

EAP-based Authentication Service for CoAP

Work in progress for:
draft-ietf-ace-wg-coap-eap-01

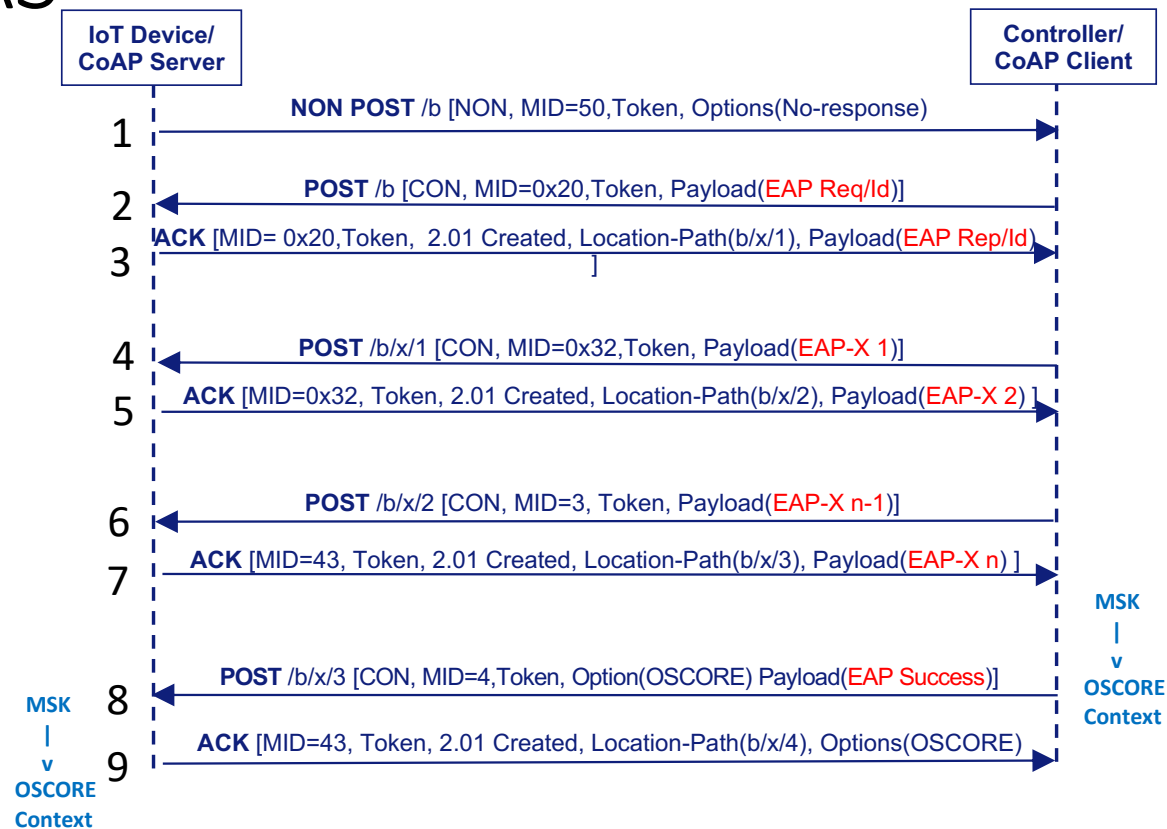
Rafael Marín-López, University of Murcia
Dan García-Carrillo, University of Oviedo

ACE Interim Meeting, May 11th, 2021

CoAP-EAP – Ordering guarantee following HATEOAS

- General Service URI **/b**
 - Each step within the authentication creates a new resource with structure
 - **/b/x/i**
 - x -> Random number representing the ongoing authentication process
 - i -> Monotonic increasing number that represents the current step in the authN process
- (Note:** that it is a lockstep protocol)

CoAP-EAP – Ordering guarantee following HATEOAS

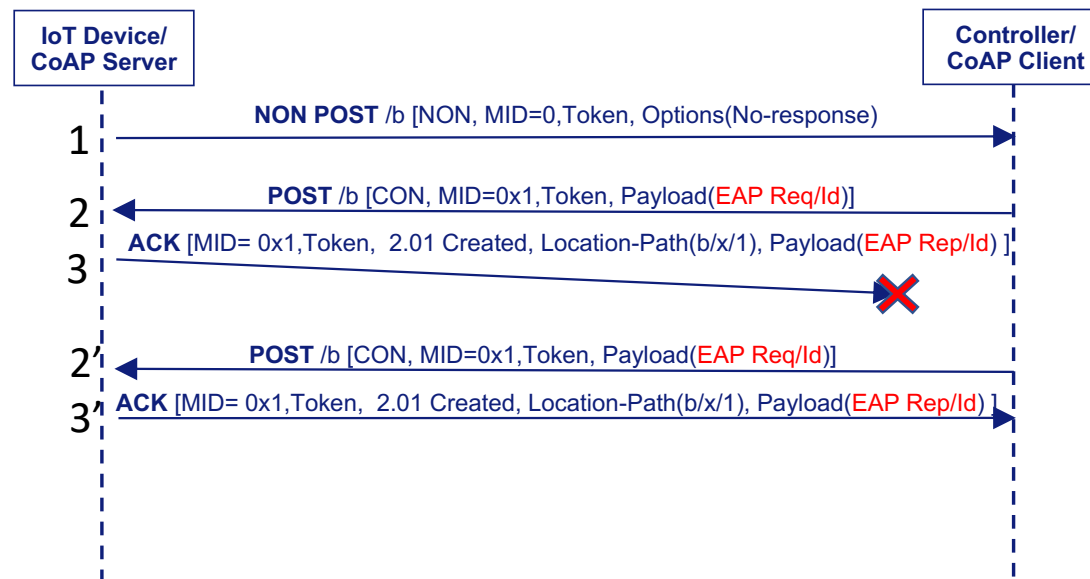


CoAP-EAP – Ordering guarantee following HATEOAS

- For each CoAP request (which contains an EAP Request) the process is
 - Receive the EAP payload and process it
 - Send the content to the EAP state machine
 - Receive the response from the EAP state machine
 - If an error occurs an error message is returned depending on the cause of the error.
 - If everything goes as expected:
 - A new resource is created, `/b/x/i+1`
 - The previous resource `/b/x/l` is deleted
 - A response specifying the new resource is sent back

CoAP-EAP – Ordering guarantee following HATEOAS

- Casuistic when messages are lost
 - If the piggybacked response with a new resource is lost
 - The CoAP client will continue to retransmit until the response arrives
 - The CoAP server will recognize the message as retransmission and resend the message



CoAP-EAP – Ordering guarantee following HATEOAS

- Casuistic when an old message arrives

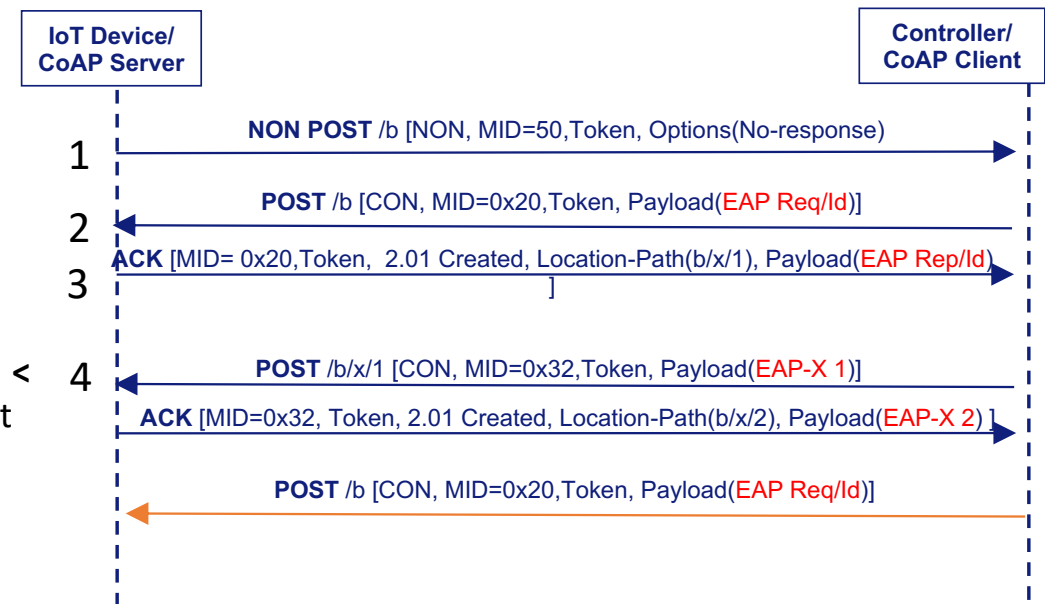
IF managed at CoAP engine

- If the CoAP engine takes care of it, as the server recognizes de old message it can send a stored copy
Then the client would recognize MSGID and that he got the response already, dropping it

IF managed at Application

- If the control in the server goes up to the application, it generates a 4.04 not found since its deleted

Then the client would recognize the MSGID
and that he got the response already, dropping it



THANK YOU