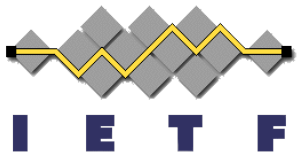
Diameter End-to-End Security: Keyed Message Digests, Digital Signatures, and Encryption

draft-korhonen-dime-e2e-security-00

Jouni Korhonen, Hannes Tschofenig

Dime WG, IETF#83



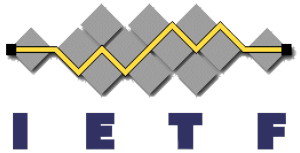
Overview

- Two aspects:
  - Authentication and Key Exchange
  - Actual AVP protection.
- Requirements.
- Strawman solutions proposal.

Requirements

- Provide end-to-end security properties to Diameter on top of existing hop-by-hop security model.
- Works with existing request routing and through agents that might modify parts of the message (like rearrange AVPs).
- Decouple key management from actual e2e AVP security protection.
- Offer both integrity and confidentiality protection.
- Easy to integrate into existing Diameter applications (integrity protection).

Strawman Proposal in
draft-korhonen-dime-e2e-security-00



- This solution focuses on protecting Diameter AVPs. To offer the functionality two AVPs are defined:
  - Signed-Data (octet string) for integrity protection of one or more AVPs.
  - Encrypted-Data (octet string) for confidentiality protection of one or more AVPs.

- Re-use existing security mechanisms. A few choices available including:
  - CMS
  - XML DSIG/Encryption
  - JSON
- We use JSON due to ease of implementation:
  - JSON Web signature (JWS) for integrity protection
  - JSON Web Encryption (JWE) for confidentiality protection

- Not tied to a specific Diameter application.

- Authentication and key management is not part of this proposal:
  - Likely that “one size fits all” approach will not work due to different deployment environments

Signed-Data AVP

- The AVP carries JSON Web Signature (JWS) of one or more of AVPs. Each protected AVP is hashed and the hash is included into the JWS payload.
- Hashed AVPs are linked to “originals” using their AVP Code. If there are multiple instances of the same AVP, you hash them all and do one by one verification -> allows for rearranging AVPs and detection of addition/removal/modification of AVPs.
- Both JWS Payload and signature use the same hash algorithm of the cryptographic algorithm indicated in the JWS Header.
- Can be included into ***existing*** Diameter applications.

Encrypted-Data AVP

- The AVP carries JSON Web Encryption (JWE) data structure and the JWE Payload embeds of one or more protected AVPs.

- Cannot be used with existing Diameter applications since encrypted AVPs are embedded inside the Encrypted-Data AVP(s).

Error Handling

- Transient failures
  - DIAMETER_KEY_UNKNOWN - A Signed-Data or an Encrypted-Data AVP is received that was generated using a key that cannot be found in the key store. To recover a new end-to-end key establishment procedure may need to be invoked.

- Permanent failures
  - DIAMETER_DECRYPTION_ERROR - This error code is returned when an Encrypted-Data AVP is received and the decryption fails for an unknown reason.
  - DIAMETER_SIGNATURE_ERROR - This error code is returned when a Signed-Data AVP is received and the verification fails for an unknown reason.

Questions? Comments?