BPSec COSE Context

IETF 115 DTN WG

Brian Sipos
JHU/APL
Background

• BPSec and its Default Security Context are usable but intentionally limited in scope:
  - A limited number of symmetric-keyed encryption and MAC algorithms
  - Defines a variable additional authenticated data (AAD) binding to the block/bundle
  - No explicit key identifiers are available

• For internet-facing nodes, possibly as subnetwork gateways, there is a need for PKI-integrated security
  - This was indicated by IETF SECDIR review of BPSec draft and also discussed as a near-future need by NASA DTN planning group

• Don’t want to reinvent the wheel, and CBOR Object Signing and Encryption (COSE) already provides syntax and semantics for current and future PKI security
  - Even COSE (with a restricted profile as used here) still provides a lot of variability, in the same sense that TLS or S/MIME does, which must be managed out-of-band (e.g. don’t use ECC algorithms if security acceptors can’t support it)
Goals for the BPSec COSE Context

• Do not alter BPSec structures or requirements
  - This is purely an extension within the existing security context mechanism

• Handle current symmetric-keyed and PKI algorithms
  - Leverage existing algorithm definitions

• Follow algorithm-use and key-use best practices
  - Avoid key overuse, use random content encryption keys
  - Allow Diffie-Hellman static-ephemeral algorithms to be used (both Elliptic and Edwards curves)

• Add as little encoded overhead as possible

• Inherit future gains made by COSE off-the-shelf algorithms
  - Allow using CWT as a future alternative to X.509 (PKIX) for node identity allocation
  - Planning is already underway for hybrid public key encryption (HPKE) and post-quantum cryptography (PQC)
Proposed COSE Context Contents

• One BPSec context codepoint defined to use in BIB and BCB

• Parameter and result types defined for each BPSec block type:
  - AAD scope parameter (same semantics as RFC 9173 for consistency)
  - De-duplicated last-layer COSE header parameters
  - Integrity results (COSE MAC and Signature messages)
  - Confidentiality results (COSE Encrypt messages)

• Public key identifiers in parameters to de-duplicate data
  - Keys/certificates/CWT can be transported in-parameter or externally
  - Potential future extensions could provide additional supporting data (e.g. OCSP stapling)

• Full COSE messages contained in each target’s result
  - Reuse COSE message tags as result type codes
  - Allows an application to use any current or future COSE algorithm types (and combinations)
  - Allows multiple recipients for a single security block (both BIB and BCB)
  - Interoperability requirements are defined in a COSE Profile (next slide)
Interoperability Profile

- Required algorithms for AES-GCM-256, AES key-wrap, and HMAC-SHA2-256
- Recommended algorithms for Elliptic Curve, Edwards Curve, and RSA signing and key-wrap/key-generation
- Additional public key material can be included in an “additional header map”, applying to all results in the block

<table>
<thead>
<tr>
<th>BPSeq Block</th>
<th>COSE Layer</th>
<th>Name</th>
<th>Code</th>
<th>Implementation Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrity</td>
<td>1</td>
<td>HMAC 256/256</td>
<td>5</td>
<td>Required</td>
</tr>
<tr>
<td>Integrity</td>
<td>1</td>
<td>ES256</td>
<td>-7</td>
<td>Recommended</td>
</tr>
<tr>
<td>Integrity</td>
<td>1</td>
<td>EdDSA</td>
<td>-8</td>
<td>Recommended</td>
</tr>
<tr>
<td>Integrity</td>
<td>1</td>
<td>PS256</td>
<td>-37</td>
<td>Recommended</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>1</td>
<td>A256GCM</td>
<td>3</td>
<td>Required</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>2</td>
<td>A256KW</td>
<td>-5</td>
<td>Required</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>2</td>
<td>ECDH-ES + A256KW</td>
<td>-31</td>
<td>Recommended</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>2</td>
<td>ECDH-SS + A256KW</td>
<td>-34</td>
<td>Recommended</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>2</td>
<td>RSAES-OAEP w/ SHA-256</td>
<td>-41</td>
<td>Recommended</td>
</tr>
</tbody>
</table>

Table 5: Interoperability Algorithms
Next Steps

• This is not intended to replace or supersede existing BPSec interoperability contexts in RFC 9173

• The point of this security context is to allow BPSec in a PKIX environment in the very near term
  - COSE is a known quantity with existing coding and processing tools
  - Identifying bundle security purpose and validation of a Node ID within a PKIX certificate are already defined in RFC 9174
  - An extension to ACME to automate validation of a Node ID is under review

• Some secondary questions remain, for example:
  - How does a security acceptor handle a BIB signed by a key with a certificate for a different Node ID than the security source? Base BPSec doesn’t really deal with identity/authentication logic
  - Is there a more strict minimum COSE header content? S/MIME makes requirements about full certificate presence, while the current draft allows an “x5t” thumbprint as a placeholder for compact encoding