

CoRE Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

B. Silverajan
Tampere University of Technology
T. Savolainen
Nokia Technologies
March 5, 2018

CoAP Communication with Alternative Transports
draft-silverajan-core-coap-alternative-transport-11

Abstract

The aim of this document is to provide a way forward to best decide upon how alternative transport information can be expressed in a CoAP URI. This draft examines the requirements for a new URI format for representing CoAP resources over alternative transports. Various potential URI formats are presented. Benefits and drawbacks of embedding alternative transport information in various ways within the URI components are also discussed. From all listed formats, the document finds scheme-based model to be the most technically feasible.

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 6, 2018.

Copyright Notice

Copyright (c) 2018 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

Internet-Draft

CoAP Alternative Transports

March 2018

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

1.	Introduction	2
2.	Conformance and Design Considerations	4
3.	Situating Transport Information in CoAP URIs	5
3.1.	Using the URI scheme component	5
3.1.1.	Analysis	6
3.2.	Using the URI authority component	6
3.2.1.	Analysis	7
3.3.	Using the URI path component	7
3.3.1.	Analysis	7
3.4.	Using the URI query component	7
3.4.1.	Analysis	8
4.	Discussion	8
5.	IANA Considerations	8
6.	Security Considerations	9
7.	Acknowledgements	9
8.	References	9
8.1.	Normative References	9
8.2.	Informative References	9
Appendix A.	Expressing transport in the URI in other ways	10
A.1.	Transport information as part of the URI authority	10
A.1.1.	Usage of DNS records	11
A.2.	Making CoAP Resources Available over Multiple Transports	12
A.3.	Transport as part of a 'service:' URL scheme	13
Authors' Addresses	14

[1.](#) Introduction

The Constrained Application Protocol (CoAP) [[RFC7252](#)] is a lightweight, binary application layer protocol designed for constrained environments. Owing to its operating environment, CoAP uses UDP and DTLS as its underlying transports between communicating endpoints. However, with an increase in deployment experiences as well as its popularity, compelling reasons exist for extending CoAP messaging to work over alternative transports. These allow CoAP to better address firewall and NAT traversal issues, to operate in Web

browser-based and HTML5 applications as well as for energy-constrained M2M communication in cellular networks. At the time of writing, these transports are:

- o TCP, TLS and Websockets [[RFC8323](#)]

- o SMS for cellular networks[I-D.becker-core-coap-sms-gprs]
- o SLIP for serial interfaces[I-D.bormann-t2trg-slipmux]

CoAP uses a REST-based model similar to HTTP, where URIs are used to identify resources at servers. An important factor of allowing CoAP communication over alternative transports, is to express not only the resource identifier, but also the alternative transport information in the URI.

CoAP URIs contain information, such as the endpoint address as well as the location of the resource hosted at the endpoint. CoAP URIs beginning with "coap://" are using UDP, while those beginning with "coaps://" are using DTLS.

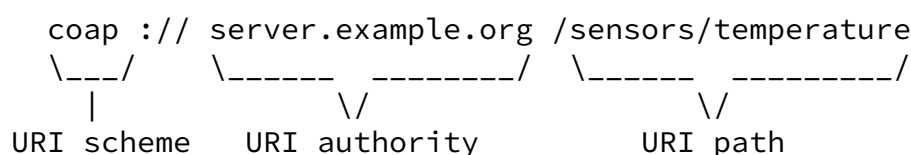


Figure 1: A CoAP URI

Figure 1 shows the structure of a simple example CoAP URI, in which the various URI components can be interpreted as follows:

- o The URI scheme component (e.g. "coap") contains an application-level identifier which typically identifies the protocol being used as well as its transport and network level protocol configurations. Such configurations are defined by convention or standardisation of the protocol using the scheme.
- o The URI authority component ("server.example.com") contains the endpoint identification, which is typically a fully qualified domain name or a network-level host address.
- o The URI path component ("/sensors/temperature") contains a

parameterised resource identifier providing the location and identity of the resource at the endpoint.

In addition to these URI components, Figure 2 shows how specific queries on resource representations are provided by CoAP clients to servers, by specifying one or more URI query components in the URI.

```
coap :// server.example.org /sensors/temperature ?u=cel
                                     \___/
                                     |
                                     URI query
```

Figure 2: A CoAP URI with query

This document focuses on how CoAP URIs can be extended to contain information about alternative transports. For deriving the new URI format, the main design considerations are presented in the next section. Following that, various potential URIs are presented. These URIs provide examples of how transport identifiers can be situated in the URI scheme, authority, path or query components. The proposed URIs are analysed to select feasible formats while disqualifying those not meeting the design criteria.

2. Conformance and Design Considerations

In order to understand which URI formats are best suited for expressing transport information, certain considerations firstly need to be taken into account. Doing so eliminates URI formats that do not meet or conform to the stated requirements. The main criteria are:

1. Conformance to the generic syntax for a URI described in [\[RFC3986\]](#). A URI format needs to be described in which each URI component clearly meets the syntax and percent-encoding rules described.

2. Alignment with best practices for URI design, as described in [\[RFC7320\]](#). This is particularly important when it pertains to establishing or standardising the structure and usage of URIs with respect to the various URI components.
3. Request messages sent to a CoAP endpoint using a CoAP Transport URI may be responded to with a relative URI reference. [\[RFC3986\]](#) provides an algorithm to establish how relative references can be resolved against a base URI to obtain a target URI. Given this algorithm, a URI format needs to be described in which relative reference resolution does not result in a target URI that loses its transport-specific information
4. The URI can be supplied as a Proxy-Uri option by a CoAP end-point to a CoAP forward proxy. This allows communication with a CoAP end-point residing in a network using a different transport. [Section 6.4 of \[RFC7252\]](#) provides an algorithm for parsing a received URI to obtain the request's options. Conformance to

[RFC3986] is also necessary in order for the parsing algorithm to be successful.

In addition to the above mentioned requirements, where possible, the following considerations need to be borne in mind:

1. The URI format is able to represent a resource and the transport information for use in constrained environments, without requiring the presence of a naming infrastructure, such as DNS or a directory/lookup service.
2. Alternative transport information can be easily retrieved by computationally constrained nodes. In other words, the URI format does not result in unnecessarily complex code or logic in such nodes to parse and extract the transport to be used, nor the endpoint address.
3. URIs are designed to uniquely identify resources. When a single resource is represented with multiple URIs, URI aliasing [\[WWWArchv1\]](#) occurs. Avoiding URI aliasing is considered good practice.

4. CoAP URIs do not support fragment identifiers.

3. Situating Transport Information in CoAP URIs

The following subsections aim to describe potential URI formats in which the alternative transport information is placed in various URI components.

3.1. Using the URI scheme component

Expressing the transport information in the URI scheme component can be achieved by using new schemes. These can conform to an agreed-upon convention such as "coap+alternative_transport_name" for each new alternative transport and/or "coaps+alternative_transport_name" for its secure counterpart.

Examples of such URIs are:

- o coap+tcp://server.example.org/sensors/temperature for using CoAP over TCP
- o coap+sms://0015105550101/sensors/temperature for using CoAP over SMS with the endpoint identifier being a telephone subscriber number

- o coaps+tcp://server.example.org/sensors/temperature for using CoAP over TLS

3.1.1. Analysis

Expressing transport information in the URI scheme delivers a URI which is human-readable and computationally as easy to parse as standard CoAP URIs, to extract transport identification information. The URI syntax conforms to [\[RFC3986\]](#), and relative URI resolution does not result in the loss of transport identification information. However, each new alternative transport requires minting new schemes, and IANA intervention is required for the registration of each scheme name. The registration process follows the guidelines stipulated in [\[RFC7595\]](#). Additionally, should a CoAP server wish to expose its resources over multiple transports (such as both UDP and TCP) , URI

aliasing can occur if the URI scheme components of these multiple URIs differ in describing the same resource.

[3.2.](#) Using the URI authority component

Expressing the transport information within the authority component can result in two possible URI formats.

The first approach is to structure the URI authority's host sub-component with a transport prefix to the endpoint identifier and a delimiter, such as "<transport-name>-endpoint_identifier".

Examples of resulting URIs are:

- o `coap://tcp-server.example.org/sensors/temperature` for using CoAP over TCP
- o `coap://sms-0015105550101/sensors/temperature` for using CoAP over SMS

The second approach is to hint at the alternative transport information, by explicitly specifying using the URI authority's port sub-component, thereby differentiating them from standard CoAP URIs.

Examples of resulting URIs are:

- o `coap://server.example.org:5684/sensors/temperature` for using CoAP over TLS
- o `coap://server.example.org:80/sensors/temperature` for using CoAP over WebSockets

[3.2.1.](#) Analysis

Embedding the transport information in the host would violate the guidelines for the structure of URI authorities in [section 2.2 of \[RFC7320\]](#). Consequently, the host in a URI authority component cannot be used as a basis for a new CoAP URI for alternative transports.

Embedding the transport information in the port, on the other hand, would not violate the guidelines for the structure of URI authorities in [section 2.2 of \[RFC7320\]](#). It would result in a CoAP URI that is less human-readable, but URI aliasing is minimised.

On the other hand, if a CoAP request message using a CoAP Transport URI of this form elicits a CoAP Response containing a relative URI, for example, of the form `"/server2.example.org/path/to/another/resource"`, relative URI resolution rules of [\[RFC3986\]](#) would result in the loss of transport identification information. Consequently, using the URI authority component cannot be used as a basis for a new CoAP URI for alternative transports.

[3.3.](#) Using the URI path component

Should the URI path component be used, then special characters or keywords need to be supplied in the path to make the transport explicit. Here, many proposals can exist. In general however, this will result in a URI format such as:

- o `coap://server.example.org/sensors/temperature;tcp` for using CoAP over TCP, by appending the transport information at the end of the URI.

[3.3.1.](#) Analysis

Embedding the transport information in the URI path directly results in a URI that is human-readable. However, if a CoAP request message using a CoAP Transport URI of this form elicits a CoAP Response containing a relative URI, for example, of the form `"../..../path/to/another/resource"`, relative URI resolution rules of [\[RFC3986\]](#) would result in the loss of transport identification information. Consequently, using the URI path component cannot be used as a basis for a new CoAP URI for alternative transports.

[3.4.](#) Using the URI query component

The alternative transport information, should URI query components be used, would result in a URI format such as:

- o `coap://server.example.org/sensors/temperature?alternative-`

transport=wss for using CoAP over secure WebSockets.

[3.4.1.](#) Analysis

Embedding the transport information in a URI query also results in a URI that is human-readable. However, if a CoAP request message using a CoAP Transport URI of this form elicits a CoAP Response containing a relative URI, for example, of the form "../..path/to/another/resource", relative URI resolution rules of [\[RFC3986\]](#) would result in the loss of transport identification information. Consequently, using the URI query component cannot be used as a basis for a new CoAP URI for alternative transports.

[4.](#) Discussion

Based on the analysis of the various options for embedding alternative transport information in a CoAP URI, the most technically feasible option is to use the URI scheme component, as described in [Section 3.1](#). To date, this has also been the WG consensus.

A discussion with IESG members during review of [\[RFC8323\]](#) revealed however, that using the URI scheme to express transport information is not desirable, to avoid the proliferation of new URI schemes for the same application-layer protocol. A strategy was instead proposed to preserve the existing CoAP URI and reuse it for alternative transports, by employing a combination of UDP Confirmable messages and timeouts to determine the eventual correct transport to use between a client and server [\[IESG-feedback\]](#). The undertaken strategy would have obvious implications regarding interoperability, application and protocol logic, resource usage, for both new CoAP and existing CoAP implementations and deployments. Although URI aliasing can theoretically be avoided with this approach, at the time of writing, its technical feasibility over using the simpler strategy of using URI schemes, has yet to be validated. An obvious drawback is therefore that implementers and other SDOs may choose to provisionally or permanently register new URI schemes with IANA, for CoAP over alternative transports anyway, as was done by the Open Connectivity Foundation (OCF) [\[CoAP-TCP-TLS-registration\]](#).

[5.](#) IANA Considerations

This memo includes no request to IANA.

6. Security Considerations

New security risks are not envisaged to arise from the guidelines given in this document, for describing a new URI format containing transport identification within the URI scheme component. However, when specific alternative transports are selected for implementing support for carrying CoAP messages, risk factors or vulnerabilities can be present. Examples include privacy trade-offs when MAC addresses or phone numbers are supplied as URI authority components, or if specific URI path components employed for security-specific interpretations are accidentally encountered as false positives. While this document does not make it mandatory to introduce a security mode with each transport, it recommends ascribing meaning to the use of "coap+" and "coaps+" prefixes in the scheme component, with the "coaps+" prefix used for secure transports for CoAP messages.

7. Acknowledgements

Email discussions, comments and ideas from Thomas Fossati, Akbar Rahman, Klaus Hartke, Martin Thomson, Mark Nottingham, Dave Thaler, Graham Klyne, Carsten Bormann and Markus Becker greatly helped previous versions of this draft.

8. References

8.1. Normative References

- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.
- [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>>.
- [RFC7320] Nottingham, M., "URI Design and Ownership", [BCP 190](#), [RFC 7320](#), DOI 10.17487/RFC7320, July 2014, <<https://www.rfc-editor.org/info/rfc7320>>.

8.2. Informative References

- [CoAP-TCP-TLS-registration]
, <<https://www.iana>>.

Internet-Draft

CoAP Alternative Transports

March 2018

[I-D.becker-core-coap-sms-gprs]

Kuladinithi, K., Becker, M., Li, K., and T. Poetsch, "Transport of CoAP over SMS", [draft-becker-core-coap-sms-gprs-06](#) (work in progress), February 2017.

[I-D.bormann-t2trg-slipmux]

Bormann, C. and T. Kaupat, "Slipmux: Using an UART interface for diagnostics, configuration, and packet transfer", [draft-bormann-t2trg-slipmux-02](#) (work in progress), January 2018.

[IESG-feedback]

, <<https://www.ietf.org/mail-archive/web/core/current/msg08768>>.

[RFC2609] Guttman, E., Perkins, C., and J. Kempf, "Service Templates and Service: Schemes", [RFC 2609](#), DOI 10.17487/RFC2609, June 1999, <<https://www.rfc-editor.org/info/rfc2609>>.

[RFC7595] Thaler, D., Ed., Hansen, T., and T. Hardie, "Guidelines and Registration Procedures for URI Schemes", [BCP 35](#), [RFC 7595](#), DOI 10.17487/RFC7595, June 2015, <<https://www.rfc-editor.org/info/rfc7595>>.

[RFC8323] Bormann, C., Lemay, S., Tschofenig, H., Hartke, K., Silverajan, B., and B. Raymor, Ed., "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets", [RFC 8323](#), DOI 10.17487/RFC8323, February 2018, <<https://www.rfc-editor.org/info/rfc8323>>.

[WWWArchv1]

, <<http://www.w3>>.

[Appendix A](#). Expressing transport in the URI in other ways

Other means of indicating the transport as a distinguishable component within the CoAP URI are possible, but have been deemed unsuitable by not meeting the design considerations listed, or are incompatible with existing practices outlined in [[RFC7252](#)]. They are

however, retained in this section for historical documentation and completeness.

[A.1.](#) Transport information as part of the URI authority

A single URI scheme, "coap-at" can be introduced, as part of an absolute URI which expresses the transport information within the authority component. One approach is to structure the component with

a transport prefix to the endpoint identifier and a delimiter, such as "<transport-name>-endpoint_identifier".

Examples of resulting URIs are:

- o coap-at://tcp-server.example.com/sensors/temperature
- o coap-at://sms-0015105550101/sensors/temperature

An implementation note here is that some generic URI parsers will fail when encountering a URI such as "coap-at://tcp-[2001:db8::1]/sensors/temperature". Consequently, an equivalent, but parseable URI from the ip6.arpa domain needs to be formulated instead. For [2001:db8::1] using TCP, this would result in the following URL:

```
coap-at://tcp-1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0
.1.0.0.2.ip6.arpa:5683/sensors/temperature
```

Usage of an IPv4-mapped IPv6 address such as [::ffff.192.100.0.1] can similarly be expressed with a URI from the ip6.arpa domain.

This URI format allows the usage of a single scheme to represent multiple types of transport end-points. Consequently, it requires consistency in ensuring how various transport-specific endpoints are identified, as a single URI format is used. Attention must be paid towards the syntax rules and encoding for the URI host component. Additionally, against a base URI of the form "coap-at://tcp-server.example.com/sensors/temperature", resolving a relative reference, such as "//example.net/sensors/temperature" would result in the target URI "coap-at://example.net/sensors/temperature", in which transport information is lost.

[A.1.1.](#) Usage of DNS records

DNS names can be used instead of IPv6 address literals to mitigate lengthy URLs referring to the ip6.arpa domain, if usage of DNS is possible.

DNS SRV records can also be employed to formulate a URL such as:

```
coap-at://srv-_coap._tcp.example.com/sensors/temperature
```

in which the "srv" prefix is used to indicate that a DNS SRV lookup should be used for _coap._tcp.example.com, where usage of CoAP over TCP is specified for example.com, and is eventually resolved to a numerical IPv4 or IPv6 address.

[A.2.](#) Making CoAP Resources Available over Multiple Transports

The CoAP URI used thus far is as follows:

```
URI          = scheme ":" hier-part [ "?" query ]
hier-part    = "://" authority path-abempty
```

A new URI format could be introduced, that does not possess an "authority" component, and instead defining "hier-part" to instead use another component, "path-rootless", as specified by [RFC3986](#) [[RFC3986](#)]. The partial ABNF format of this URI would then be:

```
URI          = scheme ":" hier-part [ "?" query ]
hier-part    = path-rootless
path-rootless = segment-nz *( "/" segment )
```

The full syntax of "path-rootless" is described in [[RFC3986](#)]. A generic URI defined this way would conform to the syntax of [[RFC3986](#)], while the path component can be treated as an opaque string to indicate transport types, endpoints as well as paths to CoAP resources. A single scheme can similarly be used.

A constrained node that is capable of communicating over several types of transports (such as UDP, TCP and SMS) would be able to convey a single CoAP resource over multiple transports. This is also

beneficial for nodes performing caching and proxying from one type of transport to another.

Requesting and retrieving the same CoAP resource representation over multiple transports could be rendered possible by prefixing the transport type and endpoint identifier information to the CoAP URI. This would result in the following example representation:

```
coap-at:tcp://example.com?coap://example.com/sensors/temperature
      \-----/ \-----/
        \/\
        Transport-specific      CoAP Resource
          Prefix
```

Figure 3: Prefixing a CoAP URI with TCP transport

Such a representation would result in the URI being decomposed into its constituent components, with the CoAP resource residing within the query component as follows:

Scheme: coap-at

Path: tcp://example.com

Query: coap://example.com/sensors/temperature

The same CoAP resource, if requested over a WebSocket transport, would result the following URI:

```
coap-at:ws://example.com/endpoint?coap://example.com/sensors/temperature
      \-----/ \-----/
        \/\
        Transport-specific      CoAP Resource
          Prefix
```

Figure 4: Prefixing a CoAP URI with WebSocket transport

While the transport prefix changes, the CoAP resource representation remains the same in the query component:

Scheme: coap-at

Path: `ws://example.com/endpoint`

Query: `coap://example.com/sensors/temperature`

The URI format described here overcomes URI aliasing [[WWWArchv1](#)] when multiple transports are used, by ensuring each CoAP resource representation remains the same, but is prefixed with different transports. However, against a base URI of this format, resolving relative references of the form `"/example.net/sensors/temperature"` and `"/sensor2/temperature"` would again result in target URIs which lose transport-specific information.

Implementation note: While square brackets are disallowed within the path component, the '[' and ']' characters needed to enclose a literal IPv6 address can be percent-encoded into their respective equivalents. The ':' character does not need to be percent-encoded. This results in a significantly simpler URI string compared to [section 2.2](#), particularly for compressed IPv6 addresses. Additionally, the URI format can be used to specify other similar address families and formats, such as Bluetooth addresses.

[A.3](#). Transport as part of a 'service:' URL scheme

The "service:" URL scheme name was introduced in [[RFC2609](#)] and forms the basis of service description used primarily by the Service Location Protocol. An abstract service type URI would have the form `"service:<abstract-type>:<concrete-type>"`

where `<abstract-type>` refers to a service type name that can be associated with a variety of protocols, while the `<concrete-type>`

then providing the specific details of the protocol used, authority and other URI components.

Adopting the "service:" URL scheme to describe CoAP usage over alternative transports would be rather trivial. To use a previous example, a CoAP service to discover a Resource Directory and its base RD resource using TCP would take the form

`service:coap:tcp://host.example.com/.well-known/core?rt=core-rd`

The syntax of the "service:" URL scheme differs from the generic URI

syntax and therefore such a representation should be treated as an opaque URI as [Section 2.1 of \[RFC2609\]](#) recommends.

Authors' Addresses

Bilhanan Silverajan
Tampere University of Technology
Korkeakoulunkatu 10
FI-33720 Tampere
Finland

Email: bilhanan.silverajan@tut.fi

Teemu Savolainen
Nokia Technologies
Hatanpäään valtatie 30
FI-33100 Tampere
Finland

Email: teemu.savolainen@nokia.com