

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: February 18, 2020

A. Keranen  
Ericsson  
M. Mohajer  
u-blox UK  
August 17, 2019

**FETCH & PATCH with Sensor Measurement Lists (SenML)**  
**draft-ietf-core-senml-etch-05**

Abstract

The Sensor Measurement Lists (SenML) media type and data model can be used to send collections of resources, such as batches of sensor data or configuration parameters. The CoAP iPATCH, PATCH, and FETCH methods enable accessing and updating parts of a resource or multiple resources with one request. This document defines new media types for the CoAP iPATCH, PATCH, and FETCH methods for resources represented with the SenML data model.

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 February 18, 2020.

Copyright Notice

Copyright (c) 2019 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="#">2.</a>	Terminology . . . . .	<a href="#">3</a>
<a href="#">3.</a>	Using FETCH and (i)PATCH with SenML . . . . .	<a href="#">3</a>
<a href="#">3.1.</a>	SenML FETCH . . . . .	<a href="#">4</a>
<a href="#">3.2.</a>	SenML (i)PATCH . . . . .	<a href="#">4</a>
<a href="#">4.</a>	Fragment Identification . . . . .	<a href="#">5</a>
<a href="#">5.</a>	Security Considerations . . . . .	<a href="#">6</a>
<a href="#">6.</a>	IANA Considerations . . . . .	<a href="#">6</a>
<a href="#">6.1.</a>	CoAP Content-Format Registration . . . . .	<a href="#">6</a>
<a href="#">6.2.</a>	senml-etch+json Media Type . . . . .	<a href="#">6</a>
<a href="#">6.3.</a>	senml-etch+cbor Media Type . . . . .	<a href="#">7</a>
<a href="#">7.</a>	Acknowledgements . . . . .	<a href="#">9</a>
<a href="#">8.</a>	References . . . . .	<a href="#">9</a>
<a href="#">8.1.</a>	Normative References . . . . .	<a href="#">9</a>
<a href="#">8.2.</a>	Informative References . . . . .	<a href="#">10</a>
	Authors' Addresses . . . . .	<a href="#">10</a>

## [1.](#) Introduction

The Sensor Measurement Lists (SenML) media type [[RFC8428](#)] and data model can be used to transmit collections of resources, such as batches of sensor data or configuration parameters.

An example of a SenML collection is shown below:

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850", "vb":true},
  {"n":"5851", "v":42},
  {"n":"5750", "vs":"Ceiling light"}
]
```

Here three resources "3311/0/5850", "3311/0/5851", and "3311/0/5750", of an IPSO dimmable light smart object [[IPSO](#)] are represented using a single SenML Pack with three SenML Records. All resources share the same base name "2001:db8::2/3311/0/", hence full names for resources are "2001:db8::2/3311/0/5850", etc.

The CoAP [[RFC7252](#)] iPATCH, PATCH, and FETCH methods [[RFC8132](#)] enable accessing and updating parts of a resource or multiple resources with one request.



This document defines two new media types, one using the JavaScript Object Notation (JSON) [[RFC8259](#)] and one using the Concise Binary Object Representation (CBOR) [[RFC7049](#)], that can be used with the CoAP iPATCH, PATCH, and FETCH methods for resources represented with the SenML data model. The semantics of the new media types are the same for the CoAP PATCH and iPATCH methods. The rest of the document uses term "(i)PATCH" when referring to both methods.

## 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.

Readers should also be familiar with the terms and concepts discussed in [[RFC8132](#)] and [[RFC8428](#)]. Also the following terms are used in this document:

Fetch Record: One set of parameters that is used to match SenML Record(s).

Fetch Pack: One or more Fetch Records in an array structure.

Patch Record: One set of parameters similar to Fetch Record but also containing instructions on how to change existing SenML Pack(s).

Patch Pack: One or more Patch Records in an array structure.

Target Record: A Record in a SenML Pack that is matching the selection criteria of a Fetch or Patch Record and hence is a target for a Fetch or Patch operation.

(i)PATCH: A term that refers to both CoAP "PATCH" and "iPATCH" methods when there is no difference in this specification in which one is used.

## 3. Using FETCH and (i)PATCH with SenML

The FETCH/(i)PATCH media types for SenML are modeled as extensions to the SenML media type to enable re-use of existing SenML parsers and generators, in particular on constrained devices. Unless mentioned otherwise, FETCH and PATCH Packs are constructed with the same rules and constraints as SenML Packs.

The key difference to the SenML media type is allowing the use of a "null" value for removing records with the (i)PATCH method. Also the



Fetch and Patch Records do not have default time or base version when the fields are omitted.

### **3.1. SenML FETCH**

The FETCH method can be used to select and return a subset of records, in sequence, of one or more SenML Packs. The SenML Records are selected by giving a set of names that, when resolved, match resolved names in a SenML Pack. The names for a Fetch Pack are given using the SenML "name" and/or "base name" Fields. The names are resolved by concatenating the base name with the name field as defined in [\[RFC8428\]](#).

For example, to select the IPSO resources "5850" and "5851" from the example in [Section 1](#), the following Fetch Pack can be used:

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850"},
  {"n":"5851"}
]
```

The result to a FETCH request with the example above would be:

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850", "vb":true},
  {"n":"5851", "v":42},
]
```

When SenML Records contain also time values, a name may no longer uniquely identify a single Record. When no time is given in a Fetch Record, all SenML Records with the given name are matched (i.e., unlike with SenML Records, lack of time field in a Fetch Record does not imply time value zero). When time is given in the Fetch Record, only a SenML Record (if any) with equal resolved time value and name is matched.

The resolved form of records ([Section 4.6 of \[RFC8428\]](#)) is used when comparing the names and times of the Target and Fetch Records to accommodate for differences in use of the base values.

### **3.2. SenML (i)PATCH**

The (i)PATCH method can be used to change the values of SenML Records, to add new Records, and to remove existing Records. The names and times of the Patch Records are given and matched in same way as for the Fetch Records, except each Patch Record can match at most one Target Record. Patch Packs can also include new values and



other SenML Fields for the Records. Application of Patch Packs is idempotent.

When the name in a Patch Record matches with the name in an existing Record, the resolved time values are compared. If the time values either do not exist in both Records or are equal, the Target Record is replaced with the contents of the Patch Record.

If a Patch Record contains a name, or combination of a time value and a name, that do not exist in any existing Record in the Pack, the given Record, with all the fields it contains, is added to the Pack.

If a Patch Record has a value ("v") field with value null, the matched Record (if any) is removed from the Pack.

For example, the following document could be given as an (i)PATCH payload to change/set values of two SenML Records for the example in [Section 1](#):

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850", "vb":false},
  {"n":"5851", "v":10}
]
```

If the request is successful, the resulting representation of the example SenML Pack would be as follows:

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850", "vb":false},
  {"n":"5851", "v":10},
  {"n":"5750", "vs":"Ceiling light"}
]
```

As another example, the following document could be given as an (i)PATCH payload to remove the two SenML Records:

```
[
  {"bn":"2001:db8::2/3311/0/", "n":"5850", "v":null},
  {"n":"5851", "v":null}
]
```

#### **4. Fragment Identification**

Fragment identification is supported by analogously applying fragment identifiers as specified in [Section 9 of \[RFC8428\]](#) to the Fetch/Patch Records.





## 5. Security Considerations

The security and privacy considerations of SenML apply also with the FETCH and (i)PATCH methods.

In FETCH and (i)PATCH requests, the client can pass arbitrary names to the target resource for manipulation. The resource implementer must take care to only allow access to names that are actually part of (or accessible through) the target resource.

If the client is not allowed to do a GET or PUT on the full target resource (and thus all the names accessible through it), access control rules must be evaluated for each record in the pack.

## 6. IANA Considerations

This document registers two new media types and CoAP Content-Format IDs for both media types.

Note to RFC Editor: Please replace all occurrences of "RFC-AAAA" with the RFC number of this document.

### 6.1. CoAP Content-Format Registration

IANA is requested to assign CoAP Content-Format IDs for the SenML PATCH and FETCH media types in the "CoAP Content-Formats" sub-registry, within the "CoRE Parameters" registry [[RFC7252](#)]. The assigned IDs are shown in Table 1.

Media type	Encoding	ID
application/senml-etch+json	-	TBD-320
application/senml-etch+cbor	-	TBD-322

Table 1: CoAP Content-Format IDs

### 6.2. senml-etch+json Media Type

Type name: application

Subtype name: senml-etch+json

Required parameters: none

Optional parameters: none



Encoding considerations: binary

Security considerations: See [Section 5](#) of RFC-AAAA.

Interoperability considerations: Applications MUST ignore any key value pairs that they do not understand unless the key ends with the '\_' character in which case an error MUST be generated. This allows backwards compatible extensions to this specification.

Published specification: RFC-AAAA

Applications that use this media type: Applications that use the SenML media type for resource representation.

Fragment identifier considerations: Fragment identification for application/senml-etch+json is supported by using fragment identifiers as specified by RFC AAAA.

Additional information:

Magic number(s): none

File extension(s): senml-etchj

Windows Clipboard Name: "SenML FETCH/PATCH format"

Macintosh file type code(s): none

Macintosh Universal Type Identifier code: org.ietf.senml-etch-json conforms to public.text

Person & email address to contact for further information: Ari Keranen ari.keranen@ericsson.com

Intended usage: COMMON

Restrictions on usage: None

Author: Ari Keranen ari.keranen@ericsson.com

Change controller: IESG

### **[6.3.](#) senml-etch+cbor Media Type**

Type name: application

Subtype name: senml-etch+cbor



Required parameters: none

Optional parameters: none

Encoding considerations: binary

Security considerations: See [Section 5](#) of RFC-AAAA.

Interoperability considerations: Applications MUST ignore any key value pairs that they do not understand unless the key ends with the '\_' character in which case an error MUST be generated. This allows backwards compatible extensions to this specification.

Published specification: RFC-AAAA

Applications that use this media type: Applications that use the SenML media type for resource representation.

Fragment identifier considerations: Fragment identification for application/senml-etch+cbor is supported by using fragment identifiers as specified by RFC AAAA.

Additional information:

Magic number(s): none

File extension(s): senml-etchc

Macintosh file type code(s): none

Macintosh Universal Type Identifier code: org.ietf.senml-etch-cbor conforms to public.data

Person & email address to contact for further information: Ari Keranen ari.keranen@ericsson.com

Intended usage: COMMON

Restrictions on usage: None

Author: Ari Keranen ari.keranen@ericsson.com

Change controller: IESG



## **7. Acknowledgements**

The use of FETCH and (i)PATCH methods with SenML was first introduced by the OMA SpecWorks LwM2M v1.1 specification. This document generalizes the use to any SenML representation. The authors would like to thank Carsten Bormann, Christian Amsuess, Jaime Jimenez, Klaus Hartke, Michael Richardson, and other participants from the IETF CoRE and OMA SpecWorks DMSE working groups who have contributed ideas and reviews.

## **8. References**

### **8.1. Normative References**

- [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>>.
- [RFC7049] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", [RFC 7049](#), DOI 10.17487/RFC7049, October 2013, <<https://www.rfc-editor.org/info/rfc7049>>.
- [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>>.
- [RFC8132] van der Stok, P., Bormann, C., and A. Sehgal, "PATCH and FETCH Methods for the Constrained Application Protocol (CoAP)", [RFC 8132](#), DOI 10.17487/RFC8132, April 2017, <<https://www.rfc-editor.org/info/rfc8132>>.
- [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>>.
- [RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, [RFC 8259](#), DOI 10.17487/RFC8259, December 2017, <<https://www.rfc-editor.org/info/rfc8259>>.
- [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>>.





## **8.2. Informative References**

[IPSO] IPSO, "IPSO Light Control Smart Object", 2018,  
<[http://www.openmobilealliance.org/tech/profiles/  
lwm2m/3311.xml](http://www.openmobilealliance.org/tech/profiles/lwm2m/3311.xml)>.

### Authors' Addresses

Ari Keranen  
Ericsson

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

Mojan Mohajer  
u-blox UK

Email: [Mojan.Mohajer@u-blox.com](mailto:Mojan.Mohajer@u-blox.com)

