ACE Working Group
Internet-Draft
Updates: 9202 (if approved)
Intended status: Standards Track
Expires: 8 January 2026

M. Tiloca
RISE AB
J. Preu Mattsson
Ericsson AB

7 July 2025

Additional Formats of Authentication Credentials for the Datagram Transport Layer Security (DTLS) Profile for Authentication and Authorization for Constrained Environments (ACE) draft-ietf-ace-authcred-dtls-profile-02

Abstract

This document updates the Datagram Transport Layer Security (DTLS) profile for Authentication and Authorization for Constrained Environments (ACE). In particular, it specifies the use of additional formats of authentication credentials for establishing a DTLS session, when peer authentication is based on asymmetric cryptography. Therefore, this document updates RFC 9202. What is defined in this document is seamlessly applicable also if the profile uses Transport Layer Security (TLS) instead, as defined in RFC 9430.

Discussion Venues

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

Discussion of this document takes place on the Authentication and Authorization for Constrained Environments Working Group mailing list (ace@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ace/.

Source for this draft and an issue tracker can be found at https://github.com/ace-wg/ace-authcred-dtls-profile.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Authentication and Authorization for Constrained Environments (ACE) framework [RFC9200] defines an architecture to enforce access control for constrained devices. A client (C) requests an evidence of granted permissions from an authorization server (AS) in the form of an access token, then uploads the access token to the target resource server (RS), and finally accesses protected resources at the RS according to what is specified in the access token.

The framework has as main building blocks the OAuth 2.0 framework [RFC6749], the Constrained Application Protocol (CoAP) [RFC7252] for message transfer, Concise Binary Object Representation (CBOR) [RFC8949] for compact encoding, and CBOR Object Signing and Encryption (COSE) [RFC9052][RFC9053] for self-contained protection of access tokens.

Separate profile documents define in detail how the participants in the ACE architecture communicate, especially as to the security protocols that they use. In particular, the ACE profile defined in [RFC9202] specifies how Datagram Transport Layer Security (DTLS) [RFC6347][RFC9147] is used to protect communications with transport-layer security in the ACE architecture. The profile has been extended in [RFC9430], in order to allow the alternative use of Transport Layer Security (TLS) [RFC8446] when CoAP is transported over TCP or WebSockets [RFC8323].

The DTLS profile defined in [RFC9202] allows C and the RS to establish a DTLS session with peer authentication based on symmetric or asymmetric cryptography. For the latter case, the profile defines a Raw Public Key (RPK) mode (see Section 3.2 of [RFC9202]), where authentication relies on the public keys of the two peers as raw public keys [RFC7250].

That is, C specifies its public key to the AS when requesting an access token and the AS provides such public key to the target RS as included in the issued access token. Upon issuing the access token, the AS also provides C with the public key of the RS. Then, C and the RS use their asymmetric keys when performing the DTLS handshake, as defined in [RFC7250].

Per [RFC9202], the DTLS profile admits only a COSE_Key object [RFC9052] as the format of authentication credentials to use for specifying the public keys of C and the RS as raw public keys. However, it is desirable to enable additional formats of authentication credentials, as enhanced raw public keys or as public key certificates.

This document enables such additional formats in the DTLS profile, by defining how the public keys of C and the RS can be specified by means of CBOR Web Token (CWT) Claims Sets (CCSs) [RFC8392], X.509 certificates [RFC5280], or C509 certificates [I-D.ietf-cose-cbor-encoded-cert].

This document also enables the DTLS profile to use the CWT Confirmation Method 'ckt' defined in [RFC9679] when using a COSE_Key object for specifying a raw public key, thus allowing to identifying the COSE_Key object by reference alternatively to transporting it by value.

In particular, this document updates [RFC9202] as follows.

* Section 2 of this document extends the RPK mode defined in Section 3.2 of [RFC9202], by enabling:

  - The use of CCSs to wrap the raw public keys of C and the RS, i.e., as a new format of authentication credentials that can be used for specifying the public keys of C and the RS as raw public keys (see Section 2.1).

  - The use of the CWT Confirmation Method 'ckt' to identify a COSE_Key object by reference, when that is the format of authentication credentials used for specifying the public keys of C and the RS as raw public keys (see Section 2.2).

* Section 3 of this document defines a new certificate mode, which enables the use of X.509 or C509 certificates to specify the public keys of C and the RS. In either case, certificates can be transported by value or instead identified by reference.

When using the updated RPK mode, the raw public keys of C and the RS do not have to be of the same format. That is, it is possible to have both public keys as a COSE_Key object or as a CCS, or instead one as a COSE_Key object while the other one as a CCS. When both raw public keys are COSE_Keys, it is possible to have both COSE_Keys transported by value, or both identified by reference, or one transported by value while the other one identified by reference.

When using the certificate mode, the certificates of C and the RS do not have to be of the same format. That is, it is possible to have both as X.509 certificates, or both as C509 certificates, or one as an X.509 certificate while the other one as a C509 certificate. Furthermore, it is possible to have both certificates transported by value, or both identified by reference, or one transported by value while the other one identified by reference.

Also, the RPK mode and the certificate mode can be combined. That is, it is possible that one of the two authentication credentials is a certificate, while the other one is a raw public key.

The effective provisioning of an authentication credential identified by reference builds on the assumption that the recipient is storing the authentication credential by value, or is able to retrieve it from a trusted source by means of the reference obtained. If that assumption does not hold, the authentication credential will have to be provided by value.

The decision about whether providing authentication credentials by value or by reference depending on the specific situation is left to application policies at C and the AS. Furthermore, C and AS could explicitly coordinate with each other about exchanging the authentication credentials of C and the RS as transported by value or instead identified by reference, e.g., by relying on the coordination method defined in [I-D.ietf-ace-workflow-and-params].

When using the formats introduced in this document, authentication credentials are specified by means of the CWT Confirmation Methods "kccs", "x5bag", "x5chain", "x5t", "x5u", "c5b", "c5c", "c5t", and "c5u" that are defined in [I-D.ietf-ace-edhoc-oscore-profile].

What is defined in this document is seamlessly applicable if TLS is used instead, as defined in [RFC9430].

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Readers are expected to be familiar with the terms and concepts described in the ACE framework for Authentication and Authorization [RFC9200][RFC9201] and in its DTLS profile [RFC9202], as well as with terms and concepts related to CBOR Web Tokens (CWTs) [RFC8392] and CWT Confirmation Methods [RFC8747].

The terminology for entities in the considered architecture is defined in OAuth 2.0 [RFC6749]. In particular, this includes client (C), resource server (RS), and authorization server (AS).

Readers are also expected to be familiar with the terms and concepts related to CoAP [RFC7252], CBOR [RFC8949], Concise Data Definition Language (CDDL) [RFC8610], COSE [RFC9052][RFC9053], DTLS [RFC6347][RFC9147], and the use of raw public keys in DTLS [RFC7250].

Note that the term "endpoint" is used here following its OAuth definition [RFC6749], aimed at denoting resources such as /token and /introspect at the AS, and /authz-info at the RS. The CoAP definition, which is "[a]n entity participating in the CoAP protocol" [RFC7252], is not used in this document.

This document also refers to the term "authentication credential", which denotes the information associated with an entity, including that entity's public key and parameters associated with the public key. Examples of authentication credentials are CWT Claims Sets (CCSs) [RFC8392], X.509 certificates [RFC5280], and C509 certificates [I-D.ietf-cose-cbor-encoded-cert].

Examples throughout this document are expressed in CBOR diagnostic notation as defined in Section 8 of [RFC8949] and Appendix G of [RFC8610]. Diagnostic notation comments are often used to provide a textual representation of the parameters' keys and values.

In the CBOR diagnostic notation used in this document, constructs of the form e'SOME_NAME' are replaced by the value assigned to SOME_NAME in the CDDL model shown in Figure 19 of Appendix B. For example, {e'x5chain' : h'3081...cb02'} stands for {6 : h'3081...cb02'}.

Note to RFC Editor: Please delete the paragraph immediately preceding this note. Also, in the CBOR diagnostic notation used in this document, please replace the constructs of the form e'SOME_NAME' with the value assigned to SOME_NAME in the CDDL model shown in Figure 19 of Appendix B. Finally, please delete this note.

2. Updates to the RPK Mode

This section updates the RPK mode defined in Section 3.2 of [RFC9202], as detailed in the following Section 2.1 and Section 2.2.

2.1. Raw Public Keys as CCSs

This section defines how the raw public key of C and the RS can be provided as wrapped by a CCS [RFC8392], instead of as a COSE_Key object [RFC9052]. Note that only the differences from [RFC9202] are compiled below.

If the raw public key of C is wrapped by a CCS, then the following applies.

* The payload of the Access Token Request (see Section 5.8.1 of [RFC9200]) is as defined in Section 3.2.1 of [RFC9202], with the difference that the "req_cnf" parameter [RFC9201] MUST specify a "kccs" structure, with value a CCS specifying the public key of C that has to be bound to the access token.

  In particular, the CCS MUST include the "cnf" claim specifying the public key of C as a COSE_Key object, SHOULD include the "sub" claim specifying the subject name of C associated with the public key of C, and MAY include additional claims.

* The content of the access token that the AS provides to C in the Access Token Response (see Section 5.8.2 of [RFC9200]) is as defined in Section 3.2.1 of [RFC9202], with the difference that the "cnf" claim of the access token MUST specify a "kccs" structure, with value a CCS specifying the public key of C that is bound to the access token.

  In particular, the CCS MUST include the "cnf" claim specifying the public key of C as a COSE_Key object, SHOULD include the "sub" claim specifying the subject name of C associated with the public key of C, and MAY include additional claims.

If the raw public key of the RS is wrapped by a CCS, then the following applies.

* The payload of the Access Token Response is as defined in Section 3.2.1 of [RFC9202], with the difference that the "rs_cnf" parameter [RFC9201] MUST specify a "kccs" structure, with value a CCS specifying the public key of the RS.

  In particular, the CCS MUST include the "cnf" claim specifying the public key of the RS as a COSE_Key object, SHOULD include the "sub" claim specifying the subject name of the RS associated with the public key of the RS, and MAY include additional claims.

For the "req_cnf" parameter of the Access Token Request, the "rs_cnf" parameter of the Access Token Response, and the "cnf" claim of the access token, the Confirmation Method "kccs" structure and its identifier are defined in [I-D.ietf-ace-edhoc-oscore-profile].

It is not required that both public keys are wrapped by a CCS. That is, one of the two authentication credentials can be a CCS, while the other one can be a COSE_Key object transported by value as per Section 3.2 of [RFC9202] or identified by reference as per Section 2.2 of this document.

2.1.1. Examples

Figure 1 shows an example of Access Token Request from C to the AS.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor1",
  / req_cnf /     4 : {
    e'kccs' : {
      / sub / 2 : "42-50-31-FF-EF-37-32-39",
      / cnf / 8 : {
        / COSE_Key / 1 : {
          / kty /    1 : 2 / EC2 /,
          / crv /   -1 : 1 / P-256 /,
          / x /     -2 : h'd7cc072de2205bdc1537a543d53c60a6
                            acb62eccd890c7fa27c9e354089bbe13',
          / y /     -3 : h'f95e1d4b851a2cc80fff87d8e23f22af
                            b725d535e515d020731e79a3b4e47120'
        }
      }
    }
  }
}
```

Figure 1: Access Token Request Example for RPK Mode, with the Public Key of C Wrapped by a CCS Conveyed within "req_cnf"

Figure 2 shows an example of Access Token Response from the AS to C.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...643b',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's RPK in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'kccs' : {
      / sub / 2 : "AA-BB-CC-00-01-02-03-04",
      / cnf / 8 : {
        / COSE_Key / 1 : {
          / kty /  1 : 2 / EC2 /,
          / crv / -1 : 1 / P-256 /,
          / x /   -2 : h'bbc34960526ea4d32e940cad2a234148
                         ddc21791a12afbcbac93622046dd44f0',
          / y /   -3 : h'4519e257236b2a0ce2023f0931f1f386
                         ca7afda64fcde0108c224c51eabf6072'
        }
      }
    }
  }
}
```

Figure 2: Access Token Response Example for RPK Mode, with the Public Key of the RS Wrapped by a CCS, Conveyed within "rs_cnf"

2.2. Raw Public Keys as COSE_Keys Identified by Reference

As per Section 3.2 of [RFC9202], COSE_Key objects [RFC9052] used for specifying raw public keys are transported by value in the Access Token Request and Response messages, as well as within access tokens.

This section extends the DTLS profile by allowing to identifying those COSE_Key objects by reference, alternatively to transporting those by value. Note that only the differences from [RFC9202] are compiled below.

The following relies on the CWT Confirmation Method 'ckt' defined in [RFC9679]. When using a 'ckt' structure, this conveys the thumbprint of a COSE_Key object computed as per Section 3 of [RFC9679]. In particular, the used hash function MUST be SHA-256 [SHA-256], which is mandatory to support when supporting COSE Key thumbprints.

If the raw public key of C is specified as a COSE_Key object COSE_KEY_C and the intent is to identify it by reference, then the following applies.

* The payload of the Access Token Request (see Section 5.8.1 of [RFC9200]) is as defined in Section 3.2.1 of [RFC9202], with the difference that the "req_cnf" parameter [RFC9201] MUST specify a "ckt" structure, with value the thumbprint of COSE_KEY_C.

* The content of the access token that the AS provides to C in the Access Token Response (see Section 5.8.2 of [RFC9200]) is as defined in Section 3.2.1 of [RFC9202], with the difference that the "cnf" claim of the access token MUST specify a "ckt" structure, with value the thumbprint of COSE_KEY_C.

If the raw public key of the RS is specified as a COSE_Key object COSE_KEY_RS and the intent is to identify it by reference, then the following applies.

* The payload of the Access Token Response is as defined in Section 3.2.1 of [RFC9202], with the difference that the "rs_cnf" parameter [RFC9201] MUST specify a "ckt" structure, with value the thumbprint of COSE_KEY_RS.

When both public keys are specified as COSE_Key objects, it is possible to have both transported by value, or both identified by reference, or one transported by value while the other one identified by reference.

Note that the use of COSE Key thumbprints per [RFC9679] is applicable only to authentication credentials that are COSE_Key objects. That is, the ’ckt’ structure MUST NOT be used to identify authentication credentials of other formats and that include a COSE_Key object as part of their content, such as CCSs as defined in Section 2.1 of this document.

2.2.1. Examples

Figure 3 shows an example of Access Token Request from C to the AS.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor2",
  / req_cnf /     4 : {
    / ckt / 5 : h'd3550f1b5b763ee09d058fc7aef69900
                  1279903a4a15bdc3953d32b10f7cb8b1'
  }
}
```

Figure 3: Access Token Request Example for RPK Mode, with the Public Key of C Specified as a COSE_Key Object Identified by Reference within "req_cnf"

Figure 4 shows an example of Access Token Response from the AS to C.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...5532',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's RPK in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    / ckt / 5 : h'db60f4d371fffac3e1040566154a5c36
                  1e0bf835a4ad4c58069cf6edc9ac58a3'
  }
}
```

Figure 4: Access Token Response Example for RPK Mode, with the Public Key of the RS Specified as a COSE_Key Object Identified by Reference within "rs_cnf"

3. Certificate Mode

This section defines a new certificate mode of the DTLS profile, which enables the use of public key certificates to specify the public keys of C and the RS. Compared to the RPK mode defined in Section 3.2 of [RFC9202] and extended in Section 2 of this document, the certificate mode displays the differences compiled below.

The authentication credential of C and/or the RS is a public key certificate, i.e., an X.509 certificate [RFC5280] or a C509 certificate [I-D.ietf-cose-cbor-encoded-cert].

* The CWT Confirmation Methods "x5chain", "x5bag", "c5c", and "c5b" defined in [I-D.ietf-ace-edhoc-oscore-profile] are used to transport such authentication credentials by value.

* The CWT Confirmation Methods "x5t", "x5u", "c5t", and "c5u" defined in [I-D.ietf-ace-edhoc-oscore-profile] are used to identify such authentication credentials by reference.

If the authentication credential AUTH_CRED_C of C is a public key certificate, then the following applies.

* The "req_cnf" parameter [RFC9201] of the Access Token Request (see Section 5.8.1 of [RFC9200]) specifies AUTH_CRED_C as follows.

  If AUTH_CRED_C is an X.509 certificate, the "req_cnf" parameter MUST specify:

  - An "x5chain" or "x5bag" structure, in case AUTH_CRED_C is transported by value within a certificate chain or a certificate bag, respectively; or

  - An "x5t" or "x5u" structure, in case AUTH_CRED_C is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

  If AUTH_CRED_C is a C509 certificate, the "req_cnf" parameter MUST specify:

  - A "c5c" or "c5b" structure, in case AUTH_CRED_C is transported by value within a certificate chain or a certificate bag, respectively; or

  - A "c5t" or "c5u" structure, in case AUTH_CRED_C is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

* The "cnf" claim of the access token that the AS provides to C in the Access Token Response (see Section 5.8.2 of [RFC9200]) specifies AUTH_CRED_C as follows.

  If AUTH_CRED_C is an X.509 certificate, the "cnf" claim MUST specify:

  - An "x5chain" or "x5bag" structure, in case AUTH_CRED_C is transported by value within a certificate chain or a certificate bag, respectively; or

- An "x5t" or "x5u" structure, in case AUTH_CRED_C is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

If AUTH_CRED_C is a C509 certificate, the "cnf" claim MUST specify:

- A "c5c" or "c5b" structure, in case AUTH_CRED_C is transported by value within a certificate chain or a certificate bag, respectively; or

- A "c5t" or "c5u" structure, in case AUTH_CRED_C is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

If the authentication credential AUTH_CRED_RS of the RS is a public key certificate, then the following applies.

* The "rs_cnf" parameter [RFC9201] of the Access Token Response specifies AUTH_CRED_RS as follows.

  If AUTH_CRED_RS is an X.509 certificate, the "rs_cnf" parameter MUST specify:

  - An "x5chain" or "x5bag" structure, in case AUTH_CRED_RS is transported by value within a certificate chain or a certificate bag, respectively; or

  - An "x5t" or "x5u" structure, in case AUTH_CRED_RS is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

  If AUTH_CRED_RS is a C509 certificate, the "rs_cnf" parameter MUST specify:

  - A "c5c" or "c5b" structure, in case AUTH_CRED_RS is transported by value within a certificate chain or a certificate bag, respectively; or

  - A "c5t" or "c5u" structure, in case AUTH_CRED_RS is identified by reference through a hash value (a thumbprint) or a URI [RFC3986], respectively.

For the "req_cnf" parameter of the Access Token Request, the "rs_cnf" parameter of the Access Token Response, and the "cnf" claim of the access token, the structures "x5bag", "x5chain", "x5t", "x5u", "c5b", "c5c", "c5t", and "c5u" are defined in [I-D.ietf-ace-edhoc-oscore-profile], together with their identifiers.

When using either of the structures, the specified authentication credential is just the end-entity certificate.

As per [RFC6347] and [RFC9147], a public key certificate is specified in the Certificate message of the DTLS handshake. For X.509 certificates, the TLS Certificate Type is "X509", as defined in [RFC6091]. For C509 certificates, the TLS certificate type is "C509 Certificate", as defined in [I-D.ietf-cose-cbor-encoded-cert].

It is not required that AUTH_CRED_C and AUTH_CRED_RS are both X.509 certificates or both C509 certificates. Also, it is not required that AUTH_CRED_C and AUTH_CRED_RS are both transported by value or both identified by reference.

Finally, one of the two authentication credentials can be a public key certificate, while the other one can be a raw public key. This is consistent with the admitted, combined use of raw public keys and certificates, as discussed in Section 5.3 of [RFC7250].

3.1. Examples

Figure 5 shows an example of Access Token Request from C to the AS. In the example, C specifies its authentication credential by means of an "x5chain" structure, transporting by value only its X.509 certificate.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor3",
  / req_cnf /     4 : {
    e'x5chain' : h'3081ee3081a1a00302010202046231 9ec430
                   0506032b6570301d311b301906035504030c
                   124544484f4320526f6f7420456432353531
                   39301e170d3232303331363038323433365a
                   170d3239313233313233303030305a302231
                   20301e06035504030c174544484f43205265
                   73706f6e6465722045643235353139302a30
                   0506032b6570032100a1db47b95184854ad1
                   2a0c1a354e418aace33aa0f2c662c00b3ac5
                   5de92f9359300506032b6570034100b723bc
                   01eab0928e8b2b6c98de19cc3823d46e7d69
                   87b032478fecfaf14537a1af14cc8be829c6
                   b73044101837eb4abc949565d86dce51cfae
                   52ab82c152cb02'
  }
}
```

Figure 5: Access Token Request Example for Certificate Mode with an X.509 Certificate as Authentication Credential of C, Transported by Value within "req_cnf"

Figure 6 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of an "x5chain" structure, transporting by value only the X.509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...2fa6',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's X.509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'x5chain' : h'3081ee3081a1a00302010202046231 9ea030
                   0506032b6570301d311b301906035504030c
                   124544484f4320526f6f7420456432353531
                   39301e170d3232303331363038323430305a
                   170d3239313233313233303030305a302231
                   20301e06035504030c174544484f4320496e
                   69746961746f722045643235353139302a30
                   0506032b6570032100ed06a8ae61a829ba5f
                   a54525c9d07f48dd44a302f43e0f23d8cc20
                   b73085141e300506032b6570034100521241
                   d8b3a770996bcfc9b9ead4e7e0a1c0db353a
                   3bdf2910b39275ae48b756015981850d27db
                   6734e37f67212267dd05eeff27b9e7a813fa
                   574b72a00b430b'
  }
}
```

Figure 6: Access Token Response Example for Certificate Mode with an X.509 Certificate as Authentication Credential of the RS, Transported by Value within "rs_cnf"

The following shows a variation of the two previous examples, where X.509 certificates used as authentication credentials are instead identified by reference.

Figure 7 shows an example of Access Token Request from C to the AS. In the example, C specifies its authentication credential by means of an "x5t" structure, identifying by reference its X.509 certificate.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor4",
  / req_cnf /     4 : {
    e'x5t' : [-15, h'79f2a41b510c1f9b']
    / SHA-2 256-bit Hash truncated to 64-bits /
  }
}
```

Figure 7: Access Token Request Example for Certificate Mode with an X.509 Certificate as Authentication Credential of C, Identified by Reference within "req_cnf"

Figure 8 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of an "x5t" structure, identifying by reference the X.509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...cda0',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's X.509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'x5t' : [-15, h'c24ab2fd7643c79f']
    / SHA-2 256-bit Hash truncated to 64-bits /
  }
}
```

Figure 8: Access Token Response Example for Certificate Mode with an X.509 Certificate as Authentication Credential of the RS, Identified by Reference within "rs_cnf"

4. Security Considerations

The security considerations from [RFC9200] and [RFC9202] apply to this document as well. Furthermore:

* When using the CWT Confirmation Method 'ckt' for identifying by reference a COSE_Key object that is used for specifying a raw public key, the security considerations from [RFC9679] apply.

* When using public key certificates as authentication credentials, the security considerations from Appendix C.2 of [RFC8446] apply.

* When using X.509 certificates as authentication credentials, the security considerations from [RFC5280], [RFC6818], [RFC9598], [RFC9549], [RFC9608], and [RFC9618] apply.

* When using C509 certificates as authentication credentials, the security considerations from [I-D.ietf-cose-cbor-encoded-cert] apply.

Consistently with the ACE architecture, C and the RS securely obtain each others' authentication credential from the AS acting as trusted third party, i.e., through the Access Token Response sent to C and the issued access token uploaded to the RS, respectively.

Nevertheless, C and the RS are responsible for verifying the integrity and validity of obtained authentication credentials when those are CCSs or public key certificates as defined in this document.

For public key certificates, verifying their validity may require using a Real-Time Clock (RTC). Trusted certification authorities (CAs) should be selected very carefully and certificate revocation should be supported. The revocation mechanism specifically used depends on the application. For example Certificate Revocation Lists [RFC5280] or the Online Certificate Status Protocol (OCSP) [RFC6960] may be used when authentication credentials are X.509 certificates.

Similarly for CCSs, verifying their validity and handling their revocation require C and the RS to very carefully select relevant trust anchors and to have a well-defined trust-establishment process.

Note that self-signed certificates or CCSs provided to C and the RS cannot result in modifying the set of trust anchors. A common way for a new trust anchor to be added to (or removed from) a device is by performing a firmware upgrade. A longer discussion on trust and validation in constrained devices is provided by [RFC9360].

5. IANA Considerations

This document has no actions for IANA.

6. References

6.1. Normative References

[I-D.ietf-ace-edhoc-oscore-profile]
Selander, G., Mattsson, J. P., Tiloca, M., and R. Hglund, "Ephemeral Diffie-Hellman Over COSE (EDHOC) and Object Security for Constrained Environments (OSCORE) Profile for Authentication and Authorization for Constrained Environments (ACE)", Work in Progress, Internet-Draft, draft-ietf-ace-edhoc-oscore-profile-08, 7 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ace-edhoc-oscore-profile-08>.

[I-D.ietf-cose-cbor-encoded-cert]
Mattsson, J. P., Selander, G., Raza, S., Hglund, J., and M. Furuhed, "CBOR Encoded X.509 Certificates (C509 Certificates)", Work in Progress, Internet-Draft, draft-ietf-cose-cbor-encoded-cert-14, 23 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-cose-cbor-encoded-cert-14>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[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/rfc/rfc3986>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <https://www.rfc-editor.org/rfc/rfc6347>.

[RFC6749] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework", RFC 6749, DOI 10.17487/RFC6749, October 2012, <https://www.rfc-editor.org/rfc/rfc6749>.

[RFC6818] Yee, P., "Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 6818, DOI 10.17487/RFC6818, January 2013, <https://www.rfc-editor.org/rfc/rfc6818>.

[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/rfc/rfc7250>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/rfc/rfc7252>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[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/rfc/rfc8323>.

[RFC8392] Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "CBOR Web Token (CWT)", RFC 8392, DOI 10.17487/RFC8392, May 2018, <https://www.rfc-editor.org/rfc/rfc8392>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/rfc/rfc8610>.

[RFC8747] Jones, M., Seitz, L., Selander, G., Erdtman, S., and H. Tschofenig, "Proof-of-Possession Key Semantics for CBOR Web Tokens (CWTs)", RFC 8747, DOI 10.17487/RFC8747, March 2020, <https://www.rfc-editor.org/rfc/rfc8747>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/rfc/rfc8949>.

[RFC9052] Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", STD 96, RFC 9052, DOI 10.17487/RFC9052, August 2022, <https://www.rfc-editor.org/rfc/rfc9052>.

[RFC9053] Schaad, J., "CBOR Object Signing and Encryption (COSE): Initial Algorithms", RFC 9053, DOI 10.17487/RFC9053, August 2022, <https://www.rfc-editor.org/rfc/rfc9053>.

[RFC9147] Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", RFC 9147, DOI 10.17487/RFC9147, April 2022, <https://www.rfc-editor.org/rfc/rfc9147>.

[RFC9200] Seitz, L., Selander, G., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "Authentication and Authorization for Constrained Environments Using the OAuth 2.0 Framework (ACE-OAuth)", RFC 9200, DOI 10.17487/RFC9200, August 2022, <https://www.rfc-editor.org/rfc/rfc9200>.

[RFC9201] Seitz, L., "Additional OAuth Parameters for Authentication and Authorization for Constrained Environments (ACE)", RFC 9201, DOI 10.17487/RFC9201, August 2022, <https://www.rfc-editor.org/rfc/rfc9201>.

[RFC9202] Gerdes, S., Bergmann, O., Bormann, C., Selander, G., and L. Seitz, "Datagram Transport Layer Security (DTLS) Profile for Authentication and Authorization for Constrained Environments (ACE)", RFC 9202, DOI 10.17487/RFC9202, August 2022, <https://www.rfc-editor.org/rfc/rfc9202>.

[RFC9430] Bergmann, O., Preu Mattsson, J., and G. Selander, "Extension of the Datagram Transport Layer Security (DTLS) Profile for Authentication and Authorization for Constrained Environments (ACE) to Transport Layer Security (TLS)", RFC 9430, DOI 10.17487/RFC9430, July 2023, <https://www.rfc-editor.org/rfc/rfc9430>.

[RFC9549] Housley, R., "Internationalization Updates to RFC 5280", RFC 9549, DOI 10.17487/RFC9549, March 2024, <https://www.rfc-editor.org/rfc/rfc9549>.

[RFC9598] Melnikov, A., Chuang, W., and C. Bonnell, "Internationalized Email Addresses in X.509 Certificates", RFC 9598, DOI 10.17487/RFC9598, May 2024, <https://www.rfc-editor.org/rfc/rfc9598>.

[RFC9608] Housley, R., Okubo, T., and J. Mandel, "No Revocation Available for X.509 Public Key Certificates", RFC 9608, DOI 10.17487/RFC9608, June 2024, <https://www.rfc-editor.org/rfc/rfc9608>.

[RFC9618] Benjamin, D., "Updates to X.509 Policy Validation", RFC 9618, DOI 10.17487/RFC9618, August 2024, <https://www.rfc-editor.org/rfc/rfc9618>.

[RFC9679] Isobe, K., Tschofenig, H., and O. Steele, "CBOR Object Signing and Encryption (COSE) Key Thumbprint", RFC 9679, DOI 10.17487/RFC9679, December 2024, <https://www.rfc-editor.org/rfc/rfc9679>.

[SHA-256] NIST, "Secure Hash Standard", NIST FIPS PUB 180-4, DOI 10.6028/NIST.FIPS.180-4 , August 2015, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf>.

6.2. Informative References

[I-D.ietf-ace-workflow-and-params] Tiloca, M. and G. Selander, "Short Distribution Chain (SDC) Workflow and New OAuth Parameters for the Authentication and Authorization for Constrained Environments (ACE) Framework", Work in Progress, Internet-Draft, draft-ietf-ace-workflow-and-params-04, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ace-workflow-and-params-04>.

[RFC6091] Mavrogiannopoulos, N. and D. Gillmor, "Using OpenPGP Keys for Transport Layer Security (TLS) Authentication", RFC 6091, DOI 10.17487/RFC6091, February 2011, <https://www.rfc-editor.org/rfc/rfc6091>.

[RFC6960] Santesson, S., Myers, M., Ankney, R., Malpani, A., Galperin, S., and C. Adams, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP", RFC 6960, DOI 10.17487/RFC6960, June 2013, <https://www.rfc-editor.org/rfc/rfc6960>.

[RFC9360] Schaad, J., "CBOR Object Signing and Encryption (COSE): Header Parameters for Carrying and Referencing X.509 Certificates", RFC 9360, DOI 10.17487/RFC9360, February 2023, <https://www.rfc-editor.org/rfc/rfc9360>.

Appendix A. Examples with Hybrid Settings

This section provides additional examples where, within the same ACE execution workflow, C and the RS use different formats of raw public keys (see Appendix A.1), or different formats of certificates (see Appendix A.2), or a combination of the RPK mode and certificate mode (see Appendix A.3).

A.1. RPK Mode (Raw Public Keys of Different Formats)

Figure 9 shows an example of Access Token Request from C to the AS, where the public key of C is conveyed as a COSE Key.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor5",
  / req_cnf /     4 : {
    / COSE_Key / 1 : {
      / kty /    1 : 2 / EC2 /,
      / crv /   -1 : 1 / P-256 /,
      / x /     -2 : h'd7cc072de2205bdc1537a543d53c60a6
                      acb62eccd890c7fa27c9e354089bbe13',
      / y /     -3 : h'f95e1d4b851a2cc80fff87d8e23f22af
                      b725d535e515d020731e79a3b4e47120'
    }
  }
}
```

Figure 9: Access Token Request Example for RPK Mode, with the Public Key of C Conveyed as a COSE Key within "req_cnf"

Figure 10 shows an example of Access Token Response from the AS to C, where the public key of the RS is wrapped by a CCS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...c41a',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's RPK in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'kccs' : {
      / sub / 2 : "DD-EE-FF-05-06-07-08-09",
      / cnf / 8 : {
        / COSE_Key / 1 : {
          / kty /  1 : 2 / EC2 /,
          / crv / -1 : 1 / P-256 /,
          / x /   -2 : h'ac75e9ece3e50bfc8ed6039988952240
                         5c47bf16df96660a41298cb4307f7eb6',
          / y /   -3 : h'6e5de611388a4b8a8211334ac7d37ecb
                         52a387d257e6db3c2a93df21ff3affc8'
        }
      }
    }
  }
}
```

Figure 10: Access Token Response Example for RPK Mode, with the Public Key of the RS Wrapped by a CCS within "rs_cnf"

A.2. Certificate Mode (Certificates of Different Formats)

Figure 11 shows an example of Access Token Request from C to the AS. In the example, C specifies its authentication credential by means of an "x5chain" structure, transporting by value only its X.509 certificate.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor6",
  / req_cnf /     4 : {
    e'x5chain' : h'308201383081dea003020102020301f50d30
                   0a06082a8648ce3d0403023016311430120 6
                   035504030c0b52464320746573742043413 0
                   1e170d323330313031303030303030305a170d
                   323630313031303030303030305a3022312030
                   1e06035504030c1730312d32332d34352d46
                   462d46452d36372d38392d41423059301306
                   072a8648ce3d020106082a8648ce3d030107
                   03420004b1216ab96e5b3b3340f5bdf02e69
                   3f16213a04525ed44450b1019c2dfd3838ab
                   ac4e14d86c0983ed5e9eef2448c6861cc406
                   547177e6026030d051f7792ac206a30f300d
                   300b0603551d0f040403020780300a06082a
                   8648ce3d0403020349003046022100d4320b
                   1d6849e309219d30037e138166f2508247dd
                   dae76cceea55053c108e90022100d551f6d6
                   0106f1abb484cfbe6256c178e4ac3314ea19
                   191e8b607da5ae3bda16'
  }
}
```

Figure 11: Access Token Request Example for Certificate Mode with an X.509 Certificate as Authentication Credential of C, Transported by Value within "req_cnf"

Figure 12 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of a "c5c" structure, transporting by value only the C509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...001a',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's C509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'c5c' : h'03487e7661d7b54e46328a23625553066243
                41086b4578616d706c6520496e63096d6365
                72746966696361746 96f6e016a3830322e31
                41522043411a5c52dc0cf68c236255530662
                434105624c41086b6578616d706c6520496e
                630963496f5422665774313233340 15821fd
                c8b421f11c25e47e3ac57123bf2d9fdc494f
                028bc351cc80c03f150bf50cff958a042101
                5496600d8716bf7fd0e752d0ac760777ad66
                5d02a0075468d16551f951bfc82a431d0d9f
                08bc2d205b116021050382208 2492b060104
                01b01f0a014401020304005840c0d81996d2
                507d693f3c48eaa5ee9491bda6db214099d9
                8117c63b361374cd86a774989f4c321a5cf2
                5d832a4d336a08ad67df20f1506421188a0a
                de6d349236'
  }
}
```

Figure 12: Access Token Response Example for Certificate Mode with a C509 Certificate as Authentication Credential of the RS, Transported by Value within "rs_cnf"

The following shows a variation of the two previous examples, where certificates used as authentication credentials are instead identified by reference.

Figure 13 shows an example of Access Token Request from C to the AS. In the example, C specifies its authentication credential by means of an "x5t" structure, identifying by reference its X.509 certificate.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor7",
  / req_cnf /     4 : {
    e'x5t' : [-15, h'6ac62b8f41ba5d99']
    / SHA-2 256-bit Hash truncated to 64-bits /
  }
}
```

Figure 13: Access Token Request Example for Certificate Mode with an X.509 Certificate as Authentication Credential of C, Identified by Reference within "req_cnf"

Figure 14 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of a "c5t" structure, identifying by reference the C509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...cc04',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's X.509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'c5t' : [-15, h'cb247f29c82b933a']
    / SHA-2 256-bit Hash truncated to 64-bits /
  }
}
```

Figure 14: Access Token Response Example for Certificate Mode with a C509 Certificate as Authentication Credential of the RS, Identified by Reference within "rs_cnf"

A.3. Combination of RPK Mode and Certificate Mode

Figure 15 shows an example of Access Token Request from C to the AS, where the public key of C is wrapped by a CCS.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor8",
  / req_cnf /     4 : {
    e'kccs' : {
      / sub / 2 : "55-11-44-AB-CD-EF-00-00",
      / cnf / 8 : {
        / COSE_Key / 1 : {
          / kty /    1 : 2 / EC2 /,
          / crv /   -1 : 1 / P-256 /,
          / x /     -2 : h'cd4177ba62433375ede279b5e18e8b91
                            bc3ed8f1e174474a26fc0edb44ea5373',
          / y /     -3 : h'a0391de29c5c5badda610d4e301eaaa1
                            8422367722289cd18cbe6624e89b9cfd'
        }
      }
    }
  }
}
```

Figure 15: Access Token Request Example for RPK Mode, with the Public Key of C Wrapped by a CCS within "req_cnf"

Figure 16 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of an "x5chain" structure, transporting by value only the X.509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...0f7b',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's X.509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'x5chain' : h'3082023d308201e2a00302010202087e7661
                   d7b54e4632300a06082a8648ce3d04030230
                   5d310b30090603550406130255533310b3009
                   06035504080c024341311430120603550 40a
                   0c0b4578616d706c6520496e633116301406
                   0355040b0c0d6365727469666936174696f
                   6e311330110603550 4030c0a3830322e3141
                   522043413020170d31393031333131313239
                   31365a180f393939393133233313233353935
                   395a305c310b30090603550 4061302555331
                   0b300906035504080c024341310b30090603
                   5504070c024c4131143012060355040a0c0b
                   6578616d706c6520496e63310c300a060355
                   040b0c03496f54310f300d06035504051306
                   577431323334305930 1306072a8648ce3d02
                   0106082a8648ce3d03010703420004c8b421
                   f11c25e47e3ac57123bf2d9fdc494f028bc3
                   51cc80c03f150bf50cff958d75419d81a6a2
                   45dffae790be95cf75f602f9152618f816a2
                   b23b5638e59fd9a3818a3081873009060355
                   1d1304023000301d0603551d0e0416041496
                   600d8716bf7fd0e752d0ac760777ad665d02
                   a0301f0603551d2304183016801468d16551
                   f951bfc82a431d0d9f08bc2d205b1160300e
                   0603551d0f0101ff0404030205a0302a0603
                   551d1104233021a01f06082b060105050708
                   04a013301106092b06010401b43b0a010404
                   01020304300a06082a8648ce3d0403020349
                   003046022100c0d81996d2507d693f3c48ea
                   a5ee9491bda6db214099d98117c63b361374
                   cd86022100a774989f4c321a5cf25d832a4d
                   336a08ad67df20f1506421188a0ade6d3492
                   36'
  }
}
```

Figure 16: Access Token Response Example for Certificate Mode with an X.509 Certificate as Authentication Credential of the RS, Transported by Value within "rs_cnf"

The following shows a variation of the two previous examples, where one authentication credential is a raw public key specified by a COSE_Key Object and the other authentication credential is an X.509 certificate, with both credentials identified by reference.

Figure 17 shows an example of Access Token Request from C to the AS. In the example, C specifies its authentication credential by means of a "ckt" structure, identifying by reference the COSE_Key Object that specifies its public key.

```
POST coaps://as.example.com/token
Content-Format: 19 (application/ace+cbor)
Payload:
{
  / grant_type / 33 : 2 / client_credentials /,
  / audience /    5 : "tempSensor9",
  / req_cnf /     4 : {
    / ckt / 5 : h'29e8a588da26249fc88f3b3f059f2144
                  475c895619d64b2ad4aa2f8a051e8dc9'
  }
}
```

Figure 17: Access Token Request Example for RPK Mode, with the Public Key of C Specified as a COSE_Key Object Identified by Reference within "req_cnf"

Figure 18 shows an example of Access Token Response from the AS to C. In the example, the AS specifies the authentication credential of the RS by means of an "x5t" structure, identifying by reference the X.509 certificate of the RS.

```
2.01 Created
Content-Format: 19 (application/ace+cbor)
Max-Age: 3560
Payload:
{
  / access_token / 1 : h'd83dd083...f3c5',
    / (remainder of CWT omitted for brevity;
    CWT contains the client's X.509 certificate in the cnf claim) /
  / expires_in /   2 : 3600,
  / rs_cnf /      41 : {
    e'x5t' : [-15, h'e35464981de8d29c']
    / SHA-2 256-bit Hash truncated to 64-bits /
  }
}
```

Figure 18: Access Token Response Example for Certificate Mode with an X.509 Certificate as Authentication Credential of the RS, Identified by Reference within "rs_cnf"

Appendix B. CDDL Model

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

```
; CWT Confirmation Methods
x5t = 6
c5t = 8
kccs = 11
x5chain = 24
c5c = 26
```

Figure 19: CDDL Model

Appendix C. Document Updates

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

C.1. Version -01 to -02

* Considerations on providing credentials by value or by reference.
* Minor fixes in examples.
* Added more examples with hybrid settings.
* Extended security considerations.
* Updated CBOR abbreviations for a more efficient use of codepoints.

* Updated references.

* Editorial improvements.

C.2. Version -00 to -01

* Enabled use of COSE Keys identified by reference with a thumbprint.

* Changed CBOR abbreviations to not collide with existing codepoints.

* Fixes in the examples in CBOR diagnostic notation.

* Updated references.

* Editorial improvements.

Acknowledgments

The authors sincerely thank Rikard Hglund and Gran Selander for their comments and feedback.

This work was supported by the Sweden's Innovation Agency VINNOVA within the EUREKA CELTIC-NEXT project CYPRESS; and by the H2020 project SIFIS-Home (Grant agreement 952652).

Authors' Addresses

Marco Tiloca
RISE AB
Isafjordsgatan 22
SE-164 40 Kista
Sweden
Email: marco.tiloca@ri.se

John Preu Mattsson
Ericsson AB
SE-164 80 Stockholm
Sweden
Email: john.mattsson@ericsson.com