

CoRE  
Internet-Draft  
Intended status: Standards Track  
Expires: January 3, 2019

K. Lynn  
P. van der Stok  
Consultants  
M. Koster  
SmartThings  
C. Amsuess  
Energy Harvesting Solutions  
July 02, 2018

**CoRE Resource Directory: DNS-SD mapping  
draft-ietf-core-rd-dns-sd-02**

**Abstract**

Resource and service discovery are complimentary. Resource discovery provides fine-grained detail about the content of a server, while service discovery can provide a scalable method to locate servers in large networks. This document defines a method for mapping between CoRE Link Format attributes and DNS-Based Service Discovery fields to facilitate the use of either method to locate RESTful service interfaces (APIs) in heterogeneous HTTP/CoAP environments.

**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 January 3, 2019.

**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 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>	Terminology . . . . .	<a href="#">3</a>
<a href="#">1.2.</a>	CoRE Resource Discovery . . . . .	<a href="#">3</a>
<a href="#">1.3.</a>	CoRE Resource Directories . . . . .	<a href="#">4</a>
<a href="#">1.4.</a>	DNS-Based Service Discovery . . . . .	<a href="#">5</a>
<a href="#">2.</a>	New Link-Format Attributes . . . . .	<a href="#">6</a>
<a href="#">2.1.</a>	Resource Instance attribute "ins" . . . . .	<a href="#">6</a>
<a href="#">2.2.</a>	Export attribute "exp" . . . . .	<a href="#">7</a>
<a href="#">3.</a>	Mapping CoRE Link Attributes to DNS-SD Record Fields . . . . .	<a href="#">7</a>
<a href="#">3.1.</a>	Mapping Resource Instance attribute "ins" to <Instance> . . . . .	<a href="#">7</a>
<a href="#">3.2.</a>	Mapping Resource Type attribute "rt" to <ServiceType> . . . . .	<a href="#">7</a>
<a href="#">3.3.</a>	Domain mapping . . . . .	<a href="#">8</a>
<a href="#">3.4.</a>	TXT Record key=value strings . . . . .	<a href="#">8</a>
<a href="#">3.5.</a>	Importing resource links into DNS-SD . . . . .	<a href="#">9</a>
<a href="#">4.</a>	IANA considerations . . . . .	<a href="#">9</a>
<a href="#">5.</a>	Security considerations . . . . .	<a href="#">9</a>
<a href="#">6.</a>	References . . . . .	<a href="#">10</a>
<a href="#">6.1.</a>	Normative References . . . . .	<a href="#">10</a>
<a href="#">6.2.</a>	Informative References . . . . .	<a href="#">11</a>
	Acknowledgments . . . . .	<a href="#">11</a>
	Authors' Addresses . . . . .	<a href="#">11</a>

## [1.](#) Introduction

The Constrained RESTful Environments (CoRE) working group aims at realizing the REST architecture in a suitable form for the most constrained devices (e.g. 8-bit microcontrollers with limited RAM and ROM) and networks (e.g. 6LoWPAN). CoRE is aimed at machine-to-machine (M2M) applications such as smart energy and building automation. The main deliverable of CoRE is the Constrained Application Protocol (CoAP) specification [[RFC7252](#)].

Automated discovery of resources hosted by a constrained server is critical in M2M applications where human intervention is minimal and static interfaces result in brittleness. CoRE Resource Discovery is intended to support fine-grained discovery of hosted resources, their attributes, and possibly other resource relations [[RFC6690](#)].







In contrast to resource discovery, service discovery generally refers to a coarser-grained resolution of an endpoint's IP address, port number, and protocol. This definition may be extended to include multi-function devices, where the result of the discovery process may include a path to a resource representing a RESTful service interface and possibly a reference to a description of the interface such as a JSON Hyper-Schema document [[I-D.handrews-json-schema-hyperschema](#)] per function.

Resource and service discovery are complimentary in the case of large networks, where the latter can facilitate scaling. This document defines a mapping between CoRE Link Format attributes and DNS-Based Service Discovery (DNS-SD) [[RFC6763](#)] fields that permits discovery of CoAP services by either method. It also addresses the CoRE charter goal to interoperate with DNS-SD.

### **[1.1.](#) 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 [[RFC2119](#)]. The term "byte" is used in its now customary sense as a synonym for "octet".

This specification requires readers to be familiar with all the terms and concepts that are discussed in [[RFC6690](#)] and [[RFC8288](#)]. Readers should also be familiar with the terms and concepts discussed in [[RFC7252](#)]. To describe the REST interfaces defined in this specification, the URI Template format is used [[RFC6570](#)].

This specification also incorporates the terminology of [[I-D.ietf-core-resource-directory](#)].

### **[1.2.](#) CoRE Resource Discovery**

[[RFC8288](#)] defines a Web Link (link) as a typed connection between two resources, comprised of:

- o a link context, o a link relation type (see [Section 2.1 of \[\[RFC8288\]\(#\)\]](#), o a link target, and o optionally, target attributes (see [Section 2.2 of \[\[RFC8288\]\(#\)\]](#)).

A link can be viewed as a statement of the form "link context has a link relation type resource at link target, which (optionally) has target attributes", where link target (and context) is typically a Universal Resource Identifier (URI [[RFC3986](#)]).







For example, "https://www.example.com/" has a "canonical" resource at "https://example.com", which has a "type" of "text/html".

The main function of Resource Discovery is to provide links for the resources hosted by the server, complemented by attributes about those resources and perhaps additional link relations. In CoRE this collection of links and attributes is itself a resource (as opposed to HTTP headers delivered with a specific resource).

[RFC6690] specifies a link format for use in CoRE Resource Discovery by extending the HTTP Link Header Format [[RFC8288](#)] to describe these link descriptions. The CoRE Link Format is carried as a payload and is assigned an Internet media type. CoRE Resource Discovery is accomplished by sending a GET request to the well-known URI "/.well-known/core", which is defined as a default entry-point for requesting the collection of links about resources hosted by a server.

Resource Discovery can be performed either via unicast or multicast. When a server's IP address is already known, either a priori or resolved via the Domain Name System (DNS) [[RFC1034](#)][RFC1035], unicast discovery is performed in order to locate a URI for the resource of interest. This is performed using a GET to /.well-known/core on the server, which returns a payload in the CoRE Link Format. A client would then match the appropriate Resource Type, Interface Description, and possible Content-Type [[RFC2045](#)] for its application. These attributes may also be included in the query string in order to filter the number of links returned in a response.

### **1.3. CoRE Resource Directories**

In many M2M scenarios, direct discovery of resources is not practical due to sleeping nodes, limited bandwidth, or networks where multicast traffic is inefficient. These problems can be solved by deploying a network element called a Resource Directory (RD), which hosts descriptions of resources held on other servers (referred to as "end-points") and allows lookups to be performed for those resources. An endpoint is a web server associated with specific IP address and port; thus a physical device may host one or more endpoints. End-points may also act as clients.

The Resource Directory implements a set of REST interfaces for end-points to register and maintain collections of links, called resource directory entries. [[I-D.ietf-core-resource-directory](#)] specifies the web interfaces that an RD supports in order for web servers to discover the RD and to register, maintain, lookup and remove resource descriptions; for the RD to validate entries; and for clients to lookup resources from the RD. Furthermore, new link attributes useful in conjunction with an RD are defined.







#### **1.4. DNS-Based Service Discovery**

DNS-Based Service Discovery (DNS-SD) defines a conventional method of naming and configuring DNS PTR, SRV, and TXT resource records to facilitate discovery of services (such as CoAP servers in a subdomain) using the existing DNS infrastructure. This section gives a brief overview of DNS-SD; see [[RFC6763](#)] for a detailed specification.

DNS-SD Service Names are limited to 255 bytes and are of the form:

Service Name = <Instance>.<ServiceType>.<Domain>

The Service Name identifies a SRV/TXT resource record (RR) pair. The SRV RR specifies the host and the port of the endpoint. The TXT RR provides additional information in the form of key/value pairs. DNS-Based Service Discovery is accomplished by sending a DNS request for PTR records with the name <ServiceType>.<Domain>, which will return a list of zero or more Service Names.

The <Domain> part of the Service Name is identical to the global (DNS subdomain) part of the authority in URIs that identify the resources on an individual server or group of servers.

The <ServiceType> part is composed of at least two labels. The first label of the pair is the application protocol name [[RFC6335](#)] preceded by an underscore character. For example, an organization such as the Open Connectivity Foundation (OCF) that specifies resources [ref?] might register the application protocol name "\_oic", which all servers that advertise OCF resources would use as part of their ServiceType. The second label indicates the transport and is typically "\_udp" for CoAP services. In cases where narrowing the scope of the search may be useful, these labels may be optionally preceded by a subtype name followed by the "\_sub" label. An example of this more specific <ServiceType> is "light.\_sub.\_oic.\_udp".

The default <Instance> part of the Service Name SHOULD be set to a default value at the factory and MAY be modified during the commissioning process. It SHOULD uniquely identify an instance of <ServiceType> within a <Domain>. Taken together, these three elements comprise a unique name for an SRV/TXT record pair within the DNS subdomain.

The granularity of a Service Name MAY be that of a host or group, or it could represent a particular resource within a CoAP server. The SRV record contains the host name (AAAA record name) and port of the service while protocol is part of the Service Name. In the case







where a Service Name identifies a particular resource, the path part of the URI must be carried in a corresponding TXT record.

A DNS TXT record is in practice limited to a few hundred bytes in length, which is indicated in the resource record header in the DNS response message [RFC6763]. The data consists of one or more strings comprising a key/value pair. By convention, the first pair is txtver=<number> (to support different versions of a service description). An example string is:

```
-----  
| 0x08 | t | x | t | v | e | r | = | 1 |  
-----
```

## 2. New Link-Format Attributes

When using the CoRE Link Format to describe resources being discovered by or posted to a resource directory service, additional information about those resources is useful. This specification defines the following new attributes for use in the CoRE Link Format [RFC6690]:

```
link-extension    = ( "ins" "=" (ptoken | quoted-string) )  
                  ; The token or string is max 63 bytes  
link-extension    = ( "exp" )
```

### 2.1. Resource Instance attribute "ins"

The Resource Instance "ins" attribute is an identifier for this resource, which makes it possible to distinguish it from other similar resources. This attribute is equivalent in use to the <Instance> portion of a DNS-SD record (see [Section 1.4](#)), and SHOULD be unique across resources with the same Resource Type attribute in the domain in which it is used. A Resource Instance SHOULD be a descriptive string like "Ceiling Light, Room 3", but MAY be a short ID like "AF39" or a unique UUID. This attribute is used by a Resource Directory to distinguish between multiple instances of the same resource type within the directory.

This attribute MUST NOT be more than 63 bytes in length. The resource identifier attribute MUST NOT appear more than once in a link description. This attribute MAY be used as a query parameter in the RD Lookup Function Set defined in Section 7 of [\[I-D.ietf-core-resource-directory\]](#).







## **2.2. Export attribute "exp"**

The Export "exp" attribute is used as a flag to indicate that a link description MAY be exported from a resource directory to external directories.

The CoRE Link Format is used for many purposes between CoAP endpoints. Some are useful mainly locally; for example checking the observability of a resource before accessing it, determining the size of a resource, or traversing dynamic resource structures. However, other links are very useful to be exported to other directories, for example the entry point resource to a functional service. This attribute MAY be used as a query parameter in the RD Lookup Function Set defined in Section 7 of [[I-D.ietf-core-resource-directory](#)].

## **3. Mapping CoRE Link Attributes to DNS-SD Record Fields**

### **3.1. Mapping Resource Instance attribute "ins" to <Instance>**

The Resource Instance "ins" attribute maps to the <Instance> part of a DNS-SD Service Name. It is stored directly in the DNS as a single DNS label of canonical precomposed UTF-8 [[RFC3629](#)] "Net-Unicode" (Unicode Normalization Form C) [[RFC5198](#)] text. However, if the "ins" attribute is chosen to match the DNS host name of a service, it SHOULD use the syntax defined in [Section 3.5 of \[RFC1034\]](#) and [Section 2.1 of \[RFC1123\]](#).

The <Instance> part of the name of a service being offered on the network SHOULD be configurable by the user setting up the service, so that he or she may give it an informative name. However, the device or service SHOULD NOT require the user to configure a name before it can be used. A sensible choice of default name can allow the device or service to be accessed in many cases without any manual configuration at all (see [Appendix D of \[RFC6763\]](#)).

DNS labels are limited to 63 bytes in length and the entire Service Name may not exceed 255 bytes.

### **3.2. Mapping Resource Type attribute "rt" to <ServiceType>**

The <ServiceType> part of a DNS-SD Service Name is derived from the "rt" attribute and SHOULD conform to the reg-rel-type production of the Link Format defined in [Section 2 of \[RFC6690\]](#).

In practice, the ServiceType should unambiguously identify inter-operable devices. It is up to individual standards bodies to specify how to map between their registered Resource Type (rt=) values and ServiceType values. Two approaches are possible; either a







hierachical approach as in [Section 1.4](#) above, or a flat identifier. Both approaches are shown below for illustration, but in practice only ONE would be specified.

In either case, the resulting application protocol name MUST be composed of at least a single Net-Unicode text string, without underscore '\_' or or period '.' and limited to 15 bytes in length (see [Section 5.1 of \[RFC6335\]](#)). This string is mapped to the DNS-SD <ServiceType> by prepending an underscore and appending a period followed by the "\_udp" label. For example, rt="oic.d.light" might be mapped into "\_oic-d-light.\_udp".

The application protocol name may be optionally followed by a period and a service subtype name consisting of a Net-Unicode text string, without underscore or period and limited to 63 bytes. This string is mapped to the DNS-SD <ServiceType> by appending a period followed by the "\_sub" label and then appending a period followed by the service type label pair derived as in the previous paragraph. For example, rt="oic.d.light" might be mapped into "light.\_sub.\_oic.\_udp".

The resulting string is used to form labels for DNS-SD records which are stored directly in the DNS.

### **3.3. Domain mapping**

TBD: A method must be specified to determine in which DNS zone the CoAP service should be registered. See, for example, [Section 11 in \[RFC6763\]](#).

### **3.4. TXT Record key=value strings**

A number of [\[RFC6763\]](#) key/value pairs are derived from link-format information, to be exported in the DNS-SD as key=value strings in a TXT record (See [Section 6.3 of \[RFC6763\]](#)).

The resource <URI> is exported as key/value pair "path=<URI>".

The Interface Description "if" attribute is exported as key/value pair "if=<Interface Description>".

The DNS TXT record can be further populated by importing any other resource description attributes as they share the same key=value format specified in [Section 6 of \[RFC6763\]](#).







### 3.5. Importing resource links into DNS-SD

Assuming the ability to query a Resource Directory or multicast a GET (?exp) over the local link, CoAP resource discovery may be used to populate the DNS-SD database in an automated fashion. CoAP resource descriptions (links) can be exported to DNS-SD for exposure to service discovery by using the Resource Instance attribute as the basis for a unique Service Name, composed with the Resource Type as the <ServiceType>, and registered in the correct <Domain>. The agent responsible for exporting records to the DNS zone file SHOULD be authenticated to the DNS server. The following example, using the example lookup location /rd-lookup, shows an agent discovering a resource to be exported:

```
Req: GET /rd-lookup/res?exp
```

```
Res: 2.05 Content
```

```
<coap://[FDFD::1234]:5683/light/1>;  
exp;rt="oic.d.light";ins="Spot";  
d="office";ep="node1"
```

The agent subsequently registers the following DNS-SD RRs, assuming a zone name "example.com" prefixed with "office":

```
_oic._udp.office.example.com      IN PTR  
    Spot._oic._udp.office.example.com  
light._sub._oic._udp.example.com  IN PTR  
    Spot._oic._udp.office.example.com  
Spot._oic._udp.office.example.com IN TXT  
    txtver=1;path=/light/1  
Spot._oic._udp.office.example.com IN SRV 0 0 5683  
    node1.office.example.com.  
node1.office.example.com.         IN AAAA FDFD::1234
```

In the above figure the Service Name is chosen as Spot.\_oic.\_udp.office.example.com without the light.\_sub service prefix. An alternative Service Name would be: Spot.light.\_sub.\_oic.\_udp.office.example.com.

## 4. IANA considerations

TBD

## 5. Security considerations

TBD







## 6. References

### 6.1. Normative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, [RFC 1034](#), DOI 10.17487/RFC1034, November 1987, <<http://www.rfc-editor.org/info/rfc1034>>.
- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, [RFC 1035](#), DOI 10.17487/RFC1035, November 1987, <<https://www.rfc-editor.org/info/rfc1035>>.
- [RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, [RFC 1123](#), DOI 10.17487/RFC1123, October 1989, <<http://www.rfc-editor.org/info/rfc1123>>.
- [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>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, [RFC 3629](#), DOI 10.17487/RFC3629, November 2003, <<http://www.rfc-editor.org/info/rfc3629>>.
- [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>>.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", [RFC 5198](#), DOI 10.17487/RFC5198, March 2008, <<http://www.rfc-editor.org/info/rfc5198>>.
- [RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", [BCP 165](#), [RFC 6335](#), DOI 10.17487/RFC6335, August 2011, <<http://www.rfc-editor.org/info/rfc6335>>.







- [RFC6570] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", [RFC 6570](#), DOI 10.17487/RFC6570, March 2012, <<http://www.rfc-editor.org/info/rfc6570>>.
- [RFC6690] Shelby, Z., "Constrained RESTful Environments (CoRE) Link Format", [RFC 6690](#), DOI 10.17487/RFC6690, August 2012, <<http://www.rfc-editor.org/info/rfc6690>>.
- [RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", [RFC 6763](#), DOI 10.17487/RFC6763, February 2013, <<http://www.rfc-editor.org/info/rfc6763>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", [RFC 7252](#), DOI 10.17487/RFC7252, June 2014, <<http://www.rfc-editor.org/info/rfc7252>>.
- [RFC8288] Nottingham, M., "Web Linking", [RFC 8288](#), DOI 10.17487/RFC8288, October 2017, <<https://www.rfc-editor.org/info/rfc8288>>.

## **6.2. Informative References**

- [I-D.handrews-json-schema-hyperschema]  
Andrews, H. and A. Wright, "JSON Hyper-Schema: A Vocabulary for Hypermedia Annotation of JSON", [draft-handrews-json-schema-hyperschema-01](#) (work in progress), January 2018.
- [I-D.ietf-core-resource-directory]  
Shelby, Z., Koster, M., Bormann, C., Stok, P., and C. Amsuess, "CoRE Resource Directory", [draft-ietf-core-resource-directory-14](#) (work in progress), July 2018.

## Acknowledgments

This document was split out from [[I-D.ietf-core-resource-directory](#)]. Zach Shelby was a co-author of the original version of this draft.

## Authors' Addresses

Kerry Lynn  
Consultant

Phone: +1 978-460-4253  
Email: [kerlyn@ieee.org](mailto:kerlyn@ieee.org)







Peter van der Stok  
Consultant

Phone: +31 492474673 (Netherlands), +33 966015248 (France)  
Email: [consultancy@vanderstok.org](mailto:consultancy@vanderstok.org)  
URI:    [www.vanderstok.org](http://www.vanderstok.org)

Michael Koster  
SmartThings  
665 Clyde Avenue  
Mountain View 94043  
USA

Phone: +1 707-502-5136  
Email: [Michael.Koster@smarththings.com](mailto:Michael.Koster@smarththings.com)

Christian Amsuess  
Energy Harvesting Solutions  
Hollandstr. 12/4  
1020  
Austria

Phone: +43 664-9790639  
Email: [c.amsuess@energyharvesting.at](mailto:c.amsuess@energyharvesting.at)



