Encrypted Payloads in SUIT Manifests

draft-ietf-suit-firmware-encryption-11

Hannes Tschofenig, Russ Housley, Brendan Moran,
David Brown, Ken Takayama

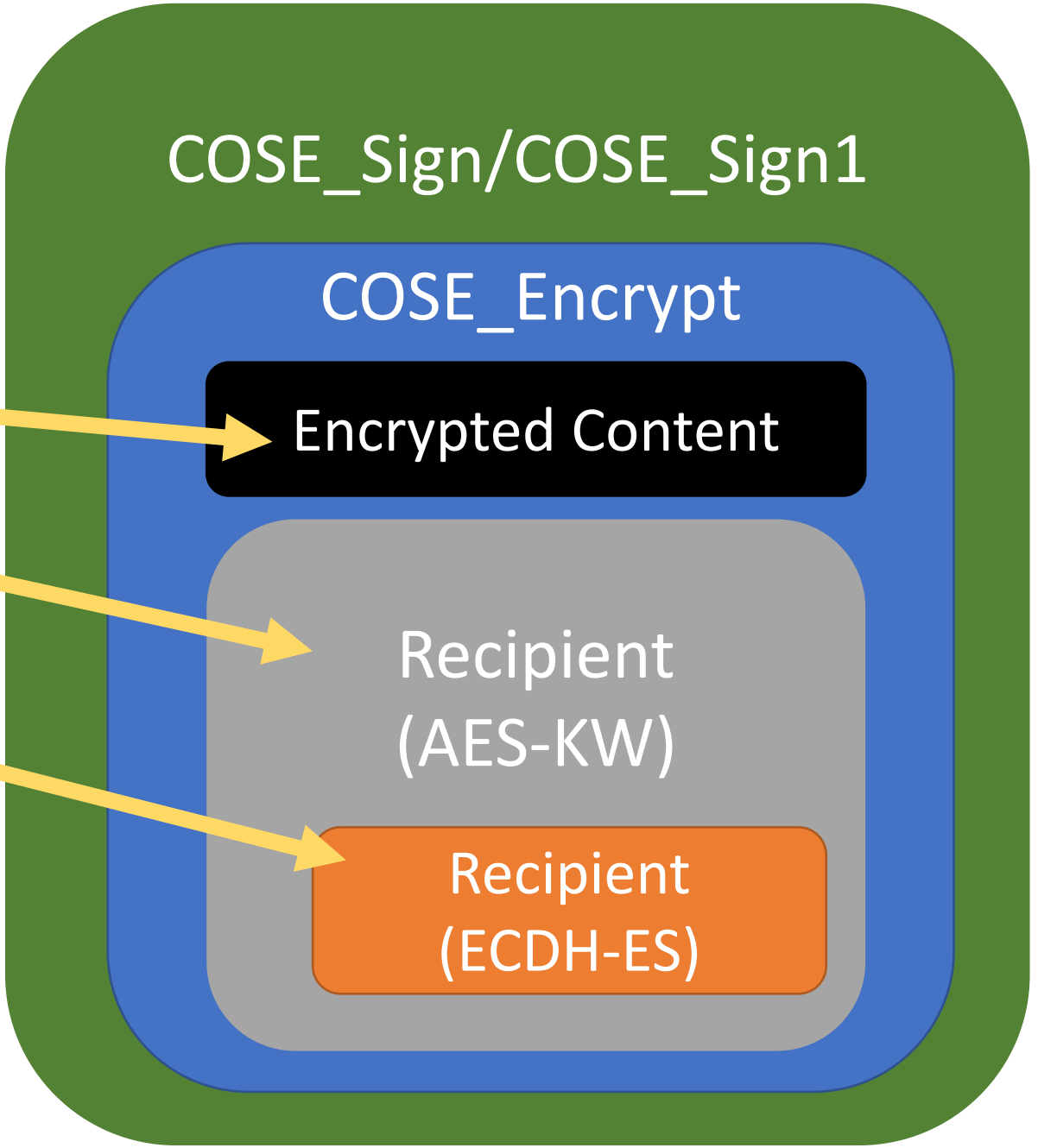
Progress Report

- draft-ietf-suit-firmware-encryption-11 fixed editorial glitches in -10.
- -10 replaced HPKE with ECDH-ES.
  - Editorial reshuffling to make the resulting document more readable.
- Why?
  - COSE-HPKE made slow progress and there are still a number of open issues.
  - ECDH-ES is already full specified in the COSE RFC, and it meets all of the SUIT requirements.
- Next slides describe how it works.

COSE_Encrypt with ECDH-ES

Three-layer structure consisting of

- Content encrypted with CEK.
- Recipient structure utilizing AES-KW to encrypt CEK with a KEK.
- KEK produced by ECDH-ES in embedded recipient structure.



ECDH-ES Recipient Structure

```
/ recipients-inner /
[
  / protected / h'a1013818' / {
     \ alg \ 1:-25 \ ECDH-ES + HKDF-256 \
   } / ,
  / unprotected / {
    / ephemeral / -1:{
      / kty / 1:2,
      / crv / -1:1,
      / x / -2:h'b2add44368ea6d641f9ca9af308b4079
              aeb519f11e9b8a55a600b21233e86e68',
      / y / -3:false
    },
    / kid / 4:'kid-1'
  },
  / ciphertext / h''
]
```

- Uses regular `COSE_recipient` with `COSE_Key` structures.
- With the Key ID the sender informs the recipient what public key was used for encryption.
  - Maybe be known from the context.
- Ciphertext field is empty.
  - ECDH-derived symmetric key is input to key derivation function (HKDF).
  - Result is the KEK, which is used a layer below.

Next Steps

- Add missing text about info and aad input.
- Implementation feedback desired.