JSON Web Key Thumbprint

Mike Jones

July 21, 2014

IETF 90

Overview

- draft-jones-jose-jwk-thumbprint-00
- Defines “jkt” (JSON Web Key SHA-256 Thumbprint) values for JWK keys
  - Analogous to “x5t#S256” for X.509 keys
- Hash input uses JWK w/ required key fields
- Sorts them into lexicographic order
- Hashes UTF-8 representation of resulting JWK

Example

- For JWK containing “kty”:“RSA”, “n”, “e”, and possibly optional fields, sort fields lexicographically into order “e”, “kty”, “n”
- Create hash input as a JWK with no white space and unescaped char representations:
  - E.g., {"e":"AQAB","kty":"RSA","n":“0vx7…"}
- Hash UTF-8 representation
- “jkt” value is base64url encoded hash value

Non-Problems

- Sorting not a problem
  - – Lexicographic order based on character values
- Representing normal characters not problem
  - – Use unescaped representation
  - – E.g. Use “a” – not “\u0061”

Problem

- Characters requiring escaping a problem
  - E.g. Use “\\” or “\u005c” or “\u005C” for backslash?
- Solving this equivalent to defining a canonical JSON representation
- Working group input sought
  - Don’t want to go down the rat hole of defining complete canonical JSON for this purpose!
  - Should I just say that result undefined if such characters present?
    - And possibly revise later if the IETF adopts a canonical JSON?
  - Not an actual problem for any JWK representations we’ve defined or would expect to define, since don’t use those chars
  - Should we limit chars allowed in registered JWK names?

Move to Working Group?

- Can we add a charter item for this?