draft-fieau-interfaces-https-delegation-subcerts

IETF 113 – CDNI WG

Christoph Neumann – March 22nd, 2022
Status

• split ietf-cdni-interfaces-https-delegation into two drafts (as discussed and agreed in the CDNi working group)
  • ietf-cdni-interfaces-https-delegation covers delegated cerficates based on STAR/ACME [RFC9115]
  • draft-fieau-interfaces-https-delegation-subcerts covers delegated credentials [I-D.iert-tls-subcerts]

• delegated credentials draft is currently handled as an individual submission ➔ ask for adoption in the CDNi working group
Current version -01

• 2 MI objects
  • MI.ConfDelegatedCredentials
    • Contains URL where dCDN can fetch delegated credentials contained in a MI. DelegatedCredentials object
    • At each access to this URL by the dCDN a (new) MI. DelegatedCredentials is provided
  • MI.DelegatedCredentials
    • Contains delegated credential structure DelegatedCredential as defined in [I-D.ietf-tls-subcerts].
    • Contains private key corresponding to the public key contained in the DelegatedCredential
To Do / Next steps

• Align with draft-ietf-cdni-interfaces-https-delegation on FCI object. Would allow advertisement of capabilities/parameters specific to each method (e.g., number of credentials required).

   "capability-type": "FCI.SupportedDelegationMethods",
   "capability-value": {
     "delegation-methods": [
       "AcmeStarDelegationMethod",
       "DelegatedCredentialsMethod",
       ...
     ]
   }

• Add sections on privacy and security considerations

• Fix open issue related to MI.DelegatedCredentials and fetching mechanism of delegated credentials (see next slide)

• Add support of public/private key generation on dCDN side (corresponding to the public key contained in the DelegatedCredential)?
Open Issues and Options

• Open issue
  • MI.DelegatedCredentials is not really an MI object in the spirit of [RFC8006]
  • Only leaving MI.ConfDelegatedCredentials in the draft is pointless if we cannot describe or point to a mechanism to fetch delegated credentials

• Options
  • 1. Rely on an FCI object allowing the dCDN to advertise the number of delegated credentials needed
    • uCDN provides array of delegatedCredentials via a MI.ConfDelegatedCredentials object
    • Problem: lack of dynamicity, difficult to handle credential renewals
  • 2. Specify a dedicated interface to fetch delegated credentials
    • Option a: Via a new interface in CDNi/SVA
    • Option b: Via an extension in ACME (to be proposed in the ACME WG)
      • [I-D.ietf-tls-subcerts] already mentions this possibility.
      • But would ACME WG accept? In the scope of ACME (no CA involved)?
Thank you