

# Ephemeral Diffie-Hellman Over COSE (EDHOC) and Object Security for Constrained Environments (OSCORE) Profile for Authentication and Authorization for Constrained Environments (ACE)

*draft-ietf-ace-edhoc-oscore-profile-10*

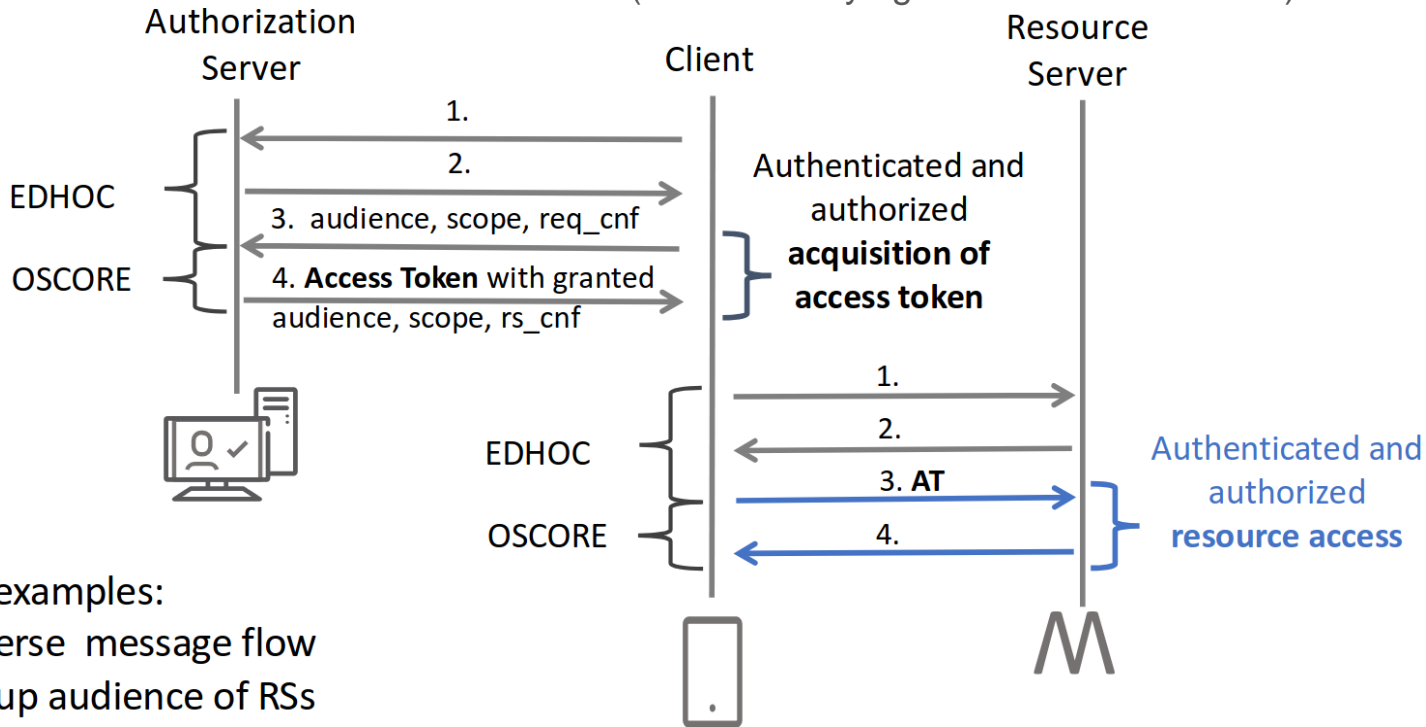
Göran Selander, Ericsson  
John Preuß Mattsson, Ericsson  
Marco Tiloca, RISE  
**Rikard Höglund, RISE**

IETF 125 meeting – Shenzhen – March 17<sup>th</sup>, 2026

# Overview

## › Profile of the ACE framework

1. Uses EDHOC for key establishment (and Access Token uploading in an EAD item)
2. Uses OSCORE for secure communication (based on keying material from EDHOC)



— Other examples:

— Reverse message flow

— Group audience of RSs

# Main Updates in v -10

- › **Removed some parameters from the EDHOC\_Information object**
  - Based on feedback from previous IETF meetings and discussions
  - Some parameters in the object were considered to be excessive or unclear in their usefulness
  - The following parameters have been removed:
    - › max\_msgsize: Maximum size of messages
    - › coap\_ct: Mandate use of the CoAP Content-Format Option
    - › ep\_id\_types: Supported types of endpoint identities
    - › transports: Supported means for transporting messages (e.g. CoAP over UDP)
  - Also removed registries "EDHOC Transports registry" & "EDHOC Endpoint Identity Types registry" as a side-effect
- › **Various editorial fixes and improvements**
- › **Minor updates based on IANA early review**

# Main Updates in v -10

## › Fixed CDDL definition of the EDHOC\_Information object

- It includes information that guides two peers on how to execute the EDHOC protocol
- CDDL definition in Section 3.4 did not validate, and has been updated to be correct

```
EDHOC_Information = {  
  ? 0 => bstr,                ; id  
  ? 1 => int / array,         ; methods  
  ? 2 => int / array,         ; cipher_suites  
  ? 3 => true / false,       ; message_4  
  ? 4 => true / false,       ; comb_req  
  ? 5 => tstr,                ; uri_path  
  ? 6 => int / array,         ; cred_types  
  ? 7 => int / tstr / array,  ; id_cred_types  
  ? 8 => uint / array,        ; eads  
  ? 9 => true / false,       ; initiator  
  ? 10 => true / false,      ; responder  
  ? 11 => uint,               ; max_msgsize  
  ? 12 => true / false,      ; coap_ct  
  ? 13 => int / array,       ; ep_id_types  
  ? 14 => int / array,       ; transports  
  ? 15 => map,                ; trust_anchors  
  * int / tstr => any  
}
```



```
EDHOC_Information = {  
  ? 0 => bstr,                ; id  
  ? 1 => int / [2* int],      ; methods  
  ? 2 => int / [2* int],      ; cipher_suites  
  ? 3 => true / false,       ; message_4  
  ? 4 => true / false,       ; comb_req  
  ? 5 => tstr,                ; uri_path  
  ? 6 => int / [2* int],      ; cred_types  
  ? 7 => int / tstr / [2* (int / tstr)], ; id_cred_types  
  ? 8 => uint / [2* uint],    ; eads  
  ? 9 => true / false,       ; initiator  
  ? 10 => true / false,      ; responder  
  ? 11 => trust_anchors_value, ; trust_anchors  
  * (int / tstr) => any  
}  
  
trust_anchors_value = {  
  1* int => trust_anchors_outer_entry_value  
}  
  
trust_anchors_outer_entry_value =  
  trust_anchors_container / [2* trust_anchors_container]  
  
trust_anchors_container = {  
  int => trust_anchors_inner_entry_value  
}  
  
trust_anchors_inner_entry_value = any
```

# Main Updates in v -10

## › Defined derivation of N\_S when combined with specific application profiles of ACE

- Scenario: The EDHOC and OSCORE profile can be used in combination ACE applications profiles such as ace-key-groupcomm-oscore and ace-coap-pubsub-profile
- Problem:
  - › These application profiles require the ACE client to receive a challenge N\_S from the RS. N\_S is in turn used by the client to perform proof of possession.
  - › Since in this profile the client does not upload the first access token of a token series to the /authz-info endpoint, N\_S is never received by the client.
- Solution: When this profile is used N\_S is derived by use of the EDHOC\_Exporter as follows:
  - › The 'exporter\_label' parameter is an unsigned integer depending on the specific application profile
    - Registration of two EDHOC Exporter Labels (26 & 27) for the two application profiles of RFC9594
  - › The 'context' parameter is h" (0x40)
  - › The 'length' parameter is 32, i.e., the intended length of N\_S in bytes

```
EDHOC_Exporter(exporter_label, context, length)  
= EDHOC_KDF(PRK_exporter, exporter_label, context, length)
```

# Early allocation

## › Early allocation of codepoints

- We'd like to do early allocation for 3 registries.
- This was raised through feedback from other IETF collaborators.

## › JWT Confirmation Methods registry

- x5c, x5b, x5t, x5u, c5c, c5b, c5t, c5u, kcwt, kccs

## › CWT Confirmation Methods

- x5chain: 24
- x5bag: 25
- x5t: 6
- x5u: 26
- c5c: 27
- c5b: 28
- c5t: 7
- c5u: 29
- kcwt: 8
- kccs: 9

## › EDHOC External Authorization Data registry

- ACE-OAuth Access Token: 24
- Session ID: 1
- Credential By Value: 15
- Request Creation Hints: 2



Already in use by aiocoap and Ariel OS's coapcore

# Next Steps

- › **Privacy & security implications of including the access token in EAD\_2, EAD\_3 or EAD\_4**
- › **Extended example for the Request Creation Hints EAD item**
  - Add a message exchange diagram showing the message flow
- › **Process Christian's review of version -09**
- › **Considerations about keeping knowledge about C and RS up-to-date at the AS**
- › **Comments and reviews are welcome!**

Thank you!

Comments/questions?

<https://github.com/ace-wg/ace-edhoc-oscore-profile>