

CoRE  
Internet-Draft  
Updates: [7641](#) (if approved)  
Intended status: Standards Track  
Expires: September 26, 2019

C. Amsuess  
March 25, 2019

Accept-Any option for CoAP  
draft-amsuess-core-accept-any-00

## Abstract

This memo defines the Accept-Any option, which provides a more flexible content negotiation than the one originally specified for the Constrained Application Protocol (CoAP) in [[RFC7252](#)]. As this is particularly useful with but ruled out in CoAP observation ([[RFC7641](#)]), that is updated to allow it.

## 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 September 26, 2019.

## 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

Internet-Draft

Accept-Any option for CoAP

March 2019

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## 1. Introduction

[ This document is being developed in git at <https://github.com/chrysn/accept-any> [1]. ]

When CoAP content format defined in [RFC7252], the choice was made to have the initial content negotiation allow the client to only pick zero or one content format. This is a good choice for many situations and ensures that proxies can cache as much as possible without added complexity. A client that does not know a usable content format would either leave the request field absent, indicating it would accept any response, or try several acceptable values in a series of requests.

In line with that choice, observation ([RFC7641]) required notification responses to carry the same content format in the notification as in the original response.

For applications that expect a representation to change its representability to change during an observation's lifetime, [I-D.ietf-core-multipart-ct] introduces a way of wrapping responses in an application/multipart-core response. That approach is convenient for bags of representations, but lacks actual content negotiation and produces representations that are not directly usable but need to be processed from inside the multipart representation.

This document introduces an additional way for the client to indicate its set of acceptable content formats, and removes the same-content-format limitation during observation.

## 2. The Accept-Any option

A new option Accept-Any is defined. It is critical [ I don't fully see why but follow Accept here ], safe-to-forward and part of the cache key. Its format is uint up to 2 long (indicating content types), it is Class E in OSCORE, usable in requests only, and repeatable

Repeatability is the only aspect in which it differs from Accept in terms of option properties.

Its values indicate a list of acceptable representations in order of decreasing preference. A server MUST answer with the first format it can represent the requested state in, or 4.06 (Not Acceptable) if it

could answer successfully but the response would not match any of the option values.

The Accept-Any option MUST NOT be used exactly once; that request's meaning would be identical to that of a single Accept option. Instead, a single Accept option is used.

### [2.1.](#) Proxy behavior

A proxy MAY ignore this option per its properties (and serve a cached response if the cache key matches), but can implement additional behavior to enhance its cache.

A proxy is allowed to serve a cached representation to a request with a different sequence of Accept-Any options, provided the second request has an Accept value of the cached representation, or all the content formats that precede the available content format in the second request's Accept-Any options also preceded the available representation in any earlier (fresh) request's list.

When a request that carries Accept-Any is answered 4.06 (or with any but the first format requested by Accept-Any in its Content-Format), a proxy SHOULD [ we can't have a MUST here w/o making it non-safe-to-forward, but I think it's sufficient ] invalidate all known representations in any of the requested formats (or the formats preceding the returned one, respectively).

## [3.](#) Update to [RFC7641](#)

### [3.1.](#) Changed behavior

The requirement that subsequent notifications carry the same Content-Format option as the original response ([\[RFC7641\] Section 3.2](#)) is lifted.

### [3.2.](#) Rationale

Observing resources whose available representations changed is a key feature of Accept-Any, and necessary to implement pub-sub topics that have no initial value (but a "null" representation with a dedicated content format) without losing content negotiation and direct usability of the response.

The requirement was introduced initially before such content negotiation was thought of, and is not a necessary part to the remainder of the observation document.

Amsuess

Expires September 26, 2019

[Page 3]

---

Internet-Draft

Accept-Any option for CoAP

March 2019

As long as the limitation is in place, the origin server has no clear action guidance when its resource changes the available content formats (see below).

### [3.3.](#) Impact

Changes to the returned media type can either happen when

- o Accept-Any was sent in the request - in which case both server and client know the updated rules, or
- o no Accept header was sent - in which case the server whose representation changes to require a new content format has no clear way of indicating that under [[RFC7641](#)] (ending with 4.06 Not Acceptable would be close but isn't the expected response to a repeated request); if the server changes the content format in a notification to an unaware client, the client would catch it as a bad response (probably similar to a response with a Content-Format not matching the sent Accept). The client might regard the observation as being aborted while the server does not, and will terminate the observation with a RST on the next notification (or close the connection in TCP).

Applications are still free to require constant content formats; clients would treat what could previously be treated as a protocol error would now treat it as an application error.

Impact on proxies: A proxy that enforces the previous rule on Content-Format staying constant would close observations (probably

with something like 5.02 Bad Gateway), and the client would need to re-establish. No proxy implementations are known that implement that behavior.

## [4.](#) References

### [4.1.](#) Normative References

- [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>>.
- [RFC7641] Hartke, K., "Observing Resources in the Constrained Application Protocol (CoAP)", [RFC 7641](#), DOI 10.17487/RFC7641, September 2015, <<https://www.rfc-editor.org/info/rfc7641>>.

Amsuess

Expires September 26, 2019

[Page 4]

---

Internet-Draft

Accept-Any option for CoAP

March 2019

### [4.2.](#) Informative References

- [I-D.ietf-core-multipart-ct]  
Fossati, T., Hartke, K., and C. Bormann, "Multipart Content-Format for CoAP", [draft-ietf-core-multipart-ct-03](#) (work in progress), March 2019.

### [4.3.](#) URIs

- [1] <https://github.com/chrysn/accept-any>

## [Appendix A.](#) Open questions

- o We could just as well make Accept repeatable with the same semantics as Accept-Any.
- o Is there any value in having an Accept-All option in parallel to this option that asks for a multipart response that contains all the representations?

Author's Address

Christian Amsuess  
Hollandstr. 12/4  
1020  
Austria

Phone: +43-664-9790639  
Email: christian@amsuess.com