Coordinating the Use of Application Profiles for Ephemeral Diffie-Hellman Over COSE (EDHOC)

draft-tiloca-lake-app-profiles-01

Marco Tiloca, RISE
Rikard Höglund, RISE

IETF 119 Meeting – Brisbane – March 21st, 2024

Motivation

- **Peers have to agree about how to run EDHOC and on certain parameters**
  - Some are exchanged during the protocol execution, when a few can be negotiated

- **In general, two peers have to rely on an EDHOC application profile, specifying:**
  - The intended use of EDHOC, including relevant processing and verification
  - Parameters for the EDHOC execution, both in-band and out-of-band ones

- **How to facilitate the definition and discovery of EDHOC application profiles to use?**
  - Related points first raised during the WG Last Call of *draft-ietf-core-oscore-edhoc*
  - Agreed that it was better to address this topic in the LAKE WG

- **From the LAKE Charter:** *The working group will also work on a Standard Track means for coordinating the use and discovery of EDHOC application profiles, the definition of a well-known application profile …*

Scope

- **Definition of "Profile ID" integer identifiers of EDHOC application profiles**
  - Supported by a new IANA registry "EDHOC Application Profiles"

- **Identification of EDHOC application profiles by their Profile ID**
  - In a Web Link, e.g., in Link-Format through a target attribute when CoAP is used
  - In a descriptive object, e.g., in the EDHOC_Information object defined in [1]

- **Canonical description of EDHOC application profiles at specification time**
  - As a CBOR data item; applicable to definition, distribution, and storage of application profiles

- **Definition of one or more "well-known" EDHOC application profiles**
  - Reflecting the most common, expected way(s) to use EDHOC

[1] https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/

Identification of profiles

- **Discovery through Web Linking**
  - When using CoAP, a link-format document (RFC 6690) can have links to EDHOC resources
- **New target attribute 'ed-prof' for such links**
  - Value taken from the 'Profile ID' column of the new "EDHOC Application Profiles" registry
  - It can occur multiple times in the same link
    → Multiple application profiles are supported

```
REQ: GET /.well-known/core

RES: 2.05 Content
    </sensors/temp>;osc,
    </sensors/light>;if=sensor,
    </.well-known/edhoc>;rt=core.edhoc;ed-csuite=0;ed-csuite=2;
        ed-method=0;ed-cred-t=1;ed-cred-t=3;ed-idcred-t=4;
        ed-i;ed-r;ed-comb-req,
    </edhoc-alt>;rt=core.edhoc;ed-prof=500
```

- **Through the EDHOC_Information object [1]**
  - Also used in the ACE workflow considered in [1]
  - Initial set of parameters also defined in [1]
- **This document defines 'app_prof'**
  - New parameter for the EDHOC_Information object
  - Value taken from the 'Profile ID' column of the new "EDHOC Application Profiles" registry

Name	CBOR label	CBOR type	Registry	Description
app_prof	14	int / array	EDHOC Application Profiles Registry	Set of supported EDHOC Application Profiles

Figure 2: EDHOC_Information Parameter "app_prof"

[1] https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/

Canonical description

- **EDHOC_Application_Profile object, as a CBOR map**
  - Defined also a Media Type – Encoding: CBOR Sequence of CBOR maps
- **Possible entries (i.e., considered namespace)**
  - Same of the EDHOC_Information_Object defined in [1]
  - Reuse CBOR abbreviations of map keys from the same “EDHOC Information” registry
- **The following entries MUST be present**
  - ‘app_prof’, identifying the profile in question
  - ‘methods’ and ‘cred_types’
- **The following entries MUST NOT be present**
  - ‘session_id’, ‘uri_path’, ‘initiator’, and ‘responder’
- **Other entries MAY be present**

```
EDHOC_Application_Profile = {
      1 => int / array,     ; methods
      9 => int / array,     ; cred_types
     14 => int,             ; app_prof
   * int / tstr => any
}
```


[1] https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/

Since IETF 118

- **Good feedback from Brian Sipos [2] – Thanks!**
  - "Yes, that draft is actually just what I was looking for. No need to reinvent an encoding for all of those EDHOC options or selection logic."
- **New version -01 submitted before the IETF 119 cut-off**
- **Main updates**
  - Editorial fixes and readability improvements
  - While 'app_prof' stays, other new EDHOC_Information Parameters have been moved out to [1]
  - Clarified that 'initiator' and 'responder' are not included in the EDHOC_Application_Profile object
  - Added further placeholders on what should be in the description of an Application Profile
    - Maximum size of EDHOC messages (see Section 3.4 of RFC 9528)
    - The transport to use for the EDHOC messages
      - This might differ between the two peers (see Section 3.9 of RFC 9528)
      - It might have to be accompanied by transport-specific information/parameters

[1] https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/

[2] https://mailarchive.ietf.org/arch/msg/lake/GQmnLrn1R29o5u3Xy8CVEQT7KYo/

Identification vs. full description

- **EDHOC profiles signaled, e.g, through:**
  - Web Linking, using target attributes
  - An EDHOC_Information object, see [1]

- **In either case, the current approach either:**
  - Spells-out the different features individually; OR
  - Identifies a whole EDHOC profile by its Profile ID

```
REQ: GET /.well-known/core

RES: 2.05 Content
    </sensors/temp>;osc,
    </sensors/light>;if=sensor,
    </.well-known/edhoc>;rt=core.edhoc;ed-csuite=0;ed-csuite=2;
        ed-method=0;ed-cred-t=1;ed-cred-t=3;ed-idcred-t=4;
        ed-i;ed-r;ed-comb-req,
    </edhoc-alt>;rt=core.edhoc;ed-prof=500
```

- **Proposal: admit one hybrid exception**
  - When using Profile ID, possible to also separately specify EAD items
  - Supported EAD items: those of the profile identified by Profile ID plus the separate ones
  - Example for Web Linking: *</edhoc-alt>;rt=core.edhoc;ed-prof=500;ed-ead=12;ed-ead=47*
  - Avoid registrations of (many) very similar EDHOC profiles differing only about supported EAD items

- **This has to align with the use of the 'app_prof' paramemer in [1]**

Thoughts? Objections?

[1] https://datatracker.ietf.org/doc/draft-ietf-ace-edhoc-oscore-profile/

Still missing parameters

- **Maximum message size**
  - This can trivially be an unsigned integer
- **Type(s) of endpoint identifiers (e.g., EUI-64)**
  - Planned new, dedicated parameter ‘id_ep_type’
  - Need for a new registry to coordinate the types of EDHOC endpoint identifiers
- **Transport(s) to use for EDHOC**
  - Planned new parameters ‘tp_i’ (for the Initiator), ‘tp_r’ (for the Responder), ‘tp’ (common to both)
  - Need for a new registry to coordinate the (integer) identifiers of EDHOC transports
  - A transport might in turn come with additional specific information (as if a “transport suite”)
- **Is something else still missing?**

Thoughts?

Well-known profiles

- **What they are supposed to mean**
  - They reflect what is most common and expected to use for EDHOC
  - NOT "default profile" to use if nothing else is said, overriding what is mandatory to implement
  - NOT necessarily supported by the /.well-known/edhoc resource if nothing else is said

Early proposal: possible well-known profiles to consider for registration – Thoughts?

MINIMAL_CS_2 cypher_suites: 2 ; methods: 3 cred_types: CCS ; id_cred_types: kid (*like the second trace of RFC 9529*)	**MINIMAL_CS_0** cypher_suites: 0 ; methods: 3 cred_types: CCS ; id_cred_types: kid	**ADVANCED** cypher_suites: 0,1,2,3 ; methods: 0,1,2,3 cred_types: CCS , CWT, X.509, C509 id_cred_types: kid, kccs, kcwt, x5t, x5chain, c5t, c5chain
BASIC_CS_2_X509 cypher_suites: 2 ; methods: 0, 3 cred_types: CCS, X.509 id_cred_types: kid, x5t	**BASIC_CS_0_X509** cypher_suites: 0 ; methods: 0, 3 cred_types: CCS, X.509 id_cred_types: kid, x5t	**INTERMEDIATE_CS_2** cypher_suites: 2 ; methods: 0, 3 cred_types: CCS, X.509, C509 id_cred_types: kid, kccs, x5t, x5chain, c5t, c5c
BASIC_CS_2_C509 cypher_suites: 2 ; methods: 0, 3 cred_types: CCS, C509 id_cred_types: kid, c5t	**BASIC_CS_0_C509** cypher_suites: 0 ; methods: 0, 3 cred_types: CCS, C509 id_cred_types: kid, c5t	**INTERMEDIATE_CS_0** cypher_suites: 0 ; methods: 0, 3 cred_types: CCS, X.509, C509 id_cred_types: kid, kccs, x5t, x5chain, c5t, c5c

Summary and next steps

- **Assist the discovery and use of EDHOC application profiles**
  - Definition of integer identifiers of EDHOC application profiles
  - Canonical description of EDHOC application profiles in CBOR
  - Definition of one or more “well-known” EDHOC application profiles

- **Plan for the next version**
  - Mention examples of how profile descriptions can be distributed (see exchange with Brian Sipos [2])
  - Define missing parameters: max message size; peer identifier type(s); transport(s) to use
  - Propose a small set of candidate EDHOC application profiles to register
    - What they specify, and their representation as a CBOR data item

- **Reviews and input are welcome!**

[2] https://mailarchive.ietf.org/arch/msg/lake/GQmnLrn1R29o5u3Xy8CVEQT7KYo/

Thank you!

https://gitlab.com/crimson84/draft-tiloca-lake-app-profiles