XMPP E2E

IETF 83
Matthew Miller
Current Status

- New ideas, new document
- Using work from JOE S WOES JOSE
Big Ideas

- Encryption for multiple end-points
- Request content keys when needed
Discovering Support

• CAPS (XEP-0115)
• supported if any resource announced e2e
Encrypting ...

- Start with stanza
- Wrap with <forwarded/>
- Serialize to UTF-8
• Generate block cipher factors
• encData == BlockCipher(cek, fwdStr)
• Package as partial JWE
  - no public key use (yet)
... Encrypted!

- Package into container
  - stanza with matching kind + type + addressing
  - `<e2e/>` child
    - ✓ ‘id’ to associate CEK
    - ✓ `<header/>` for JWE header
    - ✓ `<data/>` for encrypted data
Decrypting ...

- if key is known ...
  - continue

- if key is *NOT* known ...
  - GOTO “Keyreq”
... Still Decrypting ...

- `fwdStr == BlockCipher(cek, encData)`
- stanza parsed and unwrapped from UTF8
... Decrypted!

- Validated via JOSE (AEAD | MAC)
- Timestamp from <forwarded/>
- Others?
Making a Keyreq

- `<iq type='get'/>` to sender
  - ‘id’ for CEK
  - PK(s)
- sender accepts/rejects
Accepting a Keyreq

- Encrypt CEK using provided PK
- `<iq type='set'/>` to requester
  - `<header/>` with JWE header (key info)
  - `<cek/>` with encrypted CEK
Denying a Keyreq

- Requester does not match bare JID of recipient (or sender)?
- Certificate does not validate?
- Other ... ?
Open Issues

• Optimize for known PKs
• CEK usage
• No offline
• Signing ...
Side Benefits

- PK operations spread out
- Compatible with MUC (maybe)
Caveat Emptor

• Trust issues?
• Potential for keyreq floods
• Stanza info not completely protected
References

JSON Web Encryption (JWE)

<draft-ietf-jose-json-web-encryption>

XEP-115: Entity Capabilities


XEP-0297: Message Forwarding