COSE Message Draft

Jim Schaad

August Cellars

Current State

- Version -01 is published
  - Document structure laid out
  - Currently using CDDL for grammar
  - Numerous open issues to be discussed
- Version -02 on github
  - Includes a strawman on algorithms

Past Discussions

- Make MAC top level distinct from Signed top level
- Use tagging and msg_type for identification purposes
- Flattened encoding
- Make key management more uniform
- Include authentication tag with the cipher text

Open Discussions

- Use Array or Map at the top level
- Change to using integers and strings for map labels

Future Discussions

- When to use binary vs base64 in CBOR encodings
- CBOR encodings for building octet strings to cryptographically process
- Keep compression
- When to use Strict Mode Encodings
- Separation of attributes by layer
- Allow for authenticated data at a recipient layer
- MAC Key management
- KID Attributes – binary, text, both

Future Discussions

- Deprecation of some headers
- Requirements of presence, absence and location of attributes
- Status of CDDL in the document.
- Mandatory to Implement Algorithms
- What to do with the media-type table ****
- X.509 support
- URLs pointing to keys

Top Level Structure/Single Recipient or Signer

- Current State
- Implementation Experience
- Simplicity
- Options:
  - Map + array of recipients
  - Map + one recipient or array of recipient
  - Array + array of recipients
  - Array + unroll recipient or array of recipients
- Message type tagging if not first

Algorithm Discussion

- Document Count: One, Two, Many
- Mandatory to implement set
  - Current approach defers to the application to define
  - IESG might require, but applications are going to finesse around in any event
- Registration set of algorithms

What can a device do?

- Very low end
  - AES in hardware, SHA in hardware if lucky
  - Low bandwidth, power, memory, “slow” CPU
- Middle
  - AES and SHA in hardware
  - May have EC in hardware
- High End
  - Can do anything.

AEAD Content Encryption Algorithms

- AES Modes
  - GCM
  - CCM
  - OCB
- Cha-Cha + PolyCrypt 1305

MAC Algorithms

- HMAC
  - SHA-2 (256, 512)
  - Key size requirements
- AES modes
  - MAC
  - CMAC
- SHA-3

Key Management Methods

- Define based on services required
- Need to look at what known frailties are:
  - Long term direct use of shared secrets
  - IV re-use and other attacks from block ciphers
- Need to look at what hardware can support