EAP-based Authentication Service for CoAP

draft-ietf-ace-wg-coap-eap-04

Rafael Marín-López, University of Murcia
Dan García-Carrillo, University of Oviedo

IETF Meeting, November 9th, 2021
Summary of v04

- Discovery of EAP authenticator
- Sending server resource in the first message
- Cryptosuite negotiation and SID and RID
- OSCORE for keys confirmation in CoAP-EAP
- Considerations for Proxies
- Extensible CBOR structure
- Current flow of operation
- DTLS exchange in Annex
Discovery of the EAP authenticator

• Out of scope

• A brief discussion on this will be added to the next version - 04
  • First approach, to receive the IPv6 of the Border Router (e.g., RA) and send there the initial message
  
Other approaches to be considered

• DHCPv6 [RFC8415]

• mDNS [RFC6762]
Sending the resource on the first message

- Saves bytes over the air: well-known only sent once
- Avoids the CoAP server receiving unexpected well-known messages
Crytposuite negotiation and SID and RID

• The cryptosuite is negotiated in the next exchange
• The entities choose the RID and SID for OSCORE
OSCORE for keys confirmation in CoAP-EAP

• OSCORE for key confirmation and establishing its Security Association
  • An OSCORE message can be treated as alternate success indication
  • An OSCORE security context can be pre-defined, leaving the key to be completed after the EAP success is processed and the MSK is retrieved to complete security context
  • Recipient and Sender ID are now sent in Steps 1 and 2
Consideration for proxies

- There is a role reversal in the first and second message.
- This has to be considered when using proxies
Tagged CBOR structure

CoAP-EAP_Info = {
    ? 1 : array, ; ciphersuite
    ? 2 : bstr, ; SID
    ? 3 : bstr, ; RID
    ? 4 : uint ; Session-Lifetime
}

Extensible CBOR structure
Current flow of operation

POST /well-known/coap-eap [Options(No-response), Payload(/x)]

POST /x [ Payload(EAP Req/Id || CS || SID) ]

{ 2.01 Created, Options(Location-Path(/y), Payload(EAP Resp/Id || CS || RID) )

POST /y [Payload(EAP-Req)]

[2.01 Created, Location-Path(/z), Payload(EAP-X-Res 2) ]

POST /z [Payload(EAP-X-Req)]

[ 2.01 Created, Location-Path(/w), Payload(EAP-X-Resp) ]

POST /w [ Option(OSCORE), Payload(EAP Success || *Session-Lifetime) ]

[ 2.04 Changed, Options(OSCORE) ]
DTLS considerations in Annex

• In DTLS we reuse the same fields as OSCORE
  • Cryptosuite negotiation
  • The key ID is generated by concatenating SID and RID
  • Client Hello message is used as alternate success indication
THANK YOU