COSE HPKE

March 2022

Hannes Tschofenig, Russ Housley, Brendan Moran
Status Update

• Following the interim meeting, [draft-ietf-cose-hpke-01](https://github.com/cose-wg/HPKE/) was published with the discussed changes.

• Created initial implementation based on -01 version (utilizing the PSA Crypto API): [https://github.com/laurencelundblade/t_cose/pull/46](https://github.com/laurencelundblade/t_cose/pull/46)
  • Implementation contains COSE_Encrypt0 (with direct key agreement) and COSE_Encrypt (with HPKE).
  • Worked further improvements during the hackathon to utilize crypto adaptation layer in t_cose.

• Draft repository: [https://github.com/cose-wg/HPKE/](https://github.com/cose-wg/HPKE/)
New Structure
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'a401022001215820a596f2ca8d159c04942308ca90
cfbcfa65b108ca127df8fe191a063d00d7c517225820aef47a45d6d6c572e7bd1b9f3e69b50ad3875c68f6da0caaa90c675df4162c39',
            / kid for recipient static ECDH public key /
            4: h'6b69642d32',
        },
        / encrypted CEK /
        h'9aba6fa44e9b2cef9d646614dcd670dadb31a3b9d37c7a65b099a8152533062',
    ]
]
Open Issues
HPKE Algorithm Registry in COSE
Re-use HPKE algorithms

Approaches:

• Add “rule” to HPKE spec to automatically populate COSE registry
• Add “rule” to COSE-HPKE spec to automatically create COSE HPKE registry entries whenever new HPKE algorithms are added
  • Has to be discussed with IANA
• Require COSE HPKE algorithm registrations whenever new HPKE algorithms are created.

• Currently defines COSE HPKE algorithm as a combination of AEAD ID, KDF ID and KEM ID.
Compressed Points

- Point compression is a 20+ year old technology.
- Helps to reduce the over-the-wire size of the COSE structure.
- Proposal is to add it as an optional feature because support for it is not widely available today.
HPKE Configuration

• For configuration purposes it may be useful to define COSE_Key structure similar to the approach taken in Encrypted Client Hello (draft-ietf-tls-esni) and PEM file format for ECH (draft-farrell-tls-pemesni)

• Need to create a proposal first.
Info Structure

• COSE defines an info structure, which is used as an additional data structure for use with various algorithms.

• Draft also defines an info structure, which is optional to use.

• Proposal is to remove it because
  • HPKE itself defines ways to add extra info into the key derivation function, and
  • HPKE incorporates various fields in the key derivation process already.
New Use Case

• Mail to list: https://mailarchive.ietf.org/arch/msg/cose/9nowDz5kbfUvrGR-o6U1Tm31XAA/

• Richard Barnes and Chris Wood suggested to also support the use of HPKE in COSE without the 2-layer structure.
  • This supports those use cases where plaintext is directly encrypted with HPKE (rather than via a layer of indirection with the HPKE(CEK), {Plaintext}CEK combination)

• Feedback?
Next steps

• New draft version by mid April
• Updated reference implementation