Encrypted Header Extensions in SRTP

Jonathan Lennox
jonathan@vidyo.com

Encrypted Header Extensions in SRTP

draft-ietf-avtcore-srtp-encrypted-header-ext-00

- Now a working group item.
- No technical changes from most recent individual submission.
- Some discussions of open issues (on subsequent slides…)

Resolved issue

- The mechanism in this draft does “partial” encryption
  - Standard RFC 5285 framing of header extension elements
  - Encryption of element bodies only
- This has advantages:
  - Backward-compatibility (existing implementations will ignore an unknown extension element)
  - Ability to send some elements in the clear, for middleboxes (e.g. transmission-time offset) while encrypting others
  - Ability to offer encrypted and unencrypted versions of an element in offer/answer

Limitations of partial encryption

- Presence/absence of a header extension element is visible in the packet.
- With the two-byte form of header extensions, partial encryption doesn't encrypt zero-length "flag" extension elements, or the four "appbits" of the 0x100x field.
  - Neither of these features exist in the one-byte form.
  - I don't think this will be a problem in practice
    - Please let me know if anyone is using either of these!
  - Can re-write information to be carried in the body of an extension element, which can be encrypted.

Open Issue: Session-level signaling

- Header Extensions can be negotiated at session level in SDP.
- We need to define whether this is permitted for encrypted extension headers.
- Proposal: MUST NOT if any media lines are not SRTP; MAY if all media lines are SRTP.

Open issue: AEAD Encryption

- Mechanism in this draft is based on CTR transforms
  - All currently registered SRTP algorithms are CTR
- Not clear how to use this with Authenticated Encryption with Associated Data (AEAD) transforms: AES_GCM, AES_CCM
  - draft-ietf-avt-srtp-aes-gcm
- Need someone who understands these transforms.
- Or, defer if that work is stalled.

Remaining work

- Add text clarifying offer/answer and backward compatibility.
- Resolve open issues.
- Add test vectors.
- Done?