Format number.

Readers should also be familiar with the terms and concepts discussed in [[RFC8428](#)].

### [3.](#) SenML Content-Format ("ct") Field

When a SenML Record contains a Data Value field ("vd"), the Record MAY also include a Content-Format indication field, using label "ct". The value of this field is a Content-Format-Spec, i.e., one of:

- \* a CoAP Content-Format number in decimal form with no leading zeros (except for the value "0" itself). This value represents an unsigned integer in the range of 0-65535, similar to the "ct" attribute defined in [Section 7.2.1 of \[RFC7252\]](#) for CoRE Link Format [[RFC6690](#)]).
- \* or a Content-Format-String containing a Content-Type and zero or

more content codings (see below).

The syntax of this field is formally defined in [Section 6](#).

The CoAP Content-Format number provides a simple and efficient way to indicate the type of the data. Since some Internet media types and their content coding and parameter alternatives do not have assigned CoAP Content-Format numbers, using Content-Type and zero or more content codings is also allowed. Both methods use a string value in the "ct" field to keep its data type consistent across uses. When the "ct" field contains only digits, it is interpreted as a CoAP Content-Format number.

To indicate that one or more content codings are used with a Content-Type, each of the content coding values is appended to the Content-Type value (media type and parameters, if any), separated by a "@" sign, in the order of the content codings were applied (the same order as in Section 8.4 of [\[I-D.ietf-httpbis-semantics\]](#)). For example (using a content coding value of "deflate" as defined in Section 8.4.1.2 of [\[I-D.ietf-httpbis-semantics\]](#)):

```
text/plain; charset=utf-8@deflate
```

If no "@" sign is present after the media type and parameters, then no content coding has been specified, and the "identity" content coding is used -- no encoding transformation is employed.

#### [4.](#) SenML Base Content-Format ("bct") Field

The Base Content-Format Field, label "bct", provides a default value for the Content-Format Field (label "ct") within its range. The range of the base field includes the Record containing it, up to (but not including) the next Record containing a "bct" field, if any, or up to the end of the pack otherwise. The process of resolving ([Section 4.6 of \[RFC8428\]](#)) this base field is performed by adding its value with the label "ct" to all Records in this range that carry a "vd" field but do not already contain a Content-Format ("ct") field.

Figure 4 shows a variation of Figure 2 with multiple records, with the "nfc-reader" records resolving to the base field value "60" and the "iris-photo" record overriding this with the "image/png" media type (actual data left out for brevity).

```
[
  {"n":"nfc-reader", "vd":"gmNmb28YKg",
   "bct":"60", "bt":1627430700},
  {"n":"nfc-reader", "vd":"gmNiYXIYKw", "t":10},
  {"n":"iris-photo", "vd":".....", "ct":"image/png", "t":10},
  {"n":"nfc-reader", "vd":"gmNiYXoYLA", "t":20}
]
```

Figure 4: SenML pack with bct field

## 5. Examples

The following examples are valid values for the "ct" and "bct" fields (explanation/comments in parentheses):

- \* "60" (CoAP Content-Format number for "application/cbor")
- \* "0" (CoAP Content-Format number for "text/plain" with parameter "charset=utf-8")
- \* "application/json" (JSON Content-Type -- equivalent to "50" CoAP Content-Format number)
- \* "application/json@deflate" (JSON Content-Type with "deflate" as content coding -- equivalent to "11050" CoAP Content-Format number)
- \* "application/json@deflate@aes128gcm" (JSON Content-Type with "deflate" followed by "aes128gcm" as content codings)
- \* "text/csv" (Comma-Separated Values (CSV) [[RFC4180](#)] Content-Type)

- \* "text/csv;header=present@gzip" (CSV with header row, using "gzip" as content coding)

## 6. ABNF

This specification provides a formal definition of the syntax of Content-Format-Spec strings using ABNF notation [[RFC5234](#)], which contains three new rules and a number of rules collected and adapted

from various RFCs [[I-D.ietf-httpbis-semantic](#)] [[RFC6838](#)] [[RFC5234](#)] [[RFC8866](#)].

; New in this document

Content-Format-Spec = Content-Format-Number / Content-Format-String

Content-Format-Number = "0" / (POS-DIGIT \*DIGIT)

Content-Format-String = Content-Type \*("@" Content-Coding)

; Cleaned up from [RFC-httpbis-semantic],  
; leaving only SP as blank space,  
; removing legacy 8-bit characters, and  
; leaving the parameter as mandatory with each semicolon:

Content-Type = Media-Type-Name \*( \*SP ";" \*SP parameter )  
parameter = token "=" ( token / quoted-string )

token = 1\*tchar

tchar = "!" / "#" / "\$" / "%" / "&" / "'" / "\*" /  
/ "+" / "-" / "." / "^" / "\_" / "`" / "|" / "~"  
/ DIGIT / ALPHA

quoted-string = %x22 \*( qdtext / quoted-pair ) %x22

qdtext = SP / %x21 / %x23-5B / %x5D-7E

quoted-pair = "\" ( SP / VCHAR )

; Adapted from [section 8.4.1](#) of [RFC-httpbis-semantic]

Content-Coding = token

; Adapted from various specs

Media-Type-Name = type-name "/" subtype-name

; [RFC 6838](#)

type-name = restricted-name

subtype-name = restricted-name

restricted-name = restricted-name-first \*126restricted-name-chars

restricted-name-first = ALPHA / DIGIT



```

restricted-name-chars = ALPHA / DIGIT / "!" / "#" /
                        "$" / "&" / "-" / "^" / "_"
restricted-name-chars =/ "." ; Characters before first dot always
                             ; specify a facet name
restricted-name-chars =/ "+" ; Characters after last plus always
                             ; specify a structured syntax suffix

```

; Boilerplate from [RFC 5234](#) and [RFC 8866](#)

```

DIGIT      = %x30-39          ; 0 - 9
POS-DIGIT  = %x31-39          ; 1 - 9
ALPHA      = %x41-5A / %x61-7A ; A - Z / a - z
SP         = %x20
VCHAR      = %x21-7E          ; printable ASCII (no SP)

```

Figure 5: ABNF syntax of Content-Format-Spec

```

// RFC editor: Please replace [RFC-httpbis-semantics] by what gets
// published from [I-D.ietf-httpbis-semantics].

```

## 7. Security Considerations

The indication of a media type in the data does not exempt a consuming application from properly checking its inputs. Also, the ability for an attacker to supply crafted SenML data that specify media types chosen by the attacker may expose vulnerabilities of handlers for these media types to the attacker. This includes "decompression bombs", compressed data that is crafted to decompress to extremely large data items.

## 8. IANA Considerations

(Note to RFC Editor: Please replace all occurrences of "RFC-AAAA" with the RFC number of this specification and remove this note.)

IANA is requested to assign new labels in the "SenML Labels" subregistry of the SenML registry [[IANA.senml](#)] (as defined in [Section 12.2 of \[RFC8428\]](#)) for the Content-Format indication as per Table 1:

| Name                | Label | JSON Type | XML Type | Reference |
|---------------------|-------|-----------|----------|-----------|
| Base Content-Format | bct   | String    | string   | RFC-AAAA  |
| Content-Format      | ct    | String    | string   | RFC-AAAA  |

Table 1: IANA Registration for new SenML Labels

Note that as per [Section 12.2 of \[RFC8428\]](#), no CBOR labels or EXI schemaId values (EXI ID column) are supplied.

## 9. References

### 9.1. Normative References

[I-D.ietf-httpbis-semantics]

Fielding, R. T., Nottingham, M., and J. Reschke, "HTTP Semantics", Work in Progress, Internet-Draft, [draft-ietf-httpbis-semantics-19](#), 12 September 2021, <<https://www.ietf.org/archive/id/draft-ietf-httpbis-semantics-19.txt>>.

[IANA.core-parameters]

IANA, "Constrained RESTful Environments (CoRE) Parameters", <<https://www.iana.org/assignments/core-parameters>>.

[IANA.http-parameters]

IANA, "Hypertext Transfer Protocol (HTTP) Parameters", <<https://www.iana.org/assignments/http-parameters>>.

[IANA.media-types]

IANA, "Media Types", <<https://www.iana.org/assignments/media-types>>.

[IANA.senml]

IANA, "Sensor Measurement Lists (SenML)", <<https://www.iana.org/assignments/senml>>.

[RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", [RFC 2045](#), DOI 10.17487/RFC2045, November 1996, <<https://www.rfc-editor.org/info/rfc2045>>.

Internet-Draft

SenML Data Content-Format Indication

October 2021

- [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>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", [RFC 7252](#), DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.
- [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>>.
- [RFC8428] Jennings, C., Shelby, Z., Arkko, J., Keranen, A., and C. Bormann, "Sensor Measurement Lists (SenML)", [RFC 8428](#), DOI 10.17487/RFC8428, August 2018, <<https://www.rfc-editor.org/info/rfc8428>>.

## [9.2](#). Informative References

- [RFC1590] Postel, J., "Media Type Registration Procedure", [RFC 1590](#), DOI 10.17487/RFC1590, March 1994, <<https://www.rfc-editor.org/info/rfc1590>>.
- [RFC4180] Shafranovich, Y., "Common Format and MIME Type for Comma-Separated Values (CSV) Files", [RFC 4180](#), DOI 10.17487/RFC4180, October 2005, <<https://www.rfc-editor.org/info/rfc4180>>.
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", [RFC 4648](#), DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.

- [RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", [RFC 6690](#), DOI 10.17487/RFC6690, August 2012, <<https://www.rfc-editor.org/info/rfc6690>>.
- [RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", [BCP 13](#), [RFC 6838](#), DOI 10.17487/RFC6838, January 2013, <<https://www.rfc-editor.org/info/rfc6838>>.

Keränen & Bormann

Expires 24 April 2022

[Page 10]

---

Internet-Draft

SenML Data Content-Format Indication

October 2021

- [RFC8866] Begen, A., Kyzivat, P., Perkins, C., and M. Handley, "SDP: Session Description Protocol", [RFC 8866](#), DOI 10.17487/RFC8866, January 2021, <<https://www.rfc-editor.org/info/rfc8866>>.
- [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>>.

## Acknowledgements

The authors would like to thank Sérgio Abreu for the discussions leading to the design of this extension and Isaac Rivera for reviews and feedback. Klaus Hartke suggested not burdening this draft with a separate mandatory-to-implement version of the fields. Alexey Melnikov, Jim Schaad, and Thomas Fossati provided helpful comments at Working-Group last call. Marco Tiloca asked for clarifying and using the term Content-Format-Spec.

## Authors' Addresses

Ari Keränen  
Ericsson  
FI-02420 Jorvas  
Finland

Email: [ari.keranen@ericsson.com](mailto:ari.keranen@ericsson.com)

Carsten Bormann  
Universität Bremen TZI

Postfach 330440  
D-28359 Bremen  
Germany

Phone: +49-421-218-63921  
Email: [cabo@tzi.org](mailto:cabo@tzi.org)