Workgroup: Network Working Group
Published: 3 January 2020
Intended Status: Experimental
Expires: 6 July 2020
Authors: J. Schaad
August Cellars

CoAP Application version of Resource Directory

Abstract

This is a draft of what I think a CoRE Application should look like.

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 6 July 2020.

Table of Contents

1. Preamble

This document explores how a CoRE Resource Directory [I-D.ietf-core-resource-directory] might look if based on [CoRAL]. This document is not currently intended as something to be standardized at this time.

2. Introduction

Refer to the introduction of [I-D.ietf-core-resource-directory]. Concise Binary Object Representation (CBOR) [CBOR] is a compact self-describing binary encoding formation that is starting to be used in many different applications. One of the primary uses of CBOR is in the Internet of Things where the constrained nature means that having minimal size of encodings becomes very important. The use of the Cryptographic Message System (CMS) [CMS] is still one of the most common method for providing message-based security, although in many cases the CBOR Object Signing and Encryption (COSE) [COSE] message-based security system is starting to be used. Given that CBOR is going to be transported using CMS, it makes sense to define CMS content types for the purpose of denoting that the embedded content is CBOR. This document defines two new content types: CBOR Content Type and CBOR Sequence Content Type [I-D.ietf-cbor-sequence].

3. Vocabulary

Unless otherwise noted, all of the vocabulary defined in this document are prefixed with "http://jimsch.example.org/rd#". For convience, all item defined in this vocabulary is tagged with **strong**.

3.1. Containers

rd-endpoint This container represents a single endpoint on a resource server. The content module of this container is:

*An **endpointName**. The endpoint name MUST be present for third party registrations and is required for first party registrations unless the RD can infer the endpoint name from the security context.

*An optional **sector**

*An optional **endpointBase** URI. The value is required for third party registrations. For first party registrations, it is inferred from the registration request if not present.

*Zero or more **rfc-item** containers. Each container represents a resource or form on the endpoint. Some actions require that the rfc-items are present, where others will omit them.

rd-item This container represents a single resource that is provided by a server. There is no requirement on the content model for this container type.

rd-linkAttribute This container provides a method of pulling link attributes into the RD content model. The target of the container is the name of the link attribute. The values of the container is **value** with one field occurring for each different value. Unlink

link attributes, space separated values are listed at multiple values.

Where this document as defined an equivalent to a link attribute, that equivalent MUST be used. Where equivalents are defined in other documents, that equivalent SHOULD be used when it is new. Where equivalents are defined in other documents for long standing attributes, the RD SHOULD NOT attempt to map between them but to keep them as they were registered. In this case it is a requirement on the registering agent to ensure that when things are registered both ways they are the same.

rd-group This container allows for grouping together a set of resources. The purpose of the container is to be able to allow for an endpoint to advertise resources at different addresses but associated with that endpoint. Endpoints SHOULD NOT advertise resources on other systems, even if those resources are copies of a resource on the system. Instead, **alternative** should be used for that purpose.

3.2. Leafs

alternative

endpointBase The endpoint base URI of a registration. This represents a URI that typically gives the scheme and authority information about an endpoint. The endpoint base URI is provided at registration or update time, and is used by the RD to resolve relative references when returning resource descriptions. Separating the base URI allows for it to be patched independently of the resource items.

This is equivalent to the base link attribute defined in [I-D.ietf-core-resource-directory].

content-type This is equivalent to the ct link attribute defined in [????].

describedby

endpointName A UTF8 string indicating the name of the endpoint. The endpoint name MUST NOT include characters in the range 0-31 or 127-159.

This is equivalent to the ep link attribute defined in [I-D.ietf-core-resource-directory].

lifetime The lifetime of the registration in seconds. The range of lifetime is 60-294967295. If a registration does not include a life time, it defaults to 90000 (25 hours).

resource-type

sector
: A string indicating the sector to which an endpoint belongs. In the context of a Resource Directory, a sector is a logical grouping of endpoints.

  This is equivalent to the d link attribute defined in [I-D.ietf-core-resource-directory].

title

value This leaf occurs in an **rd-linkAttribute** and holds a single value for a link attribute.

4. Resource Interfaces

rd-endpoint The endpoint interface is used by a client to update or remove an endpoint and its associated resources from the resource directory. The interface supports the following operations:

- *DELETE removes the endpoint representation from the resource directory.

- *GET returns the current set of endpoint attributes and resources for the endpoint. Support of observe is optional for a resource directory.

- *PATCH allows for an incremental update of the attributes and resources of the endpoint.

- *PUT replaces the setup of attributes and resources for the endpoint.

rd-endpointSearch

rd-register The register interface is used by a client to register an endpoint and its associated resources with the ressource directory. This interface is intended both for an endpoint to register itself as well as third party registration of an endpoint.

The registration interface supports one operation: POST. The content of the POST operation is a CoRAL document containing the content of a rd-endpoint. The rd-endpoint container itself MAY be included, but only the content of the container is expected.

Processing a registration request involves the following steps:

1. Perform requisite checks that the party attempting to perform the registration has the permissions to do so.

2. Verify that all required content is present for the endpoint and for each resource to be registered. Part of

this may be to extract the required content from the security context used for determining permissions.

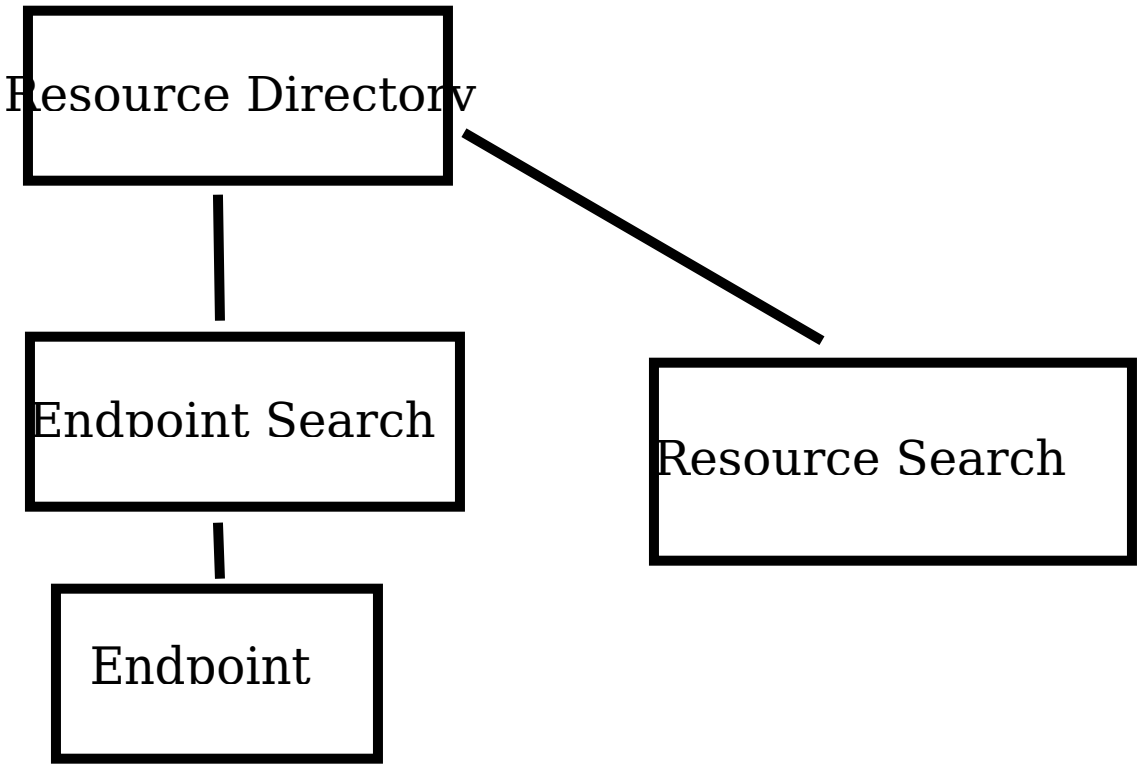
3. If the endpoint name and domain pair map to an existing endpoint registration, that registration is replaced using the same link path. Otherwise a new endpoint registration is created.

rd-resourceSearch The resource search interface is used by clients to locate and retrieve the description of resources based on some criteria.

The resource search interface supports one operation: FETCH. The content of the FETCH operation is a set of search criteria to be matched against all of the resources registered with the RD. The rules for doing a match following the rules in Appendix A.1 with one addition. The container **rd-group** is ignored when doing matching against the criteria. Specifically, the rd-item in the container are always matched against.

If an rd-endpoint is included in the search criteria, then the endpoint which hosts the resource is matched against that criteria.

5. Model Objects



Resource Directory This resource represents the entry point into the Resource Directory. The resource always exists in some form on a resource directory server. The resource will support the GET verb to return a CoRAL document describing where the interfaces on the resource directory can be found.

This resource can additionally support the rd-register and either the rd-endpointSearch or rd-resourceSearch interfaces.

Endpoint Search This resource provides for where an endpoint search can be done. As such, the resource supports the rd-endpoint interface. This resource may additionally support the rd-register interface.

6. Examples

6.1. Interop Items

In order to have interop, a number of items need to be defined. For the example below the following assumptions are made:

*TBD-CoRAL content type is 99599

*TBD-CoRAL-Dict is 99999 (TBD6 in [CoRAL]

*The dictionary used is in Table 1

Key	Value
1	http://jimsch.example.org/rd#content-type
2	http://jimsch.example.org/rd#ace-Profile
3	http://jimsch.example.org/rd#authority-type
4	http://jimsch.example.org/rd#authority
5	http://jimsch.example.org/rd#rd-register
6	http://jimsch.example.org/rd#rd-endpointSearch
7	http://jimsch.example.org/rd#rd-resourceSearch
8	http://jimsch.example.org/rd#ace-Audience

Table 1

6.2. Retrieve Resource Directory Information

```
Request:
GET coap://jimsch.example.org/rd
Accept: TBD-CoRAL

Response:
2.05 Content
Content-Format: TBD-CoRAL

#using <http://jimsch.example.org/rd#>

rd-register </rd/endpoints> [
    content-type TBD-CoRAL
    authority <coap://ace.example.org/token> [
        authority-type "ACE"
        ace-Profile "coap_oscore"
        ace-Audience "jimsch.example.org"
    ]
]

rd-endpointSearch </rd/endpoints>

rd-resourceSearch <rd/resources>

rd-register <coaps://jimsch.example.org/rd/endpoints> [
    content-type TBD-CoRAL
    authority <coap://ace.example.org/token> [
        authority-type "ACE"
        ace-Profile "coap_oscore"
        ace-Profile "coap_dtls"
        ace-Audience "jimsch.example.org"
    ]
    authority _ [
        authority-type "X.509"
    ]
]

#base <coaps://jimsch.example.org/rd>
rd-endpointSearch </rd/endpoints>

rd-resourceSearch </rd/resources>


or

[[2, 5, [5, 2, 6, "endpoints"], [
    [2, 1, 99599],
    [2, 4, [2, "ace.example.org", 6, "token"], [
        [2, 2, "coap_oscore"],
        [2, 3, "ACE"],
        [2, 8, "jimsch.example.org"]
    ]]
 ]],
```

```
[2, 6, [5, 2, 6, "endpoints"]],
[2, 7, [5, 2, 6, "resources"]],
[1, [1, "coaps", 2, "jimsch.example.org", 6, "rd"]],
[2, 5, [5, 2, 6, "endpoints"], [
   [2, 1, 99599],
   [2, 4, [2, "ace.example.org", 6, "token"], [
       [2, 2, "coap_oscore"],
       [2, 2, "coap_dtls"],
       [2, 3, "ACE"],
       [2, 8, "jimsch.example.org"]
   ]]
 ]],
 [2, 6, [5, 2, 6, "endpoints"]],
 [2, 7, [5, 2, 6, "resources"]]]
```

6.3. Registering Endpoints

Sample registration of an endpoint with four resources.

```
POST coaps://jimsch.example.com/rd/endpoints
Content-Format: TBD-CoRAL

rd-endpointName "node1"
rd-base <coaps://[2001:db8:1::1]>
rd-item </sensors/temp> [
    content-type 41
    resource-type "temperature-c"
    rd-linkAttribute "if" [ value "sensor" ]
    describedby <http://www.example.com/sensors/temp>
]
rd-item </temp> [
    content-type 0
    resource-type "temperature"
]
rd-item </light> [
    content-type 0
    resource-type "light-lux"
]
rd-item </t> [
]
```

Example registration of a resource server which exposes resources on multiple addresses. This example was made mechanically from /.well-known/core for my test server, as such it is missing several items which it would normally have dealing with security and other items as there are no uniform link attributes for these features. At some point I might go in and clean this up based on how things are enforced, such as items which cannot be read due to security issues. This example uses the CoRAL content type from [CoRAL].

```
rd-group <coap://server.example.org> [
  rd-item </authz-info>
  rd-item </rd> [
    content-type 40
    content-type 65088
    resource-type "core.rd"
  ]
  rd-item </rd/post2>
  rd-item </rd-lookup>
  rd-item </rd-lookup/ep> [
    content-type 40
    content-type 65088
    resource-type "core.rd-lookup-ep"
  ]
  rd-item </rd-lookup/res> [
    content-type 40
    resource-type "core.rd-lookup-res"
  ]
  rd-item </ace-echo>
  rd-item </ExtraLargeResource> [
    resource-type "BlockWiseTransferTester"
    title "This is a large resource for testing block-wise transfer"
  ]
  rd-item </StorageHere>
  rd-item </oscore> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/LargeResource> [
    resource-type "BlockWiseTransferTester"
    title "This is a large resource for testing block-wise transfer"
  ]
  rd-item </oscore/hello> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/hello/1> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/hello/2> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/hello/3> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/hello/6> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
```

```
  rd-item </oscore/hello/7> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/hello/coap> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/observe1> [
    rd-linkAttribute obs
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/observe2> [
    obs
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </oscore/test> [
    resource-type "OSCOAP-Tester"
    title "GET a friendly greeting!"
  ]
  rd-item </ace>
  rd-item </ace/helloWorld>
  rd-item </ace/lock>
  rd-item </hello> [
    resource-type "HelloWorldDisplayer"
    title "GET a friendly greeting!"
  ]
]
rd-group <coaps://server.example.org> [
  rd-item </authz-info>
  rd-item </rd> [
    content-type 40
    content-type 65088
    resource-type "core.rd"
  ]
  rd-item </rd/post2>
  rd-item </rd-lookup>
  rd-item </rd-lookup/ep> [
    content-type 40
    content-type 65088
    resource-type "core.rd-lookup-ep"
  ]
  rd-item </rd-lookup/res> [
    content-type 40
    resource-type "core.rd-lookup-res"
  ]
  rd-item </ace-echo>
  rd-item </ExtraLargeResource> [
    resource-type "BlockWiseTransferTester"
    title "This is a large resource for testing block-wise transfer"
```

```
]
rd-item </StorageHere>
rd-item </oscore> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/LargeResource> [
  resource-type "BlockWiseTransferTester"
  title "This is a large resource for testing block-wise transfer"
]
rd-item </oscore/hello> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/1> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/2> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/3> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/6> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/7> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/hello/coap> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/observe1> [
  rd-linkAttribute obs
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/observe2> [
  obs
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
rd-item </oscore/test> [
  resource-type "OSCOAP-Tester"
  title "GET a friendly greeting!"
]
```

```
  rd-item </ace>
  rd-item </ace/helloWorld>
  rd-item </ace/lock>
  rd-item </hello> [
    resource-type "HelloWorldDisplayer"
    title "GET a friendly greeting!"
  ]
]
```

6.4. Query Endpoints

```
FETCH coaps://jimsch.example.com/rd/endpoints
Content-Format: TBD-CoRAL

rd-linkAttribute "et" [ value "oic.d.sensor" ]

2.05 Content
Conent-Type: TBD-CoRAL

rd-endpoint <endpoints/1234> [
    endpoint-name "node5"
    resource-type "core.rd-ep"
    rd-linkAttribute "et" [ value "oic.d.sensor" ]
]
rd-endpoint <endpoints/4521> [
    endpoint-name "node7"
    domain "floor-3"
    resource-type "core.rd-ep"
    rd-linkAttribute "et" [ value "oic.d.sensor" ]
]
```

6.5. Query Resources

```
FETCH coaps://jimsch.example.com/rd/resources
Content-Format: TBD-CoRAL

rd-endpoint null [
    rd-linkAttribute "et" [ value "oic.d.sensor" ]
]

2.05 Content
Content-Format: TBD-CoRAL

#base <coap://sensor1.example.com>
rd-item </sensors> [
    content-type 40
    title "Sensor Index"
]
rd-item </sensors/temp> [
    resource-type "temperature-c"
    rd-linkAttribute "if" [ value "sensor" ]
    describedby <http://www.example.com/sensors/t123>
    alternate </t>
]
rd-item </sensors/light> [
    resource-type "light-lux"
    rd-linkAttribute "if" [ value "sensor" ]
]
#base <coap://sensor2.example.com>
rd-item </sensors> [
    content-type 40
    title "Sensor Index"
]
rd-item </sensors/temp> [
    resource-type "temperature-c"
    rd-linkAttribute "if" [ value "sensor" ]
    describedby <http://www.example.com/sensors/t123>
    alternate </t>
]
rd-item </sensors/light> [
    resource-type "light-lux"
    rd-linkAttribute "if" [ value "sensor" ]
]
```

7. IANA Considerations

There are none, this is a thought experiment.

8. Security Considerations

There are some, this is a thought experiment.

9. Normative References

[CoRAL] Hartke, K., "The Constrained RESTful Application Language (CoRAL)", Work in Progress, Internet-Draft, draft-ietf-core-coral-01, 4 November 2019, <https://tools.ietf.org/html/draft-ietf-core-coral-01>.

[I-D.ietf-core-resource-directory] Shelby, Z., Koster, M., Bormann, C., Stok, P., and C. Amsuess, "CoRE Resource Directory", Work in Progress, Internet-Draft, draft-ietf-core-resource-directory-23, 8 July 2019, <https://tools.ietf.org/html/draft-ietf-core-resource-directory-23>.

[I-D.ietf-ace-oauth-authz] Seitz, L., Selander, G., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "Authentication and Authorization for Constrained Environments (ACE) using the OAuth 2.0 Framework (ACE-OAuth)", Work in Progress, Internet-Draft, draft-ietf-ace-oauth-authz-29, 14 December 2019, <https://tools.ietf.org/html/draft-ietf-ace-oauth-authz-29>.

[CMS] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/info/rfc5652>.

[CBOR] 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>.

[COSE] Schaad, J., "CBOR Object Signing and Encryption (COSE)", RFC 8152, DOI 10.17487/RFC8152, July 2017, <https://www.rfc-editor.org/info/rfc8152>.

[RFC7193] Turner, S., Housley, R., and J. Schaad, "The application/cms Media Type", RFC 7193, DOI 10.17487/RFC7193, April 2014, <https://www.rfc-editor.org/info/rfc7193>.

[I-D.ietf-cbor-sequence] Bormann, C., "Concise Binary Object Representation (CBOR) Sequences", Work in Progress, Internet-Draft, draft-ietf-cbor-sequence-02, 25 September 2019, <https://tools.ietf.org/html/draft-ietf-cbor-sequence-02>.

Appendix A. Missing CoRAL things

The start for FETCH is on github for CoRAL. Nothing has been done for PATCH. This appendix is merely a place for me to start thinking about things.

A.1. Rules for doing a FETCH

1. Items of the same name are processed as an 'OR'.

2. Items of different names are processed as 'AND'.

3. A value of 'null' matches all values. Should really be something along the lines of 'undefined' because 'null' may be a real value.

4. Text strings ending in '*' for the search should do wild card matching.

5. Look into adding additional items to allow for doing range, relative value or set processing.

A.2. Rules for doing a PATCH

Need to look at this in detail, because it may be very complicated. I am not sure that the same CoRAL document format can be used. One of the issues is how to match the nth version of something. JSON Patch is probably a better model than SEML patch.

Appendix B. Authorization Vocabulary

Unless otherwise noted, all of the vocabulary defined in this document are prefixed with "http://jimsch.example.org/rd#". For convience, all item defined in this vocabulary is tagged with **strong**.

B.1. Containers

authority The **authority** container is used to hold information about how authentication is going to be done. The container MUST include a **authority-type**. The rest of the content of the container is dependent on the value of the authority type.

B.2. Leafs

authority-type Is a string which identifies what type of authority is being used. Currently defined values are in Table 2.

ACE	The ACE profile [I-D.ietf-ace-oauth-authz]
X.509	X.509 Certificates containing specific information are used for authentication.

Table 2

B.3. ACE Authority Type

Leafs

ace-Profile

What ace profiles are supported by the endpoint. The values of this come from the IANA registry created in [I-D.ietf-ace-oauth-authz].

ace-Audience Audience to ask for a token for

ace-Scope-format Format of the scope parameter

B.4. X.509 Authority Type

Leafs

TrustAnchorCertificate Contains the binary certificate that acts as the trust anchor. This leaf is option as the trust anchor is normally commonly known among all entities in the system.

TrustAnchorFingerprint-SHA256 Contains the SHA-256 fingerprint of the certificate that acts as the trust anchor. This leaf is option as the trust anchor is normally commonly known among all entities in the system.

Author's Address

Jim Schaad
August Cellars

Email: ietf@augustcellars.com