

CoRE  
Internet-Draft  
Intended status: Standards Track  
Expires: 13 March 2026

C. Ams端 ss  
M. Richardson  
Sandelman Software Works  
9 September 2025

Short Paths in CoAP  
draft-amsuess-core-shopinc-02

## Abstract

Applications built on CoAP face a conflict between the technical need for short message sizes and the interoperability requirement of following BCP190 and thus registering (relatively verbose) well-known URI paths. This document introduces an option that allows expressing well-known paths in as little as two bytes.

## About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at  
<https://datatracker.ietf.org/doc/draft-amsuess-core-shopinc/>.

Discussion of this document takes place on the Constrained RESTful Environments Working Group mailing list (<mailto:core@ietf.org>), which is archived at <https://mailarchive.ietf.org/arch/browse/core/>.  
Subscribe at <https://www.ietf.org/mailman/listinfo/core/>.

Source for this draft and an issue tracker can be found at  
<https://github.com/chrysn/shopinc>.

## 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 13 March 2026.

## Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction . . . . .	2
1.1. Conventions and Definitions . . . . .	3
2. The Uri-Path-Abbr option . . . . .	3
2.1. Proxy behavior . . . . .	4
2.2. Interaction with other options . . . . .	4
2.3. Repeated use . . . . .	5
2.3.1. Examples of rules . . . . .	5
2.3.2. Considerations for choosing rules and prefixes . . . . .	5
2.4. Choice of the option number . . . . .	6
3. Initial Uri-Path-Abbr values . . . . .	6
4. Security Considerations . . . . .	6
5. IANA Considerations . . . . .	7
5.1. CoAP option: Uri-Path-Abbr . . . . .	7
5.2. Uri-Path-Abbr registry . . . . .	7
6. References . . . . .	8
6.1. Normative References . . . . .	9
6.2. Informative References . . . . .	9
Appendix A. Further development . . . . .	10
Appendix B. Open questions . . . . .	11
Appendix C. Change log . . . . .	12
Acknowledgments . . . . .	12
Authors' Addresses . . . . .	12

## 1. Introduction

[ This is an early draft, please read the abstract. ]

## 1.1. Conventions and Definitions

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.

This document assumes basic familiarity with CoAP ([RFC7252]), in particular its Uri-\* options.

## 2. The Uri-Path-Abbr option

The Uri-Path-Abbr option (short for "URI path, abbreviated") expresses a request's URI path in a more compact form.

The Uri-Path-Abbr value represents a particular path, and is thus equivalent to any number of Uri-Path options. Those paths are typically in a "/.well-known" location as described in [RFC8615]. The option values are coordinated by IANA in the Uri-Path-Abbr registry established in this document in Section 5.1.

A client may use the option instead of the Uri-Path option if there is a suitable value that can express the requested path. Unless the client can be assured that the server supports it (e.g. because the specification describing the interaction mandates support for the option in the server) it SHOULD fall back to sending the path explicitly if it receives an error indicating that the option was not understood (otherwise, it would have limited interoperability).

A server receiving the option with an unknown value MUST treat it as an unprocessable critical option, returning 4.02 Bad Option and MUST NOT return a 4.04 Not Found response, because the equivalent path may be present on the server.

A server that supports a Uri-Path-Abbr value MUST also support the equivalent request composed of Uri-Path components.

No.	C	U	N	R	Name	Format	Len.	Default
CPA13	x			x	Uri-Path-Abbr	uint / opaque	any	(none)

Table 1: Uri-Path-Abbr Option Summary (C = Critical, U = Unsafe, N = NoCacheKey, R = Repeatable)

```
// RFC-Editor: This document uses the CPA (code point allocation)
// convention described in [I-D.bormann-cbor-draft-numbers]. For
// each usage of the term "CPA", please remove the prefix "CPA" from
// the indicated value and replace the residue with the value
// assigned by IANA; perform an analogous substitution for all other
// occurrences of the prefix "CPA" in the document. Finally, please
// remove this note.
```

The Uri-Path-Abbr option has an integer value in its first occurrence; the type of later occurrences differs depending on the first. It is a critical and safe-to-forward option that is part of the cache key, used in CoAP requests. Table 1 summarizes these, extending Table 4 of [RFC7252]). Its OSCORE treatment is as Class E ([RFC8613]).

Apart from the format, these properties only deviate from the Uri-Path (for which it stands in) in that this option is safe to forward. This has unfortunate consequences for the interactions with the Proxy-URI option, but is generally desirable: It allows the option to be used with proxies that do not implement the option.

## 2.1. Proxy behavior

A proxy MAY expand or introduce a Uri-Path-Abbr when forwarding a request, in particular for serving cached responses, as long as this introduces no new errors to the client.

A proxy that knows Uri-Path-Abbr but not the concrete value SHOULD forward it unmodified, which is the behavior it would apply if it did not know the option. A reason to reject the request instead is when the proxy is tasked with enforcing access control (see Section 4).

## 2.2. Interaction with other options

The option is mutually exclusive with the Uri-Path option. Receiving both options in a single request MUST be treated like the presence of a critical request option that could not be processed (that option being either the Uri-Path-Abbr option or the conflicting option).

The Uri-Path-Abbr option MUST NOT be used in combination with the Proxy-Uri option (or the similar Proxy-CRI option (of [I-D.ietf-core-href])) by clients. Proxies that understand Uri-Path-Abbr and convert Uri-\* options into Proxy-Uri MUST expand any Uri-Path-Abbr if they know the value.

By the (de)composition rules around Proxy-Uri, and because Uri-Path-Abbr is safe-to-forward, a proxy (being generally unaware of this specification) is allowed to combine the option with Proxy-Uri (or

Proxy-CRI) when it combines the Uri-\* options. In such a combined message, the Uri-Path segments to which the Uri-Path-Abbr corresponds are appended to the path as if all components were present as individual options in the request without conflicting. Servers that support both Uri-Path-Abbr and Proxy-URI/-CRI SHOULD process requests accordingly. (This is not a strict requirement, as there are no known implementations of proxies that actually compose a Proxy-URI/-CRI from individual options, nor is there a reason known why they should).

### 2.3. Repeated use

The document defining the registered value of the first Uri-Path-Abbr option may allow additional Uri-Path-Abbr options, Their value is not expanded through the Uri-Path-Abbr IANA registry, but according to rules set up in that particular registration.

To be implementable on a wide variety of platforms, those rules should allow expansion into Uri-Path options in an iterative way (i.e., any added Uri-Path-Abbr option corresponds only to appended Uri-Path options, or cause a 4.02 Bad Option error, except if there are no resources at all with that prefix, in which case 4.04 Not Found may be used instead).

#### 2.3.1. Examples of rules

Some rules anticipated to be used are:

- \* Options after the first are treated exactly like Uri-Path options.
- \* There can be only one added Uri-Path-Abbr option, and its opaque value is looked up in a table shaped like the Uri-Path-Abbr IANA registry.

#### 2.3.2. Considerations for choosing rules and prefixes

It is recommended that the expansion of the first Uri-Path-Abbr option does not end with a trailing slash. While that is a valid CoAP URI, any additional path segments generated by expanding additional Uri-Path-Abbr options would generate paths with interior double slashes, which is highly unusual and generally not intended.

When no repeated use is anticipated, it is recommended to not forbid them outright: That would make it harder to use that extension point later, as allowing it would be a breaking change to the specification. Instead, either the common behavior of treating them as extra Uri-Path strings can be specified (which does not make a server without resources under the prefix any more complex, as it may

answer react with 4.02 or 4.04 as per Section 2.3), or the semantics of repeated options can explicitly be left unspecified (which retains more flexibility in assigning meaning later).

#### 2.4. Choice of the option number

TBD: Rephrase this to either be useful for readers of the final document who can thus learn how the option number namespace is managed, or remove before publication.

It's already 1+1 -- we generally do try to keep even the 1+1 high so that later option typically paired with a low option (like EDHOC paired with OSCORE) can use the small delta. In this case, there's a good reason (being ordered before Uri-Query) though, and I don't expect that any other option would need this particular property (especially given that this option on its own has an extensible value range).

#### 3. Initial Uri-Path-Abbr values

This document registers values for the following well-known URIs:

- \* /.well-known/core
- \* /.well-known/rd (see [RFC9175])
- \* /.well-known/brski (see [I-D.ietf-anima-constrained-voucher])
- \* /.well-known/est (see [RFC9148])

For all those, later occurrences of Uri-Path-Abbr are interpreted as additional Uri-Path values. While there are currently no resources under the CoRE and RD resource, this behavior is useful in BRSKI and EST.

Note that the former two paths are commonly used with Uri-Query options.

#### 4. Security Considerations

Having alternative expressions for information that is input to policy decisions can be problematic when the mechanism performing the check has a different interpretation of the presented data than the mechanism at time of use. That concern is not new to this document: Both the Proxy-Uri of [RFC7252] and the Proxy-Cri option of [I-D.ietf-core-href] have the same properties in that regard. The appropriate behavior is for policy checkers to reject any request that contains critical options that is not understood; the

application protected by the checker may provide the checker with an allow-list of options that it will treat as unchecked input.

## 5. IANA Considerations

### 5.1. CoAP option: Uri-Path-Abbr

IANA is requested to enter an one option into the CoAP Option Numbers registry in the CoRE Parameters group:

- \* Number: CPA13
- \* Name: Uri-Path-Abbr
- \* Reference: this document

### 5.2. Uri-Path-Abbr registry

IANA is requested to establish a new registry in the CoRE parameters group: Values of the first Uri-Path-Abbr option in a CoAP request correspond to a URI path according to this registry.

The policy for adding any value is IETF Review (as described in [RFC8126]). Change control for the registry follows this document's publication stream. Initial values are given in Table 2.

Entry fields are:

- \* First option value.

An non-negative integer that can be expressed in 32 bits, unique within this registry.

All positive values whose most significant bit of the most significant byte is 1 are reserved.

The Python invocation `python3 -c 'print("reserved" if (250).bit_length() % 8 == 0 else "unreserved")'` can be used to quickly test this property for any positive number (250 in this example).

- \* Simple expanded path.

This text is the URI path (starting with /) that the option, when present only once in a request, is expanded to.

This field may be empty if the document describes that the option needs to be repeated when using this first value.

\* Reference.

A document that requested the allocation, and describes whether the option may be repeated after this first value, and how later values are expanded

Reviewer instructions:

The reviewer is instructed to be frugal with the 128 values that correspond to a single-vbyte value, focusing on applications that are expected to be useful in different constrained ecosystems.

If the considerations of Section 2.3.2 are not followed, the reviewer is asked to verify with the applicant that they are deliberately deciding otherwise.

The expanded path (or paths) are expected to be well-known paths at the time of writing, but it is up to the reviewers to exceptionally also admit paths that are not well-known.

If the registration foresees updates, those should always just allow previously unacceptable values into new path segments, and not alter the semantics of previously valid expansions.

First option value	Simple expanded path	Reference
0	/.well-known/core	Section 3 of this document
1	/.well-known/rd	Section 3 of this document, and [RFC9176]
2	/.well-known/brski	Section 3 of this document, and [I-D.ietf-anima-constrained-voucher]
3	/.well-known/est	Section 3 of this document, and [RFC9148]

Table 2: Initial values for the Uri-Path-Abbr registry

## 6. References



## 6.1. 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/rfc/rfc2119>>.
- [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/rfc/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/rfc/rfc8174>>.

## 6.2. Informative References

- [I-D.bormann-cbor-draft-numbers] Bormann, C., "Managing CBOR codepoints in Internet-Drafts", Work in Progress, Internet-Draft, draft-bormann-cbor-draft-numbers-06, 7 July 2025, <<https://datatracker.ietf.org/doc/html/draft-bormann-cbor-draft-numbers-06>>.
- [I-D.ietf-anima-constrained-voucher] Richardson, M., Van der Stok, P., Kampanakis, P., and E. Dijk, "Constrained Bootstrapping Remote Secure Key Infrastructure (cBRSKI)", Work in Progress, Internet-Draft, draft-ietf-anima-constrained-voucher-28, 6 July 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-anima-constrained-voucher-28>>.
- [I-D.ietf-core-href] Bormann, C. and H. Birkholz, "Constrained Resource Identifiers", Work in Progress, Internet-Draft, draft-ietf-core-href-24, 30 August 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-core-href-24>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/rfc/rfc8126>>.

- [RFC8613] Selander, G., Mattsson, J., Palombini, F., and L. Seitz, "Object Security for Constrained RESTful Environments (OSCORE)", RFC 8613, DOI 10.17487/RFC8613, July 2019, <<https://www.rfc-editor.org/rfc/rfc8613>>.
- [RFC8615] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, DOI 10.17487/RFC8615, May 2019, <<https://www.rfc-editor.org/rfc/rfc8615>>.
- [RFC8790] Keränen, A. and M. Mohajer, "FETCH and PATCH with Sensor Measurement Lists (SenML)", RFC 8790, DOI 10.17487/RFC8790, June 2020, <<https://www.rfc-editor.org/rfc/rfc8790>>.
- [RFC9148] van der Stok, P., Kampanakis, P., Richardson, M., and S. Raza, "EST-coaps: Enrollment over Secure Transport with the Secure Constrained Application Protocol", RFC 9148, DOI 10.17487/RFC9148, April 2022, <<https://www.rfc-editor.org/rfc/rfc9148>>.
- [RFC9175] Amss, C., Preu Mattsson, J., and G. Selander, "Constrained Application Protocol (CoAP): Echo, Request-Tag, and Token Processing", RFC 9175, DOI 10.17487/RFC9175, February 2022, <<https://www.rfc-editor.org/rfc/rfc9175>>.
- [RFC9176] Amss, C., Ed., Shelby, Z., Koster, M., Bormann, C., and P. van der Stok, "Constrained RESTful Environments (CoRE) Resource Directory", RFC 9176, DOI 10.17487/RFC9176, April 2022, <<https://www.rfc-editor.org/rfc/rfc9176>>.

## Appendix A. Further development

Several possible further directions are anticipated in this document, but not specified at this point in time; they are left for further documents:

- \* The mechanism of expanding one option into another option might be expressed using the terminology of SCHC.

Such a generalization is not aimed for in this document; authors of any future document providing such a framework are encouraged to provide an equivalent but machine-readable explanation of the mechanism specified here.

- \* The registry for Uri-Path-Abbr values is set up such that first values can not have the most significant bit of the first byte set.

This allows future documents to reuse the option for any CBOR expressions, e.g. the path component of a CRI [I-D.ietf-core-href]. Note that those CBOR structures can only use the major types 4 to 7 for the top-level item, but that includes all containers (arrays, maps and tags).

Senders and recipients of this option do not need to concern themselves with that extension mechanism unless they implement it: As the first value is compared to known registry entries, any CBOR item contained in it will simply not match any known value. Should the working group decide not to use that extension point, the registry's policy can be relaxed to also allow values with that leading bit set.

- \* A future document may update this document to allow registering values that are allowed to use together with Uri-Path values (but at the time of writing, no examples are known by which such a design could be properly vetted). In particular, that update weakens the "MUST" in Section 2.2.
- \* This option is designed to stand in for the Uri-Path option alone, not for any other option; this simplifies its interaction rules.

In particular, application authors who seek to express Uri-Query options in a more concise or easier to process way are advised to avail themselves of the FETCH method introduced in [RFC8790].

## Appendix B. Open questions

This section will be gone by the time this document is published.

- \* Is the transformation of separate options to Proxy-URI even \_legal\_ for proxies?

If not, we can simplify the handling (and Uri-Path would \_really\_ not have needed to be proxy-unsafe).

- \* This document might incentivise users to send more traffic through /.well-known/ paths, rather than go through discovery. It is up to WG discussion to decide whether this is desirable; to not make this document depend on that outcome, the registration policy is currently "IETF Review", which is extremely strict and can be relaxed in a later document if the WG decides so.
- \* Do we want to add /.well-known/edhoc here, or rather fix it by updating the EDHOC option to also work without an OSCORE option?

(The author prefers the latter).

## Appendix C. Change log

Since -01: Processing 2025-08-27 interim.

- \* Document is standards track.
- \* Change name of the option from Short-Uri-Path to Uri-Path-Abbr.
- \* Close question of whether use of option 13 is justified (it is).
- \* Minor editorials.

Since -00:

- \* Switched option type from opaque to uint (retaining the lockout for values that look like CBOR arrays/maps).
- \* MCR joined as author.
- \* Added initial values for BRSKI and EST.
- \* Allow 4.04 responses.
- \* Add guidance for choosing prefixes and rules.
- \* Large editorial changes.

## Acknowledgments

This document was created out of discussion with Esko Dijk and Michael Richardson. Carsten Bormann provided useful input on shaping the registry.

## Authors' Addresses

Christian Amsss  
Austria  
Email: christian@amsuess.com

Michael Richardson  
Sandelman Software Works  
Email: mcr+ietf@sandelman.ca