ACE/CORE requirements on COSE

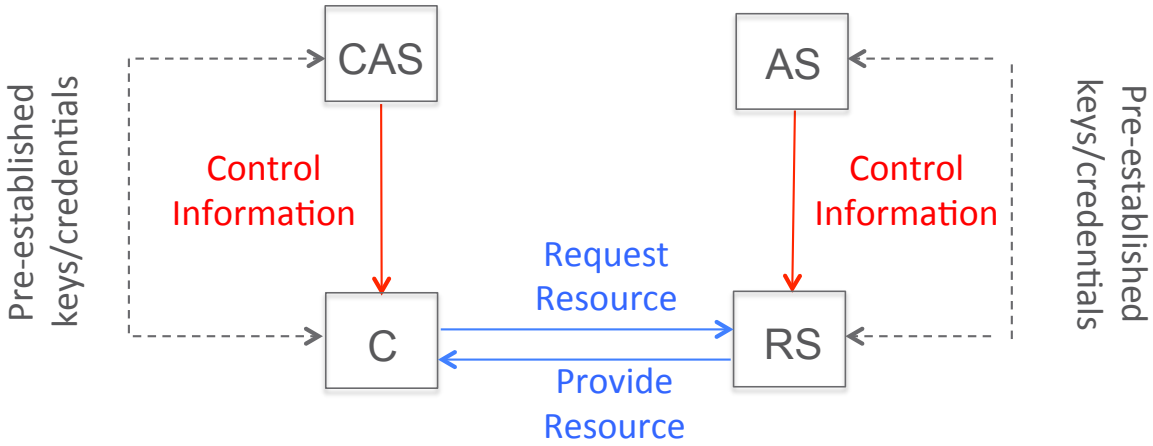
Göran Selander, Ericsson

IETF 93 COSE WG, Prague, July 20, 2015

ACE Architecture and Information Flows

Legend:

- Black boxes represent functions
  - Functions may be combined in one node
- Information flows in solid lines
  - Resource access (based on CoAP)
  - Control information (authorization information, keys, etc.)
  - Information flow may pass intermediary nodes

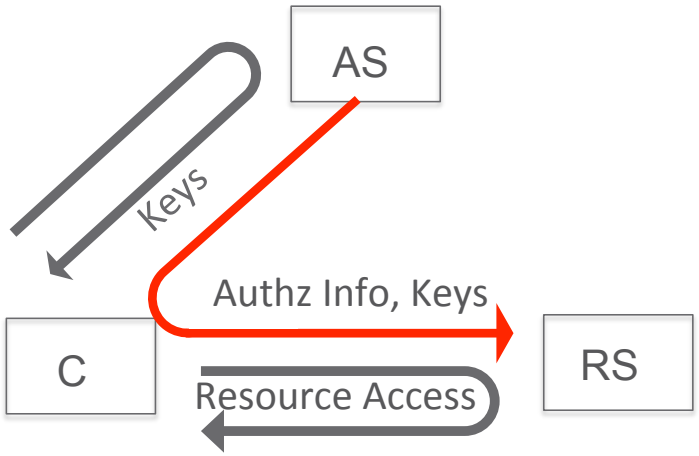


Information flows may be protected with session-based security (DTLS) or data object based security (COSE)

Source: draft-gerdes-ace-actors

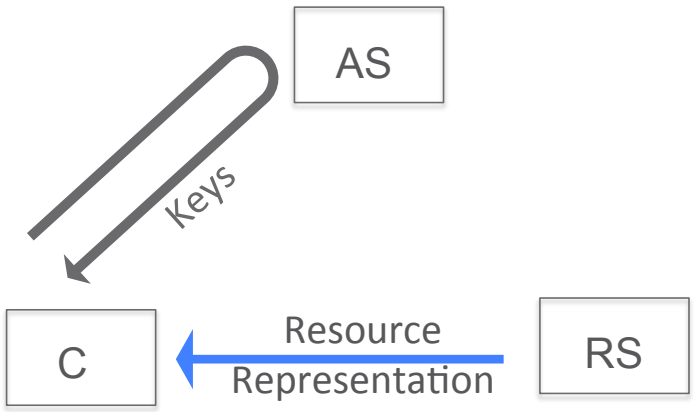
Authorization Scheme Examples

Push Scheme



Control information as secure object

Client Pull Scheme



Resource access as secure object

Source: draft-seitz-ace-core-authz

Object Secure CoAP (OSCOAP)

- draft-selander-ace-object-security-02
- Wrapping CoAP message in a “Secure Message” (SM) format
- Header consists of
  - “alg”: Ciphersuite
  - “cid”: Context Identifier (used by receiver to identify security context)
  - “seq”: Sequence Number (for replay protection)
- COSE it the candidate realization of Secure Message format
  - Appendix D: COSE profile of Secure Message
  - Appendix E: Message size estimates (later slides)
  - Based on draft-schaad-cose-msg-00
- OSCOAP comes in two modes – **Mode:COAP** and **Mode:PAYL** (next slides)

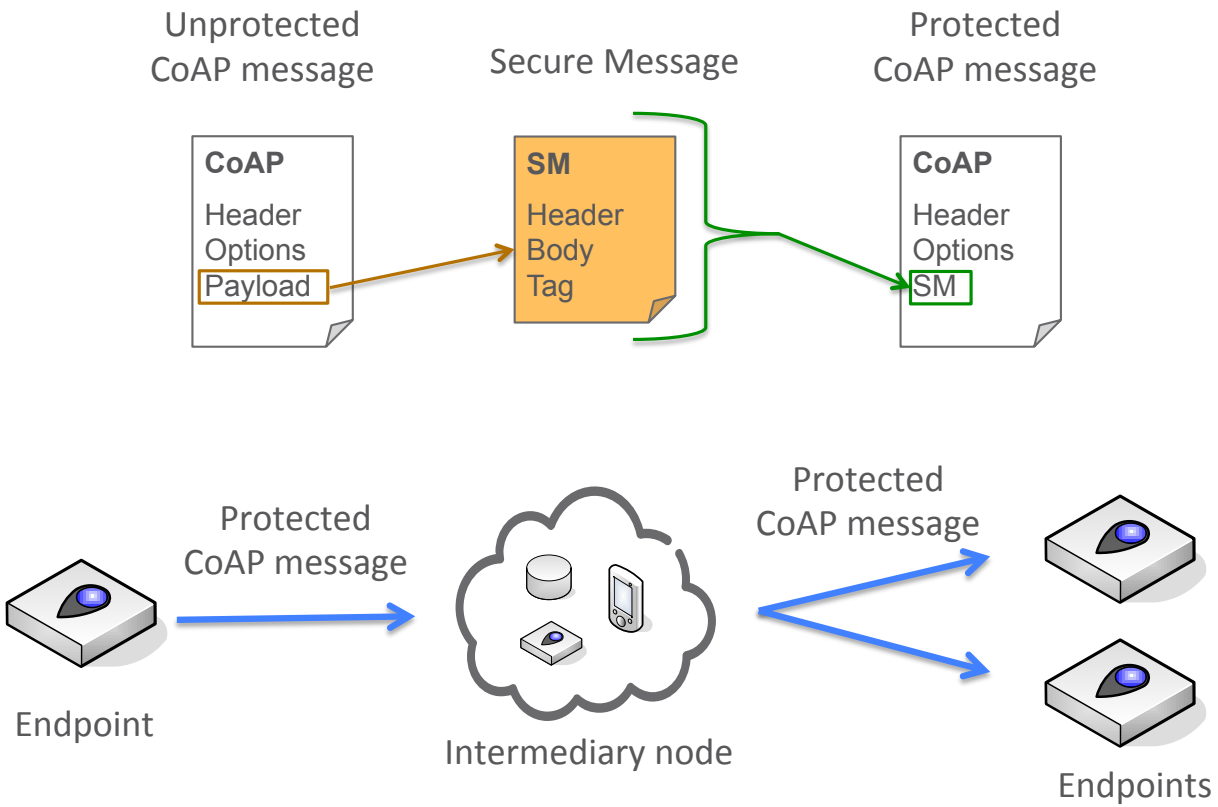
Transaction Identifier (TID)

OSCOAP Mode:PAYL

- Wrap CoAP Payload as Secure Message (SM)
- Replace CoAP Payload with SM

Properties:

- Protects CoAP Payload only
- Provides e2e confidentiality, integrity and replay protection
- Supports point-to-point and point-to-multipoint

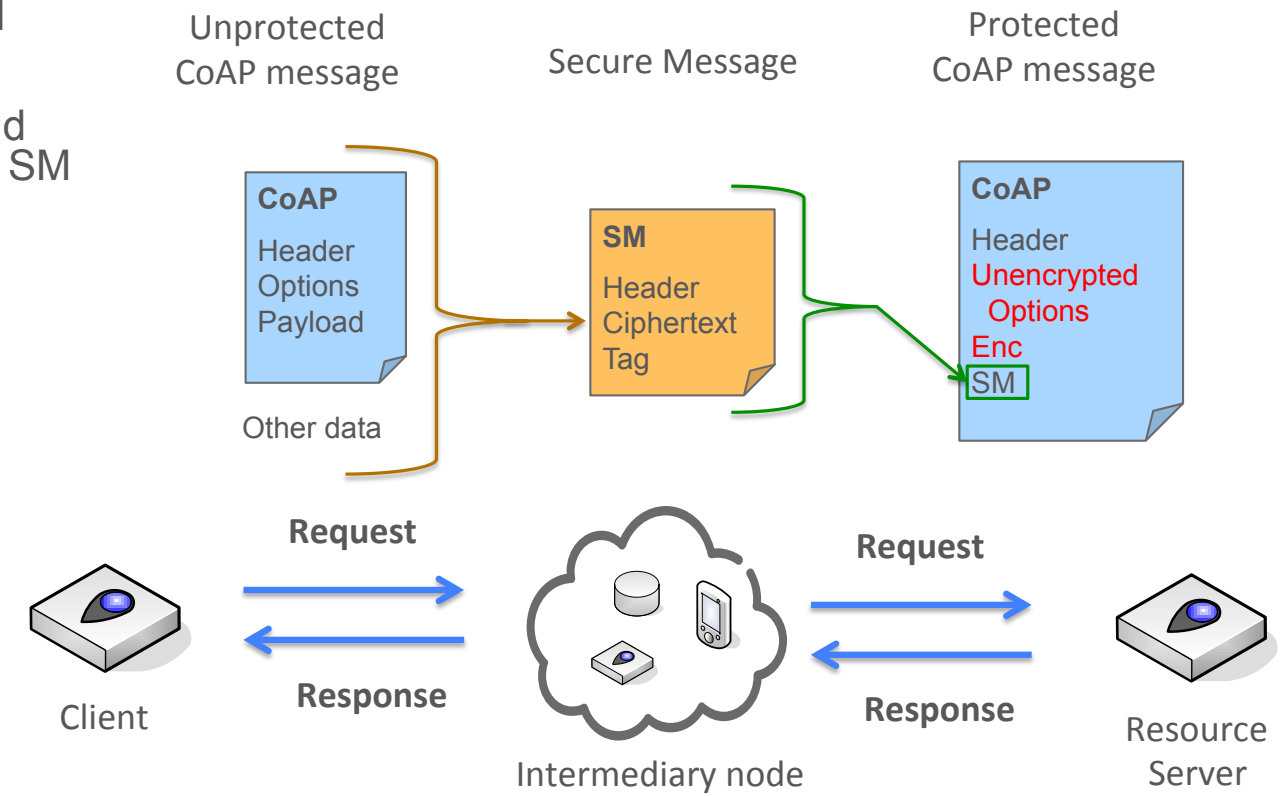


OSCOAP Mode:COAP

- Wrap CoAP message as a SM with detached non-ciphertext
- Replace encrypted Options and Payload with Enc option*) and SM

Properties:

- 2-pass point-to-point
- Protects CoAP message**)
- Provides e2e confidentiality, integrity and replay protection
- Challenge-response
- Secures messages without payload



*) Sig option similar

**) Payload, selected Header Field and Options and other data (Request TID) if response message

Additional required support

- Types of cipher suites
  - MAC, Signature, AEAD and symmetric key encryption + signature
    - The latter for closed group distribution with data origin authentication
  - Key establishment is not settled, some variant of ECDH will be needed
- Content (Mode:PAYL) and detached content (Mode:COAP)
  - Integrity protected data carried in CoAP message (but not in SM) as well as out of band

- Reduce message size. Appendix D.3 proposes optimizations:
  1. Make 'recipient' field optional
  2. Define key/label values for the introduced parameters and algorithms
  3. Allow single wrapping for symmetric key encryption + signature
  4. Allow signatures schemes with message recovery
     - Needs to be cleared with CFRG

Integrity protection only

› MAC
  – HMAC-SHA256
  – truncated to 16 bytes, where possible

Scheme	Header	MAC	Total Overhead
JWS	74 B	43 B	119 bytes
COSE	37 B	16 B	53 bytes
mod-COSE	24 B	16 B	40 bytes
CSM	13 B	16 B	29 bytes

Opt 1 & 2

› Signature
  – ECDSA with 64 byte signature

Scheme	Header	Tag	Total Overhead
JWS	74 B	86 B	162 bytes
COSE	36 B	64 B	100 bytes
mod-COSE	30 B	64 B	94 bytes
CSM	13 B	64 B	77 bytes

Opt 4 would lower this

Note: 8-byte “cid” and 3-byte “seq” is assumed throughout.

Authenticated Encryption

› AES-GCM:

Scheme	Header	IV	Tag	Total Overhead
JWE	72 B	16 B	22 B	113 bytes
COSE	31 B	12 B	16 B	59 bytes

› AES-CCM:

Scheme	Header	Tag	Total Overhead	
COSE	44 B	16 B	60 bytes	
mod-COSE	31 B	8 B	39 bytes	Opt. 1 & 2
CSM	13 B	8 B	21 bytes	

Symmetric encryption + digital signature

› AES-GCM + ECDSA

Scheme	Header	Sig	Payload	Total Overhead
JWS	74 B	86 B	113 B	275 bytes
COSE	37 B	64 B	59 B	160 bytes

› AES-CCM + ECDSA

Scheme	Header	Sig	Payload	Total Overhead
COSE	36 B	64 B	52 B	153 bytes
mod-COSE	30 B	64 B	39 B	134 bytes

Overhead based on nested objects

Opt 1 & 2

Opt 4 would lower this

› AES + ECDSA

CSM	13 B	64 B	0 B	77 bytes

Opt 3

Thank you!