

Using the Extensible Authentication Protocol with Ephemeral Diffie-Hellman over COSE (EDHOC)

draft-ingles-eap-edhoc-02

Eduardo Inglés Sanchez, University of Murcia
Dan Garcia-Carrillo, University of Oviedo (*presenter*)
Rafael Marín-López, University of Murcia
Göran Selander, Ericsson
John Preuß Mattsson, Ericsson

IETF 114, EMU WG, July 27th, 2022

Summary -02

- EAP-EDHOC exchange
- Privacy friendly Response/Identity
- Fragmentation
- Alternate success indication with EDHOC message_4
- Added error use cases

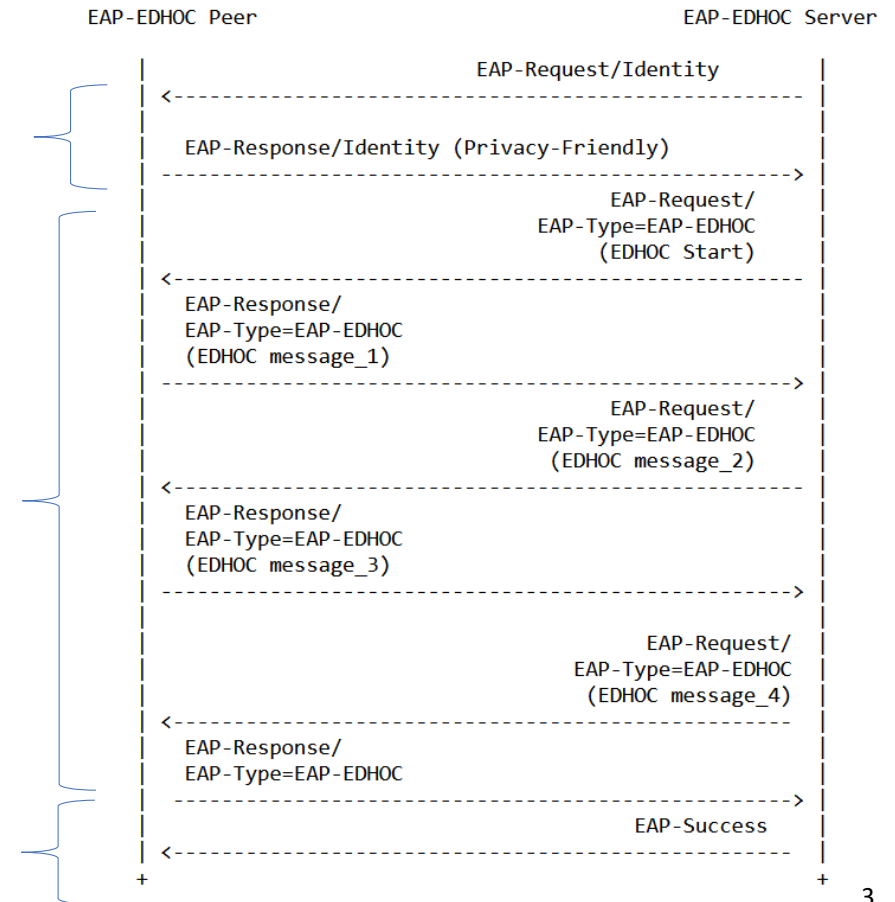
EAP-EDHOC exchange

- A bit of recap

EAP Request/Response

EAP EDHOC Exchange

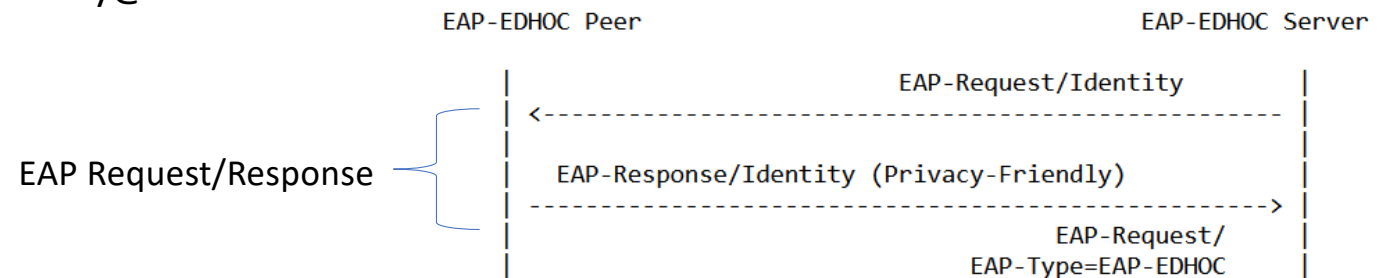
EAP Success



Privacy friendly Response/Identity

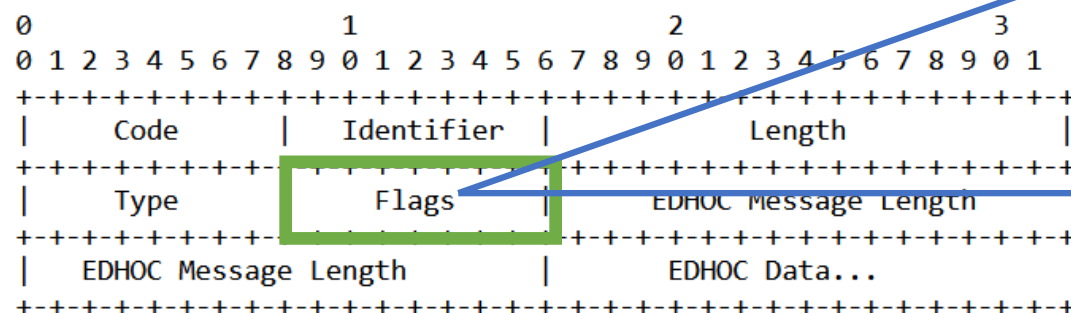
Approach to use privacy friendly Response Identity

- Added requirement to avoid permanent identifiers in clear text
- With NAI
 - Added recommendation to omit username or similar
 - @realm
 - anonymouse@realm
 - encryptedIdentity@realm



Fragmentation

- Added structure for fragmentation support in EAP request and response
 - Use of flags like EAP-TLS
 - EDHOC message field



```

0 1 2 3 4 5 6 7 8
+++++
|L M S R R R R R|
+++++
    
```

L = Length included
 M = More fragments
 S = EAP-EDHOC start
 R = Reserved

```


0 1 2 3 4 5 6 7 8
+++++
|L M R R R R R R|
+++++
    
```

L = Length included
 M = More fragments
 R = Reserved

Alternate success indication with EDHOC message_4

- Use of internal success indication
- EDHOC message_4 already provides a success indication

```
message_4 = (  
    CIPHERTEXT_4 : bstr,
```



CIPHERTEXT_4 is the 'ciphertext' of COSE_Encrypt0.
)

Added error use cases

- Added the different use cases when the different EAP-EDHOC messages are rejected by the receiving counterpart.
 - Error processing message_1, message_2, message_3, message_4
- All errors are followed by the EAP-EDHOC error message

```
error = (  
    ERR_CODE : int,  
    ERR_INFO : any,  
)
```

Next steps

- No major changes are expected
- More reviews are welcome
- Adoption?

Thank you!