EAP-based Key Establishment for CoAP

(draft-ohba-eap-based-bootstrapping-00)

Subir Das (Telcordia)

Yoshihiro Ohba (Toshiba)

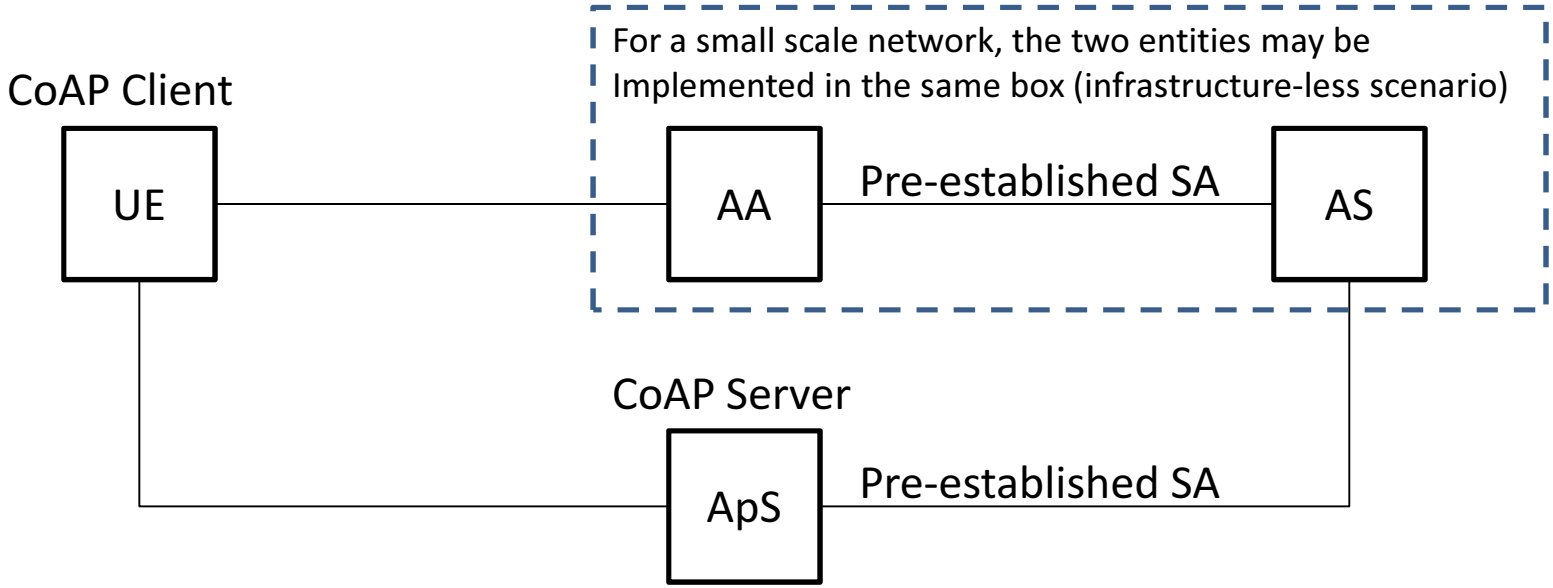
Objectives

- Providing an automated authentication & key establishment mechanism for CoAP security
  - DTLS-PSK (to establish a TLS connection)
  - PSK-mode of IKEv2 (to establish an IPsec SA)
- Goal: Reduce # of public key cryptographic operations
  - We consider resource-constrained devices each of which may communicate with multiple CoAP servers
- Initial scope: unicast security

Use Cases

- Use Case 1: Non-integrated with Network Access Authentication
  - No assumption on business relationships between access network provider and application service provider
- Use Case 2: Integrated with Network Access Authentication
  - Assumes business relationships between access network provider and application service provider

Solution Architecture



Assumptions
- Pre-configured credentials between UE and AS
- UE can discover AA and ApS
- Pre-established SAs for AA-AS and ApS-AS

UE: User Equipment
AA: Authentication Agent
AS: Authentication Server
ApS: Application Server

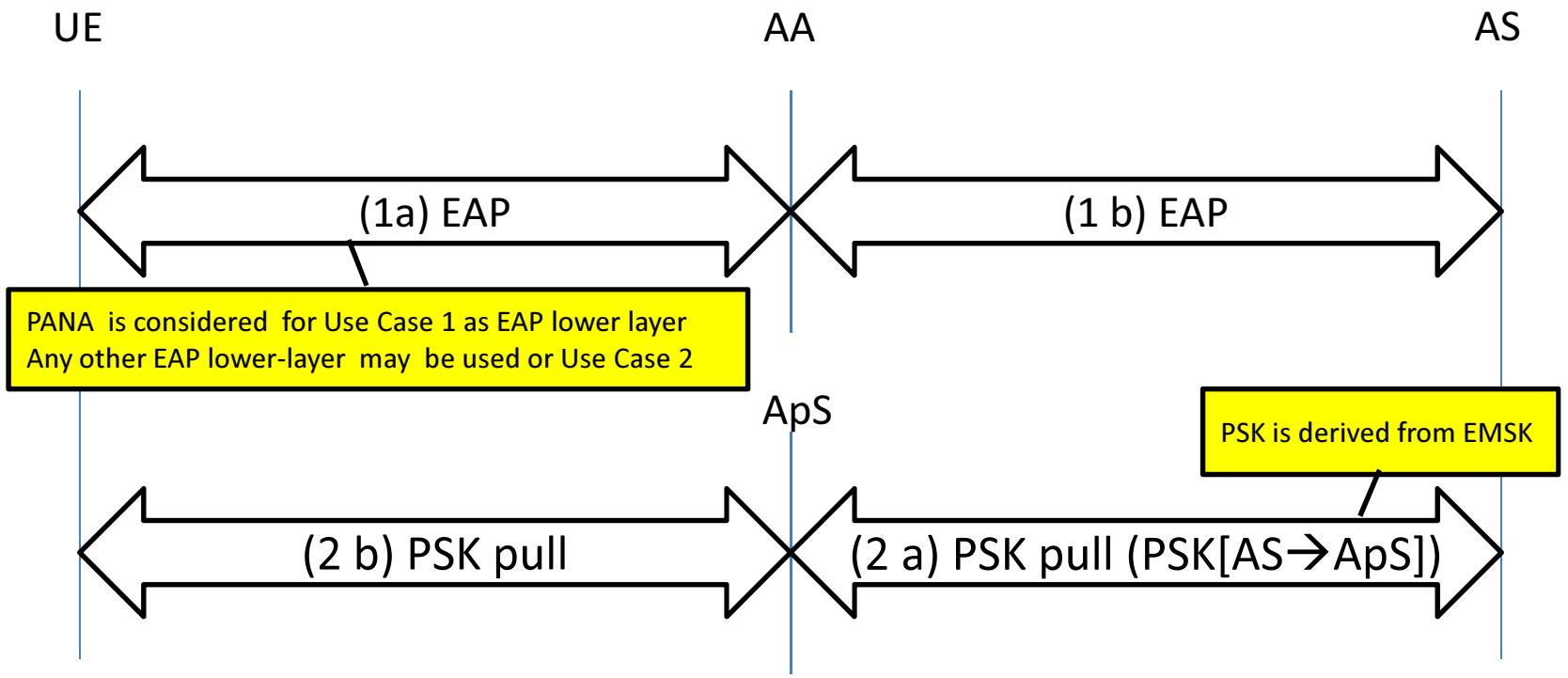
Considerations/Assumptions

- Solution should have the capability to support integration of network access authentication and application access authentication

- Configured parameters through the auth & key establishment process
  - Identity of CoAP client
  - Identity of CoAP server
  - PSK for DTLS or IKEv2

- EAP is supported for application service access authentication protocol
  - EAP invariants: {mode, media, method}-independence

Call Flow


UE
AA
AS
(1a) EAP
(1 b) EAP
PANA is considered for Use Case 1 as EAP lower layer
Any other EAP lower-layer may be used or Use Case 2
ApS
PSK is derived from EMSK
(2 b) PSK pull
(2 a) PSK pull (PSK[AS→ApS])

Support for Recommissioning

- Change of Service provider (i.e., Recommissioning) can be supported by the proposed architecture
  - by using service provider-independent credentials
    - Such credentials can be used for securely configuring application service provider-specific credentials

Summary

- We believe the proposed framework can cover a variety of deployment scenarios

- Any thoughts/comments?