anima Working Group   M. Richardson
Internet-Draft   Sandelman Software Works
Intended status: Standards Track   P. van der Stok
Expires: March 1, 2019   vanderstok consultancy
                                                       P. Kampanakis
                                                        Cisco Systems
                                                     August 28, 2018

          Constrained Voucher Artifacts for Bootstrapping Protocols
                   draft-ietf-anima-constrained-voucher-01

Abstract

   This document defines a strategy to securely assign a pledge to an
   owner, using an artifact signed, directly or indirectly, by the
   pledge's manufacturer. This artifact is known as a "voucher".

   This document builds upon the work in [RFC8366], encoding the
   resulting artifact in CBOR. Use with two signature technologies are
   described.

   Additionally, this document explains how constrained vouchers may be
   transported in the [I-D.ietf-ace-coap-est] protocol.

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 March 1, 2019.

Copyright Notice

Table of Contents

1. Introduction

   Enrollment of new nodes into constrained networks with constrained
   nodes present unique challenges.

   There are bandwidth and code space issues to contend. A solution
   such as [I-D.ietf-anima-bootstrapping-keyinfra] may be too large in
   terms of code space or bandwidth required.

   This document defines a constrained version of [RFC8366]. Rather
   than serializing the YANG definition in JSON, it is serialized into
   CBOR ([RFC7049]).

   This document follows a similar, but not identical structure as
   [RFC8366]. Some sections are left out entirely. Additional sections
   have been added concerning:

   1. Addition of voucher-request specification as defined in
       [I-D.ietf-anima-bootstrapping-keyinfra],

   2. Addition to [I-D.ietf-ace-coap-est] of voucher transport requests
       over coap.

   The CBOR definitions for this constrained voucher format are defined
   using the mechanism describe in [I-D.ietf-core-yang-cbor] using the
   SID mechanism explained in [I-D.ietf-core-sid]. As the tooling to
   convert YANG documents into an list of SID keys is still in its
   infancy, the table of SID values presented here should be considered
   normative rather than the output of the pyang tool.

   Two methods of signing the resulting CBOR object are described in
   this document:

   1. One is CMS [RFC5652].

   2. The other is COSE [RFC8152] signatures.

2. Terminology

The following terms are defined in [RFC8366], and are used
identically as in that document: artifact, imprint, domain, Join
Registrar/Coordinator (JRC), Manufacturer Authorized Signing
Authority (MASA), pledge, Trust of First Use (TOFU), and Voucher.

3. Requirements Language

In this document, the key words "MUST", "MUST NOT", "REQUIRED",
"SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY",
and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119
[RFC2119] and indicate requirement levels for compliant STuPiD
implementations.

4. Survey of Voucher Types

[RFC8366] provides for vouchers that assert proximity, that
authenticate the registrar and that include different amounts of
anti-replay protection.

This document does not make any extensions to the types of vouchers.

Time based vouchers are included in this definition, but given that
constrained devices are extremely unlikely to know the correct time,
their use is very unlikely. Most users of these constrained vouchers
will be online and will use live nonces to provide anti-replay
protection.

[RFC8366] defined only the voucher artifact, and not the Voucher
Request artifact, which was defined in
[I-D.ietf-anima-bootstrapping-keyinfra].

This document defines both a constrained voucher and a constrained
voucher-request. They are presented in the order voucher-request,
followed by voucher response as this is the time order that they
occur.

5. Discovery and URI

This section describes the BRSKI extensions to EST-coaps
[I-D.ietf-ace-coap-est] to transport the voucher between registrar,
proxy and pledge over CoAP. The extensions are targeted to low-
resource networks with small packets. Saving header space is
important and the EST-coaps URI is shorter than the EST URI.

The presence and location of (path to) the management data are
discovered by sending a GET request to "/.well-known/core" including

a resource type (RT) parameter with the value "ace.est" [RFC6690].
Upon success, the return payload will contain the root resource of
the EST resources. It is up to the implementation to choose its root
resource; throughout this document the example root resource /est is
used. The example below shows the discovery of the presence and
location of voucher resources.

```
REQ: GET /.well-known/core?rt=ace.est

RES: 2.05 Content
</est>; rt="ace.est"
```

The EST-coaps server URIs differ from the EST URI by replacing the
scheme https by coaps and by specifying shorter resource path names:

```
coaps://www.example.com/est/short-name
```

Figure 5 in section 3.2.2 of [RFC7030] enumerates the operations and
corresponding paths which are supported by EST. Table 1 provides the
mapping from the BRSKI extension URI path to the EST-coaps URI path.

```
+------------------+-----------+
| BRSKI            | EST-coaps |
+------------------+-----------+
| /requestvoucher  | /rv       |
|                  |           |
| /voucher-status  | /vs       |
|                  |           |
| /enrollstatus    | /es       |
|                  |           |
| /requestauditlog | /ra       |
+------------------+-----------+
```

Table 1: BRSKI path to EST-coaps path

/requestvoucher and /enrollstatus are needed between pledge and
Registrar.

When discovering the root path for the EST resources, the server MAY
return the full resource paths and the used content types. This is
useful when multiple content types are specified for EST-coaps
server. For example, the following more complete response is
possible.

```
   REQ: GET /.well-known/core?rt=ace.est*

   RES: 2.05 Content
   </est>; rt="ace.est"
   </est/rv>; rt="ace.est/rv";ct=50 60 TBD2 TBD3 16
   </est/vs>; rt="ace.est/vs";ct=50 60
   </est/es>; rt="ace.est/es";ct=50 60
   </est/ra>; rt="ace.est/ra";ct=TBD2 TBD3 16
```

The first line MUST be returned in response to the GET, The following
four lines MAY be returned to show the supported Content-Formats.
The return of the content-types allows the client to choose the most
appropriate one from multiple content types.

ct=16 stands for the Content-Format "application/cose", and ct=TBD2
stands for Content-Format "application/voucher-cms+cbor, and ct=TBD3
stands for Content-Format "application/voucher-cose+cbor".

Content-Formats TBD2 and TBD3 are defined in this document. The
return of the content-formts allows the client to choose the most
appropriate one from multiple content formats.

The Content-Format ("application/json") 50 MAY be supported.
Content-Formats ("application/cbor") 60, TBD2, TBD3, and 16 MUST be
supported.

6. Artifacts

This section describes the abstract (tree) definition as explained in
[I-D.ietf-netmod-yang-tree-diagrams] first. This provides a high-
level view of the contents of each artifact.

Then the assigned SID values are presented. These have been assigned
using the rules in [I-D.ietf-core-yang-cbor], with an allocation that
was made via the http://comi.space service.

((EDNOTE: it is unclear if there is further IANA work))

6.1. Voucher Request artifact

6.1.1. Tree Diagram

The following diagram is largely a duplicate of the contents of
[RFC8366], with the addition of proximity-registrar-subject-public-
key-info, proximity-registrar-cert, and prior-signed-voucher-request.

prior-signed-voucher-request is only used between the Registrar and
the MASA. proximity-registrar-subject-public-key-info replaces
proximity-registrar-cert for the extremely constrained cases.

```
module: ietf-constrained-voucher-request

  grouping voucher-request-constrained-grouping
    +---- voucher
       +---- created-on
       |       yang:date-and-time
       +---- expires-on?
       |       yang:date-and-time
       +---- assertion
       |       enumeration
       +---- serial-number                              string
       +---- idevid-issuer?                             binary
       +---- pinned-domain-cert                         binary
       +---- domain-cert-revocation-checks?             boolean
       +---- nonce?                                     binary
       +---- last-renewal-date?
       |       yang:date-and-time
       +---- proximity-registrar-subject-public-key-info?   binary
       +---- proximity-registrar-cert?                  binary
       +---- prior-signed-voucher-request?              binary
```

6.1.2. SID values

[[EDNOTE - to be fixed]]

6.1.3. YANG Module

In the constrained-voucher-request YANG module, the voucher is "used"
and not "augmented" such that one continuous set of SID values is
generated for the constrained-voucher-request module name, all
voucher attributes, and the constrained-voucher-request attribute.

```
<CODE BEGINS> file "ietf-constrained-voucher-request@2018-09-01.yang"
module ietf-constrained-voucher-request {
  yang-version 1.1;

  namespace
    "urn:ietf:params:xml:ns:yang:ietf-constrained-voucher-request";
  prefix "constrained";

  import ietf-restconf {
    prefix rc;
    description
```

```
    "This import statement is only present to access
     the yang-data extension defined in RFC 8040.";
  reference "RFC 8040: RESTCONF Protocol";
}

import ietf-voucher {
  prefix "v";
}

organization
 "IETF ANIMA Working Group";

contact
 "WG Web:   <http://tools.ietf.org/wg/anima/>
  WG List:  <mailto:anima@ietf.org>
  Author:   Michael Richardson
            <mailto:mcr+ietf@sandelman.ca>
  Author:   Peter van der Stok
            <mailto: consultancy@vanderstok.org>
  Author:   Panos Kampanakis
            <mailto: pkampana@cisco.com>";
description
 "This module defines the format for a voucher request,
  which is produced by a pledge to request a voucher.
  The voucher-request is sent to the potential owner's
  Registrar, which in turn sends the voucher request to
  the manufacturer or delegate (MASA).

  A voucher is then returned to the pledge, binding the
  pledge to the owner.  This is a constrained version of the
  voucher-request present in
  draft-ietf-anima-bootstrap-keyinfra.txt.

  This version provides a very restricted subset appropriate
  for very constrained devices.
  In particular, it assumes that nonce-ful operation is
  always required, that expiration dates are rather weak, as no
  clocks can be assumed, and that the Registrar is identified
  by a pinned Raw Public Key.

  The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL',
  'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY',
  and 'OPTIONAL' in the module text are to be interpreted as
  described in RFC 2119.";

revision "2018-09-01" {
  description
    "Initial version";
```

```
      reference
       "RFC XXXX: Voucher Profile for Constrained Devices";
    }

    rc:yang-data voucher-request-constrained-artifact {
      // YANG data template for a voucher.
      uses voucher-request-constrained-grouping;
    }

    // Grouping defined for future usage
    grouping voucher-request-constrained-grouping {
      description
        "Grouping to allow reuse/extensions in future work.";

      uses v:voucher-artifact-grouping {
        augment "voucher" {
          description "Base the constrained voucher-request upon the
            regular one";

          leaf proximity-registrar-subject-public-key-info {
            type binary;
            description
              "The proximity-registrar-subject-public-key-info replaces
               the proximit-registrar-cert in constrained uses of
               the voucher-request.
               The proximity-registrar-subject-public-key-info is the
               Raw Public Key of the Registrar. This field is encoded
               as specified in RFC7250, section 3.
               The ECDSA algorithm MUST be supported.
               The EdDSA algorithm as specified in
               draft-ietf-tls-rfc4492bis-17 SHOULD be supported.
               Support for the DSA algorithm is not recommended.
               Support for the RSA algorithm is a MAY.";
          }

          leaf proximity-registrar-cert {
            type binary;
            description
              "An X.509 v3 certificate structure as specified by
               RFC 5280,
               Section 4 encoded using the ASN.1 distinguished encoding
               rules (DER), as specified in ITU-T X.690.

               The first certificate in the Registrar TLS server
               certificate_list sequence  (see [RFC5246]) presented by
               the Registrar to the Pledge. This MUST be populated in a
               Pledge's voucher request if the proximity assertion is
               populated.";
```

```
          }

          leaf prior-signed-voucher-request {
            type binary;
            description
              "If it is necessary to change a voucher, or re-sign and
               forward a voucher that was previously provided along a
               protocol path, then the previously signed voucher
               SHOULD be included in this field.

               For example, a pledge might sign a proximity voucher,
               which an intermediate registrar then re-signs to
               make its own proximity assertion.  This is a simple
               mechanism for a chain of trusted parties to change a
               voucher, while maintaining the prior signature
               information.

               The pledge MUST ignore all prior voucher information
               when accepting a voucher for imprinting. Other
               parties MAY examine the prior signed voucher
               information for the purposes of policy decisions.
               For example this information could be useful to a
               MASA to determine that both pledge and registrar
               agree on proximity assertions. The MASA SHOULD
               remove all prior-signed-voucher-request information when
               signing a voucher for imprinting so as to minimize the
               final voucher size.";
          }

        }
      }
    }
  }
  <CODE ENDS>
```

6.1.4. Example voucher request artifacts

TBD

6.2. Voucher artifact

The voucher's primary purpose is to securely assign a pledge to an
owner. The voucher informs the pledge which entity it should
consider to be its owner.

This document defines a voucher that is a CBOR encoded instance of
the YANG module defined in Section 5.3 that has been signed with CMS
or with COSE.

6.2.1. Tree Diagram

   The following diagram is largely a duplicate of the contents of
   [RFC8366], with only the addition of pinned-domain-subject-public-
   key-info.

   module: ietf-constrained-voucher

```
     grouping voucher-constrained-grouping
       +---- voucher
          +---- created-on
          |       yang:date-and-time
          +---- expires-on?
          |       yang:date-and-time
          +---- assertion                              enumeration
          +---- serial-number                          string
          +---- idevid-issuer?                         binary
          +---- pinned-domain-cert                     binary
          +---- domain-cert-revocation-checks?         boolean
          +---- nonce?                                 binary
          +---- last-renewal-date?
          |       yang:date-and-time
          +---- pinned-domain-subject-public-key-info?   binary
```

6.2.2. SID values

6.2.3. YANG Module

   In the constraine-voucher YANG module, the voucher is "used" and not
   "augmented" such that one continuous set of SID values is generated
   for the constrained-voucher module name, all voucher attributes, and
   the constrained-voucher attribute.

```
<CODE BEGINS> file "ietf-constrained-voucher@2018-09-01.yang"
module ietf-constrained-voucher {
  yang-version 1.1;

  namespace
    "urn:ietf:params:xml:ns:yang:ietf-constrained-voucher";
  prefix "constrained";

  import ietf-restconf {
    prefix rc;
    description
      "This import statement is only present to access
       the yang-data extension defined in RFC 8040.";
    reference "RFC 8040: RESTCONF Protocol";
```

```
  }

  import ietf-voucher {
    prefix "v";
  }

  organization
   "IETF ANIMA Working Group";

  contact
   "WG Web:    <http://tools.ietf.org/wg/anima/>
    WG List:  <mailto:anima@ietf.org>
    Author:    Michael Richardson
               <mailto:mcr+ietf@sandelman.ca>
    Author:    Peter van der Stok
               <mailto: consultancy@vanderstok.org>
    Author:    Panos Kampanakis
               <mailto: pkampana@cisco.com>";
description
  "This module defines the format for a voucher, which is produced
   by a pledge's manufacturer or delegate (MASA) to securely assign
   one or more pledges to an 'owner', so that the pledges may
   establis a secure connection to the owner's network
   infrastructure.

   This version provides a very restricted subset appropriate
   for very constrained devices.
   In particular, it assumes that nonce-ful operation is
   always required, that expiration dates are rather weak, as no
   clocks can be assumed, and that the Registrar is identified
   by a pinned Raw Public Key.

   The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL',
   'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY',
   and 'OPTIONAL' in the module text are to be interpreted as
   described in RFC 2119.";

  revision "2018-09-01" {
    description
     "Initial version";
    reference
     "RFC XXXX: Voucher Profile for Constrained Devices";
  }

  rc:yang-data voucher-constrained-artifact {
    // YANG data template for a voucher.
    uses voucher-constrained-grouping;
  }
```

```
  // Grouping defined for future usage
  grouping voucher-constrained-grouping {
    description
      "Grouping to allow reuse/extensions in future work.";

    uses v:voucher-artifact-grouping {
      augment "voucher" {
        description "Base the constrained voucher upon the regular one";
        leaf pinned-domain-subject-public-key-info {
          type binary;
          description
            "The pinned-domain-subject replaces the
             pinned-domain-certificate in constrained uses of
             the voucher.  The pinned-domain-subject-public-key-info
             is the Raw Public Key of the Registrar.
             This field is encoded as specified in RFC7250, section 3.
             The ECDSA algorithm MUST be supported.
             The EdDSA algorithm as specified in
             draft-ietf-tls-rfc4492bis-17 SHOULD be supported.
             Support for the DSA algorithm is not recommended.
             Support for the RSA algorithm is a MAY.";
        }
      }
    }
  }
}
<CODE ENDS>
```

6.2.4. **Example voucher artifacts**

   Below a the CBOR serialization of the the constrained-voucher and
   constrained-voucher-request are shown in diagnostic CBOR notation.

6.2.4.1. **CBOR serialization of constrained-voucher**

```
   {
     1001051: {
       +2 : "2016-10-07T19:31:42Z", / SID = 1001053, created-on /
       +4 : "2016-10-21T19:31:42Z", / SID = 1001055, expires-on /
       +1 : "verified",             / SID = 1001052, assertion /
       +11: "JADA123456789",        / SID = 1001062, serial-number /
       +5 : h'01020D0F',            / SID = 1001056, idevid-issuer /
       +8 : h'01020D0F',            / SID = 1001059, pinned-domain-cert/
       +3 : true,                   / SID = 1001054, domain-cert
                                                     -revocation-checks/
       +6 : "2017-10-07T19:31:42Z", / SID = 1001057, last-renewal-date /
       +9 : h'01020D0F'             / SID = 1001060, pinned-domain
                                                -subject-public-key-info /
     }
   }
```

6.2.4.2. CBOR serialization of constrained-voucher-request

```
   {
     1001101: {
       +2 : "2016-10-07T19:31:42Z", / SID = 1001103, created-on /
       +4 : "2016-10-21T19:31:42Z", / SID = 1001105, expires-on /
       +1 : "verified",             / SID = 1001102, assertion /
       +11: "JADA123456789",        / SID = 1001112, serial-number /
       +5 : h'01020D0F',            / SID = 1001106, idevid-issuer /
       +8 : h'01020D0F',            / SID = 1001109, pinned-domain-cert/
       +3 : true,                   / SID = 1001104, domain-cert
                                                      -revocation-checks /
       +6 : "2017-10-07T19:31:42Z", / SID = 1001107, last-renewal-date /
       +10: h'01020D0F'             / SID = 1001111, proximity
                                        -registrar-subject-public-key-info /
     }
   }
```

6.3. CMS format voucher and voucher-request artifacts

   The IETF evolution of PKCS#7 is CMS [RFC5652]. The CMS signed
   voucher is much like the equivalent voucher defined in [RFC8366].

   A different eContentType of TBD1 is used to indicate that the
   contents are in a different format than in [RFC8366].

   The ContentInfo structure contains a payload consisting of the CBOR
   encoded voucher. The [I-D.ietf-core-yang-cbor] use of delta encoding
   creates a canonical ordering for the keys on the wire. This
   canonical ordering is not important as there is no expectation that
   the content will be reproduced during the validation process.

Normally the recipient is the pledge and the signer is the MASA.

[I-D.ietf-anima-bootstrapping-keyinfra] supports both signed and
unsigned voucher requests from the pledge to the JRC. In this
specification, voucher-request artifact is not signed from the pledge
to the registrar. From the JRC to the MASA, the voucher-request
artifact MUST be signed by the domain owner key which is requesting
ownership.

The considerations of [RFC5652] section 5.1, concerning validating
CMS objects which are really PKCS7 objects (cmsVersion=1) applies.

The CMS structure SHOULD also contain all the certificates leading up
to and including the signer's trust anchor certificate known to the
recipient. The inclusion of the trust anchor is unusual in many
applications, but without it third parties can not accurately audit
the transaction.

The CMS structure MAY also contain revocation objects for any
intermediate certificate authorities (CAs) between the voucher-issuer
and the trust anchor known to the recipient. However, the use of
CRLs and other validity mechanisms is discouraged, as the pledge is
unlikely to be able to perform online checks, and is unlikely to have
a trusted clock source. As described below, the use of short-lived
vouchers and/or pledge provided nonce provides a freshness guarantee.

6.3.1. COSE signing

The COSE-Sign1 structure discussed in section 4.2 of [RFC8152]. The
CBOR object that carries the body, the signature, and the information
about the body and signature is called the COSE_Sign1 structure. It
is used when only one signature is used on the body. The signature
algorithm is ECSDA with three curves P-256, P-384, and P-512.

Support for EdDSA is encouraged.

Unlike with the CMS structure, the COSE-Sign1 structure does not
provide a standard way for the signing keys to be included in the
structure. This will not, in general, be a problem for the Pledge,
as it the key needed to verify the signature MUST be included at
manufacturing time.

A problem arises for the Registrar, in order for it to verify the
voucher, it must have access to the MASA's public key. This document
does not specify how to transfer the relevant key.

7. Design Considerations

   The design considerations for the CBOR encoding of vouchers is much
   the same as for [RFC8366].

   One key difference is that the names of the leaves in the YANG does
   not have a material effect on the size of the resulting CBOR, as the
   SID translation process assigns integers to the names.

8. Security Considerations

8.1. Clock Sensitivity

   TBD.

8.2. Protect Voucher PKI in HSM

   TBD.

8.3. Test Domain Certificate Validity when Signing

   TBD.

9. IANA Considerations

9.1. Resource Type Registry

   Additions to the sub-registry "CoAP Resource Type", within the "CoRE
   parameters" registry are specified below. These can be registered
   either in the Expert Review range (0-255) or IETF Review range
   (256-9999).

    ace.rt.rv needs registration with IANA
    ace.rt.vs needs registration with IANA
    ace.rt.es needs registration with IANA
    ace.rt.ra needs registration with IANA

9.2. The IETF XML Registry

   This document registers two URIs in the IETF XML registry [RFC3688].
   Following the format in [RFC3688], the following registration is
   requested:

```
   URI: urn:ietf:params:xml:ns:yang:ietf-constrained-voucher
   Registrant Contact: The ANIMA WG of the IETF.
   XML: N/A, the requested URI is an XML namespace.

   URI: urn:ietf:params:xml:ns:yang:ietf-constrained-voucher-request
   Registrant Contact: The ANIMA WG of the IETF.
   XML: N/A, the requested URI is an XML namespace.
```

9.3. The YANG Module Names Registry

This document registers two YANG modules in the YANG Module Names
registry [RFC6020]. Following the format defined in [RFC6020], the
the following registration is requested:

```
   name:         ietf-constrained-voucher
   namespace:    urn:ietf:params:xml:ns:yang:ietf-constrained-voucher
   prefix:       vch
   reference:    RFC XXXX

   name:         ietf-constrained-voucher-request
   namespace:    urn:ietf:params:xml:ns:yang:ietf-constrained
                                     -voucher-request
   prefix:       vch
   reference:    RFC XXXX
```

9.4. The SMI Security for S/MIME CMS Content Type Registry

This document registers an OID in the "SMI Security for S/MIME CMS
Content Type" registry (1.2.840.113549.1.9.16.1), with the value:

```
   Decimal  Description                            References
   -------  -------------------------------------  ----------
   TBD1     id-ct-animaCBORVoucher                 [ThisRFC]
```

EDNOTE: should a separate value be used for Voucher Requests?

9.5. The SID registry

The SID range 1001100 was allocated by comi.space to the IETF-
CONSTRAINED-VOUCHER yang module.

The SID range 1001150 was allocated by comi.space to the IETF-
CONSTRAINED-VOUCHER-REQUEST yang module.

EDNOTE: it is unclear if there is further IANA work required.

9.6. Media-Type Registry

   This section registers the 'application/voucher-cms+cbor' media type
   and the 'application/voucher-cose+cbor'in the "Media Types" registry.
   These media types are used to indicate that the content is a CBOR
   voucher either signed with a cms structure or a COSE_Sign1 structure
   [RFC8152].

9.6.1. application/voucher-cms+cbor

   Type name: application
   Subtype name: voucher-cms+cbor
   Required parameters: none
   Optional parameters: none
   Encoding considerations: CMS-signed CBOR vouchers are CBOR
     encoded.
   Security considerations: See Security Considerations, Section
   Interoperability considerations: The format is designed to be
     broadly interoperable.
   Published specification: THIS RFC.
   Applications that use this media type: ANIMA, 6tisch, and other
     zero-touch imprinting systems
   Additional information:
     Magic number(s): None
     File extension(s): .vch
     Macintosh file type code(s): none
   Person & email address to contact for further information: IETF
     ANIMA WG
   Intended usage: LIMITED
   Restrictions on usage: NONE
   Author: ANIMA WG
   Change controller: IETF
   Provisional registration? (standards tree only): NO

9.6.2. application/voucher-cose+cbor

```
   Type name:  application
   Subtype name:  voucher-cose+cbor
   Required parameters:  none
   Optional parameters:  cose-type
   Encoding considerations:  COSE_Sign1 CBOR vouchers are COSE objects
                             signed with one signer.
   Security considerations:  See Security Considerations, Section
   Interoperability considerations:  The format is designed to be
     broadly interoperable.
   Published specification:  THIS RFC.
   Applications that use this media type:  ANIMA, 6tisch, and other
     zero-touch imprinting systems
   Additional information:
     Magic number(s):  None
     File extension(s):  .vch
     Macintosh file type code(s):  none
   Person & email address to contact for further information:  IETF
     ANIMA WG
   Intended usage:  LIMITED
   Restrictions on usage:  NONE
   Author:  ANIMA WG
   Change controller:  IETF
   Provisional registration? (standards tree only):  NO
```

9.7. CoAP Content-Format Registry

Additions to the sub-registry "CoAP Content-Formats", within the
"CoRE Parameters" registry are needed for two media types. These can
be registered either in the Expert Review range (0-255) or IETF
Review range (256-9999).

```
   Media type                     mime type    Encoding   ID  References
   ----------------------------   -----------  ---------  ---- ----------
   application/voucher-cms+cbor      - -         CBOR     TBD2  [This RFC]
   application/voucher-cose+cbor  "COSE-Sign1"   CBOR     TBD3  [This RFC]
```

10. Acknowledgements

We are very grateful to Jim Schaad for explaining COSE and CMS
choices.

Michel Veillette did extensive work on pyang to extend it to support
the SID allocation process, and this document was among the first
users.

We are grateful for the suggestions done by Esko Dijk.

11. Changelog

   -01

   application/json is optional, application/cbor is compulsory
   Cms and cose mediatypes are introduced

12. References

12.1. Normative References

   [I-D.ietf-ace-cbor-web-token]
            Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig,
            "CBOR Web Token (CWT)", draft-ietf-ace-cbor-web-token-15
            (work in progress), March 2018.

   [I-D.ietf-ace-coap-est]
            Stok, P., Kampanakis, P., Kumar, S., Richardson, M.,
            Furuhed, M., and S. Raza, "EST over secure CoAP (EST-
            coaps)", draft-ietf-ace-coap-est-05 (work in progress),
            July 2018.

   [I-D.ietf-anima-bootstrapping-keyinfra]
            Pritikin, M., Richardson, M., Behringer, M., Bjarnason,
            S., and K. Watsen, "Bootstrapping Remote Secure Key
            Infrastructures (BRSKI)", draft-ietf-anima-bootstrapping-
            keyinfra-16 (work in progress), June 2018.

   [I-D.ietf-core-object-security]
            Selander, G., Mattsson, J., Palombini, F., and L. Seitz,
            "Object Security for Constrained RESTful Environments
            (OSCORE)", draft-ietf-core-object-security-14 (work in
            progress), July 2018.

   [I-D.ietf-core-sid]
            Veillette, M. and A. Pelov, "YANG Schema Item iDentifier
            (SID)", draft-ietf-core-sid-04 (work in progress), June
            2018.

   [I-D.ietf-core-yang-cbor]
            Veillette, M., Pelov, A., Somaraju, A., Turner, R., and A.
            Minaburo, "CBOR Encoding of Data Modeled with YANG",
            draft-ietf-core-yang-cbor-06 (work in progress), February
            2018.

   [ieee802-1AR]
            IEEE Standard, ., "IEEE 802.1AR Secure Device Identifier",
            2009, <http://standards.ieee.org/findstds/
            standard/802.1AR-2009.html>.

   [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/info/rfc2119>.

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

   [RFC7049] 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>.

   [RFC7250] Wouters, P., Ed., Tschofenig, H., Ed., Gilmore, J.,
            Weiler, S., and T. Kivinen, "Using Raw Public Keys in
            Transport Layer Security (TLS) and Datagram Transport
            Layer Security (DTLS)", RFC 7250, DOI 10.17487/RFC7250,
            June 2014, <https://www.rfc-editor.org/info/rfc7250>.

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

   [RFC8366] Watsen, K., Richardson, M., Pritikin, M., and T. Eckert,
            "A Voucher Artifact for Bootstrapping Protocols",
            RFC 8366, DOI 10.17487/RFC8366, May 2018,
            <https://www.rfc-editor.org/info/rfc8366>.

12.2. Informative References

   [duckling]
            Stajano, F. and R. Anderson, "The resurrecting duckling:
            security issues for ad-hoc wireless networks", 1999,
            <https://www.cl.cam.ac.uk/~fms27/
            papers/1999-StajanoAnd-duckling.pdf>.

   [I-D.ietf-netmod-yang-tree-diagrams]
            Bjorklund, M. and L. Berger, "YANG Tree Diagrams", draft-
            ietf-netmod-yang-tree-diagrams-06 (work in progress),
            February 2018.

   [pledge] Dictionary.com, ., "Dictionary.com Unabridged", 2015,
            <http://dictionary.reference.com/browse/pledge>.

   [RFC6690]   Shelby, Z., "Constrained RESTful Environments (CoRE) Link
               Format", RFC 6690, DOI 10.17487/RFC6690, August 2012,
               <https://www.rfc-editor.org/info/rfc6690>.

   [RFC7030]   Pritikin, M., Ed., Yee, P., Ed., and D. Harkins, Ed.,
               "Enrollment over Secure Transport", RFC 7030,
               DOI 10.17487/RFC7030, October 2013,
               <https://www.rfc-editor.org/info/rfc7030>.

Appendix A. EST messages to EST-coaps

   This section extends the examples from Appendix A of
   [I-D.ietf-ace-coap-est]. The CoAP headers are only worked out for
   the enrollstatus example.

A.1. enrollstatus

   A coaps enrollstatus message can be :

       GET coaps://[192.0.2.1:8085]/est/es

   The corresponding coap header fields are shown below.

```
  Ver = 1
  T = 0 (CON)
  Code = 0x01 (0.01 is GET)
  Options
   Option1 (Uri-Host)
     Option Delta = 0x3  (option nr = 3)
     Option Length = 0x9
     Option Value = 192.0.2.1
   Option2 (Uri-Port)
     Option Delta = 0x4  (option nr = 4+3=7)
     Option Length = 0x4
     Option Value = 8085
   Option3 (Uri-Path)
     Option Delta = 0x4   (option nr = 7+4= 11)
     Option Length = 0x7
     Option Value = /est/es
  Payload = [Empty]
```

   A 2.05 Content response with an unsigned JSON voucher (ct=50) will
   then be:

       2.05 Content (Content-Format: application/json)
         {payload}

   With CoAP fields and payload:

```
  Ver=1
  T=2 (ACK)
  Code = 0x45 (2.05 Content)
  Options
    Option1 (Content-Format)
    Option Delta = 0xC  (option nr 12)
    Option Length = 0x2
    Option Value = 0x32 (application/json)

    Payload =
   [EDNOTE: put here voucher payload ]
```

A.2. voucher_status

A coaps voucher_status message can be :

```
   GET coaps://[2001:db8::2:1]:61616]/est/vs
```

A 2.05 Content response with a non signed CBOR voucher (ct=60) will
then be:

```
   2.05 Content (Content-Format: application/cbor)
   Payload =
   [EDNOTE: put here voucher payload ]
```

A.3. requestvoucher

A coaps requestvoucher message can be :

```
    POST coaps://[2001:db8::2:1]:61616]/est/rv
```

A 2.04 Changed response returning CBOR voucher signed with a cms
structure(ct=TBD2) will then be:

```
   2.05 Changed (Content-Format: application/voucher-cms+cbor)
   Payload =
   [EDNOTE: put here encrypted voucher payload ]
```

A.4. requestauditing

A coaps requestauditing message can be :

```
   GET coaps://[2001:db8::2:1]:61616]/est/ra
```

A 2.05 Content response returning a COSE_Sign1 object (ct=TBD3) will
then be:

         2.05 Content (Content-Format: application/voucher-cose+cbor)
         Payload =
         [EDNOTE: put here COSE_Sign1 voucher payload]

Authors' Addresses

   Michael Richardson
   Sandelman Software Works

   Email: mcr+ietf@sandelman.ca

   Peter van der Stok
   vanderstok consultancy

   Email: consultancy@vanderstok.org

   Panos Kampanakis
   Cisco Systems

   Email: pkampana@cisco.com