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 21\textsuperscript{st}, 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

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

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

Identification of profiles

REQ: GET /.well-known/core
RES: 2.05 Content
  
&lt;/sensors/temp&gt;;osc,
&lt;/sensors/light&gt;;if=sensor,
&lt;/well-known/edhoc&gt;;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-reg,
&lt;/edhoc-alt&gt;;rt=core.edhoc;ed-prof=500

Figure 2: EDHOC_Information Parameter "app_prof"

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

› Other entries MAY be present

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

[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

› 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' parameter in [1] 思前想后？反对意见？

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?**
### 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?

<table>
<thead>
<tr>
<th>Profile</th>
<th>Cypher Suites</th>
<th>Methods</th>
<th>Cred Types</th>
<th>Id Cred Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINIMAL_CS_2</td>
<td>2</td>
<td>3</td>
<td>CCS</td>
<td>kid</td>
</tr>
<tr>
<td>MINIMAL_CS_0</td>
<td>0</td>
<td>3</td>
<td>CCS</td>
<td>kid</td>
</tr>
<tr>
<td>ADVANCED</td>
<td>1, 2, 3</td>
<td>0, 1, 2, 3</td>
<td>CCS, X.509, C509</td>
<td>kid, kccs, kcwt, x5t, x5chain, c5t, c5chain</td>
</tr>
<tr>
<td>BASIC_CS_2_X509</td>
<td>2</td>
<td>0, 3</td>
<td>CCS, X.509</td>
<td>kid, x5t</td>
</tr>
<tr>
<td>BASIC_CS_0_X509</td>
<td>0</td>
<td>0, 3</td>
<td>CCS, X.509</td>
<td>kid, x5t</td>
</tr>
<tr>
<td>BASIC_CS_2_C509</td>
<td>2</td>
<td>0, 3</td>
<td>CCS, C509</td>
<td>kid, c5t</td>
</tr>
<tr>
<td>BASIC_CS_0_C509</td>
<td>0</td>
<td>0, 3</td>
<td>CCS, C509</td>
<td>kid, c5t</td>
</tr>
<tr>
<td>INTERMEDIATE_CS_2</td>
<td>2</td>
<td>0, 3</td>
<td>CCS, X.509, C509</td>
<td>kid, kccs, x5t, x5chain, c5t, c5c</td>
</tr>
<tr>
<td>INTERMEDIATE_CS_0</td>
<td>0</td>
<td>0, 3</td>
<td>CCS, X.509, C509</td>
<td>kid, kccs, x5t, x5chain, c5t, c5c</td>
</tr>
</tbody>
</table>
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