EDHOC Status & Open Issues
draft-ietf-lake-edhoc-08
IETF 111, LAKE WG, July 29, 2021
Outline

— Main changes since IETF 110
  — 05 → 08
— Changes post 08
— Security levels and design goals of EDHOC
— Open Github issues
— Next steps
Main changes to message flow

- Correlation factor “corr” removed
- METHOD_CORR → METHOD
- Use of optional connection identifiers for correlation delegated to transport
- Auxiliary Data → External Authorization Data
1. Introduction
2. EDHOC Outline
3. Protocol Elements
   3.1. General
   3.2. Method and Correlation
   3.3. Authentication Parameters
   3.4. Cipher Suites
   3.5. Ephemeral Public Keys
   3.6. Auxiliary Data
   3.7. Communication of Protocol Features
4. Key Derivation
5. Message Formatting and Processing
   5.1. Encoding of bstr_identifier
   5.2. EDHOC Message 1
   5.3. EDHOC Message 2
   5.4. EDHOC Message 3
6. Error Handling
   6.1. EDHOC Error Message
7. Transferring EDHOC and Deriving an OSCORE Context
   7.1. EDHOC Message 4
   7.2. Transferring EDHOC in CoAP
8. Security Considerations
9. IANA Considerations
10. References
Appendix A. Use of CBOR, CDDL and COSE in EDHOC
Appendix B. Test Vectors
Appendix C. Applicability Statement
Appendix E. Applicability Template

Appendix A. Use with OSCORE and Transfer over CoAP
Appendix B. Compact Representation
Appendix C. Use of CBOR, CDDL and COSE in EDHOC
Appendix D. Test Vectors
Appendix E. Applicability Template
Appendix F. EDHOC Message Deduplication
Appendix G. Transports Not Natively Providing Correlation
Appendix H. Change Log
Main changes –05 → –08

— “corr” removed
  — Transport is now responsible for correlation
  — May use connection identifier
  — Specified for CoAP

— “bstr_identifier” removed
  — Connection identifiers are bstr / int
  — Mapping to OSCORE Sender ID specified
  — Applications using OSCORE must avoid collisions
  — New COSE header “kid2” of type bstr / int

— Error message update following IETF 110
  — Error codes
  — IANA registration
  — Success, error code 0, not transported

— Changed transcript hash definition for TH_2 and TH_3
— New application defined parameter “context” in EDHOC-Exporter
— Details on compact representation of ephemeral public key
— New cipher suites based on Chacha20/Poly1305

— Changed normative language
  — for failure from “MUST” to “SHOULD” send error
  — for send error as successful request and response from “SHOULD” to “”

— IANA registration
— Success, error code 0, not transported
Main changes –05 → –08

— Terminology
  — "protocol instance" → "session"
  — "Auxiliary Data" → "External Authorization Data"

— Clarifications
  — CBOR refers to Deterministically Encoded CBOR
  — Distinction between different uses of Connection id
  — External Authorization Data requires specification and registration of EAD type
  — Added encryption of EAD_4 to message_4
  — Comments by Marco Tiloca, Mališa Vučinic and Peter van der Stok
    — Thanks!

— New appendices
  A. Use with OSCORE and Transfer over CoAP
  B. Compact representation of EC points
  C. CDDL definitions
  D. Message deduplication
  E. Transports not natively providing correlation
  F. Change log

— IANA registrations
  — New registry "EDHOC Exporter Label"
  — New registry "EDHOC Error Code"
  — New registry "EDHOC External Authorization Data"
  — COSE header parameter "cwt" for CWT or UCCS
  — COSE header parameter and COSE key common param "kid2"

— Updated references
— Additional security considerations
Post –08

- RPK by value, format of CRED_x
  - Define Unprotected CWT Claims Set (UCCS)
    - CWT without COSE wrapping
    - Informative reference to draft-ietf-rats-uccs
  - Refactoring of section 3.5
    - Added use of CWT analogous to certificates
    - Replaced example of CRED_x with UCCS instead of non-standard COSE_Key
    - Harmonization of text, editorials

- Please review Section 3.5 (on the master branch)
- Hopefully resolved issues: #125, #115, #88

- Clarifications, last fixes to be able to close issues
- Terminology
  - ‘null’ → ‘nil’

<table>
<thead>
<tr>
<th>OLD EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRED_x = {</td>
</tr>
<tr>
<td>1: 1, /OKP/</td>
</tr>
<tr>
<td>-1: 4, /X25519/</td>
</tr>
<tr>
<td>-2: h'bla3e89460e88d3a8d54211dc95f0b903ff205eb719126db8f4af980d2db83a',</td>
</tr>
<tr>
<td>&quot;subject name&quot;: &quot;42-50-31-FF-EF-37-32-39&quot;</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NEW EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRED_x = { /UCCS/</td>
</tr>
<tr>
<td>8:{ /cnf/</td>
</tr>
<tr>
<td>1:{ /COSE_Key/</td>
</tr>
<tr>
<td>1: 1,</td>
</tr>
<tr>
<td>-1: 4,</td>
</tr>
<tr>
<td>-2: h'bla3e89460e88d3a8d54211dc95f0b903ff205eb719126db8f4af980d2db83a',</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>
Security Levels and Design Goals

- John has written a short text about security levels and design goals of EDHOC
  - Request from people working on formal verification of EDHOC
- Comments welcome
Open Github Issues

— #134 Detailing EDHOC error in CoAP error
— #133 "Empty byte string"
— #125 CRED_x in CWT format
— #121 Simpler more secure MAC calculation
  #115 Transfer CWT
— #103 Optimization of message size
— #100 Scope of the applicability statement
— #99 TEE security consideration
— #88 Opportunistic use
— #84 Make .well-known/edhoc specific to OSCORE
— #81 Effects of limited amounts of randomness
— #78 JSON test vector format for automatic parsing
— #73 MTI section
— #64 Max retransmissions of EDHOC messages
— #50 Add cipher suite with Wei25519
— #47 Test vectors additions
— #22 Mandatory to implement cipher suite
— #8 Verification of intended peer
Simpler more secure MAC calculation (#121)

Current inner MACs (MAC2/3) are calculated with COSE_Encrypt0

Proposal: Replace with single invocation of EDHOC-KDF()

- Improved security, no dependence on encryption algorithm
- Simpler to specify and implement
  - "K_2m", "K_3m", "IV_2m", "IV_3m" can be removed.

Candidates:
1. context as CBOR sequence, see PR #136:
   - \text{MAC}_3 = \text{EDHOC-KDF}(\text{PRK}_{4x3m}, \text{TH}_3, \\
   \text{"MAC}_3", (\text{ID\_CRED\_R}, \text{CRED\_R}, ? \text{EAD}_3), \\
   \text{mac\_length})

2. context as CBOR array, see PR #137:
   - \text{MAC}_3 = \text{EDHOC-KDF}(\text{PRK}_{4x3m}, \text{TH}_3, \\
   \text{"MAC}_3", [\text{ID\_CRED\_R}, \text{CRED\_R}, ? \text{EAD}_3], \\
   \text{mac\_length})

OLD

* Compute an inner COSE_Encrypt0
  * protected = \text{<< ... >>}
  * external_aad = \text{<< ... >>}

  * Plaintext = 0x
  * Key K = EDHOC-KDF( ... )
  * Nonce N = EDHOC-KDF( ... )

\text{MAC}_x = \text{'ciphertext'} of COSE_Encrypt0 with AEAD

NEW

Compute \text{MAC}_x = \text{EDHOC-KDF}(\text{PRK}, \\
\text{TH}, \langle\text{label & context}\rangle, L)
Optimization of message sizes (#103)

(Discussed at a recent LAKE WG interim meeting)

— General question: What is the right level of message size optimizations?
  — Trade-off between encoding complexity and message size

— Target message sizes (draft-ietf-lake-reqs-04) derived from NB-IoT, 6TiSCH and LoRaWAN benchmarks
  — More detailed estimates of overhead in 6TiSCH available here:
    — https://docs.google.com/spreadsheets/d/1FjlGrgVu6ZSw2bt1KxAS9O5gtGXxBoGnqK3QeVFGx1Q/edit#gid=0

— Current messages comply with the LAKE requirements except for message_2 which is 1 byte too large
  — Further single-byte optimizations of message sizes are possible
  — E.g., using known lengths, concatenate G_Y and CIPHERTEXT2 into one CBOR bstr
    — Should we do this?

— In the interim meeting we agreed to revisit this issue just before WGLC when message sizes are fixed
Mandatory to implement cipher suite (#22)

- Has been discussed at length.
- Potential resolution in the draft.

- Summary of the issue in
  https://mailarchive.ietf.org/arch/msg/lake/75nRaD6czYG6RqLT06Qe8C_lsaM/

- Proposal: Keep open until WGLC
  - then close if people are happy with the formulation in the spec.
  - else make a vote
‘kid2’

— Background
  — ‘kid’ is both COSE header parameter and COSE key common parameter
    — the former is used to reference a key/credential using the latter
  — ‘kid’ is of type CBOR bstr, which only has one 1-byte value: the empty byte string h"
  — want to be able to use the type CBOR int for more 1-byte values (integers -24 to 23)
— Candidate solutions
  1. Redefine ‘kid’ as bstr / int
     — Would apply to both COSE header and COSE key common parameter ‘kid’
     — Backward compatibility issues?
  2. Re-introduce bstr_identifier
  3. Define new COSE header parameter and COSE key common parameter ‘kid2’ as bstr / int
     — may be used to identify a key stored in a UCCS, in a CWT, or in a certificate.
     — may point to a COSE key common parameter ‘kid’ or ‘kid2’

Discuss
Next steps

- Close issue #121 Simplify MAC
- Resolve ‘kid2’
- Submit version -09
  - Unless further substantial issues, this is the next implementation version
- Update test vectors in the draft
- Help with test vector related issues #47 and #78 is welcome!
- Close minor issues

More reviews are welcome!