Persistent Symmetric Keys

Daniel Huigens
What & Why?

- Encrypt messages using long-term symmetric keys, e.g. for archival
- Create attestations using long-term symmetric keys, e.g. to record signature verifications
- Faster & smaller than asymmetric crypto, & post-quantum secure
How?

- The semantics we need are those of PKESK and signature packets
- So: retcon “Public Key Algorithms” to “Persistent Key Algorithms”
- Define two new Persistent Key Algorithms: AEAD and HMAC
## New Public Persistent Key Algorithms

<table>
<thead>
<tr>
<th>ID</th>
<th>Alg.</th>
<th>Public Key</th>
<th>Secret Key</th>
<th>Signature</th>
<th>PKESK</th>
</tr>
</thead>
<tbody>
<tr>
<td>128?</td>
<td>AEAD</td>
<td>sym. algo, hash(seed)</td>
<td>seed, key material</td>
<td>N/A</td>
<td>AEAD algo, IV, length, ciphertext</td>
</tr>
<tr>
<td>129?</td>
<td>HMAC</td>
<td>hash algo, hash(seed)</td>
<td>seed, key material</td>
<td>authentication tag</td>
<td>N/A</td>
</tr>
</tbody>
</table>
### New Public Persistent Key Algorithms

<table>
<thead>
<tr>
<th>ID</th>
<th>Alg.</th>
<th>Public Key</th>
<th>Secret Key</th>
<th>Signature</th>
<th>PKESK</th>
</tr>
</thead>
<tbody>
<tr>
<td>128?</td>
<td>AEAD</td>
<td>sym. algo, hash(seed)</td>
<td>seed, key material</td>
<td>N/A</td>
<td>AEAD algo, IV, length, ciphertext</td>
</tr>
<tr>
<td>129?</td>
<td>HMAC</td>
<td>hash algo, hash(seed)</td>
<td>seed, key material</td>
<td>authentication tag</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Current Status

- Personal draft: draft-huigens-openpgp-persistent-symmetric-keys
- https://twisstle.gitlab.io/openpgp-persistent-symmetric-keys/
- Experimental implementations in forks/branches of OpenPGP.js and go-crypto
Questions for the WG

- Feedback?
- Call for adoption soon?
Thanks!
Questions?