Firmware Encryption

draft-ietf-suit-firmware-encryption

(Russ H., Brendan M., Hannes T.)

Changes since last IETF meeting

- Updated draft to reflect changes to the COSE-HPKE draft.
  - Two layer structure simplifies payload.
  - Fixed references.
- Focused on COSE-HPKE

Reference Implementation

- HPKE for Mbed TLS: https://github.com/ARMmbed/mbedtls/pull/5078
- COSE-HPKE for t_cose: https://github.com/laurencelundblade/t_cose/pull/46
  - Updated code based on hackathon will be uploaded soon
- Firmware encryption with AES-KW not yet available because AES-KW code is not yet incorporated into the PSA Crypto API (see https://github.com/ARMmbed/mbed-crypto/pull/364/)

Open Issues

- Ensure that COSE-HPKE stable
- Offer complete examples
- Binding context to encryption

COSE_Encrypt

```
96_0([
      / protected header with alg=AES-GCM-128 /
      h'a10101',
      / unprotected header with nonce /
      {5: h'938b528516193cc7123ff037809f4c2a'},
      / detached ciphertext /
      null,
      / recipient structure /
      [
          / protected field with alg for HPKE /
          h'a1013863',
          / unprotected header /
          {
              / ephemeral public key with x / y coordinate /
              -1: h'a40102200121..90c675df4162c39',
               /  kid for recipient static ECDH public key /
               4: h'6b69642d32',
          },
          / encrypted CEK /
          h'9aba6fa44e…b31a3b9d37c7',
      ],
  ])
```

COSE_Sign1

```
18(
   [
     / protected / h'a10126' / {
              \ alg \ 1:-7 \ ECDSA 256 \
           } / ,
     / unprotected / {
           / kid / 4:’alice@example.com'
        },
     / payload /     h'AA19...B80C',
     / signature /   h'E3B8...25B8'
   ]
)
```

Additional Authenticated Data (AAD) for COSE_Encrypt

```
Enc_structure = [
  context : "Encrypt",
  protected : empty_or_serialized_map,
  external_aad : bstr
]
```

- external_aad is set to null.
- Protected structure refers to the protected field in the COSE_Encrypt, which contains the algorithm, e.g. AES-GCM-128.

Context Information Structure

- The context information structure is used to ensure that the derived keying material is "bound" to the context of the transaction.
- Not used in AES-KW.
- For HPKE, a structure is defined in COSE-HPKE draft.
  - It includes the algorithm information, key length, and identifiers of the sender and the recipient.
  - The identities are not yet included in the HPKE KDF.
  - For HPKE:
    - The recipient identity is the kid in the recipient layer.
    - The sender identity is found in the COSE Sign layer – not in the COSE Encrypt layer.