KEMS IN CMS

COMPOSITE KEYS, SIGS, ENCRYPTION

Mike Ounsworth, John Gray, Serge Mister, Julien Pret, Ludovic Perret

LAMPS 112

ENTRUST

SECURING A WORLD IN MOTION

Outline

- CMS KEM Recipient Info (L. Parret’s draft)
- Composite drafts:
  - Public keys / Certificates
  - Composite Signatures
  - Composite Encryption

CMS KEM RecipientInfo

- A draft has been started by Ludovic Perret, Julien Prat, and myself to provide a generic KEM-based RecipientInfo in CMS (generalizing RSA-KEM RFC 5990).
- Draft not published yet.
- Several ways to approach:
  1. (current) Use KeyTransRecipientInfo with an AlgID OID indicating it's actually a wrapped KEM, and AlgID Params containing AlgIDs of {KEM, KDF, WRAP}.
  2. Use OtherRecipientInfo with content similar to (1).
  3. Define a new top-level KEMRecipientInfo
  - ◦ Question: Is this worth a discussion on-list, or are they all sorta equivalent?

Core idea:

Params: KEM, KDF, WRAP
Input: recipPubKey, cek

ss, ct = KEM.encaps(rPK)
kek = KDF(ss)
wk = WRAP(kek, cek)
ek = ct || wk

Composite / dual / hybrid landscape

Keys / Certs

draft-ounsworth-pq-composite-keys-00
draft-ounsworth-pq-explicit-composite-keys-00

- Defines composite public and private keys
- Could go with either a generic (open container) or explicit (pairwise OIDs) approach

Non-composite multi-cert

- Alison Becker's proposal
- *(which I have not seen at time of writing slides)*

Signatures

draft-ounsworth-pq-composite-sigs-05

- Defines composite dual signatures
- Stable, mature draft.
- Currently references the composite keys draft, but could easily be made to work with multi-cert instead or in addition.

Encryption

draft-ounsworth-pq-composite-encryption-00

- Defines composite hybrid encryption for use with CMS EnvelopedData
- Still undergoing heavy design iteration.
- Currently references the composite keys draft, but could easily be made to work with multi-cert instead or in addition.

Composite Keys

draft-ounsworth-pq-composite-keys-00
draft-ounsworth-pq-explicit-composite-keys-00

- We heard feedback at the Sept 13 interim LAMPS mtg that explicit is preferred.
  - That is, providing an ASN.1 “factory” for producing and using pre-defined pairs of algs.
- Still working on Explicit Composite ASN.1.
- Plan to re-work to make Generic a sub-type of Explicit
  - ie register an OID for “pk-AnyWithAny”

Security properties of composite keys (for comparison against a multi-cert approach):

- Strongly binds multiple keys to same identity.
- Can enforce strong multi-key binding to the root CA.
- Allows certificate issuer to control whether sub-keys must be used in AND or OR mode.

Composite Signatures

draft-ounsworth-pq-composite-sigs-05

- Mature draft, no change since last time.
  - Some design decisions that we'll bring up if / when this gets WG Adoption.
- Working on Explicit Composite ASN.1 for defining SigAlgs of pre-defined pairs.

- Regardless of how pub keys are conveyed (composite vs multi-cert), you'll need a mechanism for producing a multi-key signature.
- This draft can easily work with composite or multi-cert.

Composite Encryption / KEM

draft-ounsworth-pq-composite-encryption-00

- Goal: composite hybrid encryption for use with CMS EnvelopedData
  - IE given a recipient with multiple KEM, KeyEx, and/or Encr public keys, produce an EnvelopedData that requires all their private keys to open it.
- Still undergoing heavy design iteration .
- Debate over what interface it should expose:
  - KeyTrans: Take a CEK and a recipient (composite / multi) pub key, and produces an enciphered CEK. This would fit directly into KeyTransRecipientInfo.
  - KEM: Take a recipient (composite / multi) pub key, and produce a shared secret and an enciphered shared secret. This would fit into Ludovic’s new KEM RecipientInfo draft.
- Debate over underlying mechanism:
  1. Establish a shared secret under each algorithm, use these (via a KDF?) as one-time-pad XOR keys to wrap the CEK.
     - Advantage: fewer parameters to go stale over time. Is a KeyTrans.
  2. Establish a shared secret under each algorithm, roll these through a KDF to produce an AES key; AES-wrap the CEK.
     - Advantage: more standard, follows NIST SP 800-56C-r2. Could be either KeyTrans or KEM.