Guidance for COSE and JOSE Protocol Designers and Implementers

draft-tschofenig-jose-cose-guidance-00

Hannes Tschofenig , Les Hazlewood
Motivation

• Developers like to use JSON Object Signing and Encryption (JOSE) and CBOR Object Signing and Encryption (COSE) to secure their applications.

• As deployments increase, incorrect use also becomes more common.

• The purpose of this document is to provide guidance to reduce security-relevant misuse.

• Starting point is the way how keys are identified. Possible other content is:
  • API design guidance (suggested by Ilari)
  • Context Information Structure (suggested by Ilari)
  • Enc_structure/Enc_Recipient structure (suggested Laurence)
Key Identification

• To verify the signature covering a JOSE structure you need to look up the public key of the signer.

• This should be easy. Use the “kid” in the header.
  • Some hacks use the value in the kid to include a script (in the form of SQL injection) ...

• In the meanwhile there are also other fields carrying X.509 certificates and alike ...

• No indication what the semantic is when multiple identifiers are combined.
Key Identification, cont.

• Nothing in RFC 7515 states that the key identification values individually must be globally unique (and therefore "collision resistant").

• Developers sometimes combine different identifiers to accomplish this unique identification.

• Worse, information in payloads is used:
  • This requires a library to parse the unverified payload – often a JWT.
  • Fetch claim from the JWT.
  • Use claim or combination of claim and header info for key lookup.
  • Perform cryptographic verification of the JWT.
  • Parse JWT content again.

• Requires the developer to keep in mind that the information from the initial lookup was not cryptographically verified.
Next Steps

• Address feedback received on the list. Key identification was the starting point.
• Reach out to developers to involve them in a discussion.
  • Should we organize a workshop or another event?
  • How should good JOSE/COSE library design look like?
• Can we develop automated test cases?
• Looking for other contributors!