draft-fieau-cdni-interfaces-https-delegation-02

CDNI WG

Frédéric Fieau, Emile Stephan — Orange
Sanjay Mishra — Verizon

IETF 100 – Singapore

Agenda

- Provide an update since last proposal
- Added support for delegation methods as defined by ACME/STAR and TLS/SubCerts drafts
- Define a new "SecureDelegation" metadata. Can be added via:
  - Option1: add a top level SecureDelegation object
  - Option2: extension to PathMetaData
- Pros and cons of options 1 and 2
- Other areas for consideration?
  - Identify other needs on CDNI interfaces for supporting HTTPS delegation
  - Discuss other delegation solutions for CDNI

Updates to `draft-fieau-cdni-interfaces-https-delegation` since -01

- `draft-fieau-cdni-interfaces-https-delegation` proposes extensions to the CDNi interfaces to exchange delegation metadata.

- This -02 version updates the delegation objects to support both:
  - Short Term Automatically Renewed certificates (STAR)
    - draft-ietf-acme-star
  - Delegated Credentials for TLS / SubCerts
    - draft-ietf-tls-subcerts (former draft-rescorla-tls-subcerts)

Support for ACME/STAR draft-ietf-acme-star

- Use case:
  - uCDN delegates HTTPS delivery to dCDN requesting the CA to issue a short-term automatically renewed certificate.

- Proposal:
  - Add metadata object in RFC8006 to support the draft ACME/STAR delegation model (draft-ietf-acme-star).

```
AcmeStarDelegationMethod: {
    "generic-metadata-type": "MI.AcmeStarDelegationMethod",
    "generic-metadata-value": {
        "starproxy": "10.2.2.2",
        "acmeserver" : "10.2.3.3",
        "credentialslocationuri": “www.ucdn.com/credentials",
        "periodicity": 36000
    }
}
```

update: support for TLS/SubCerts draft-ietf-tls-subcerts

- Use case:
  - uCDN delegates HTTPS delivery to dCDN using its own credentials without the need to request a certificate from the CA

- Proposal:
  - Add a new metadata object in RFC8006 to support the draft TLS/SubCerts delegation model (draft-ietf-tls-subcerts).

```
SubCertDelegationMethod: {
        "generic-metadata-type": "MI.SubcertsDelegationMethod",
        "generic-metadata-value": {
            "credentialsdelegatingentity": Endpoint,
            "credentialrecipiententity": Endpoint,
            "credentialslocationuri": Link,
            "periodicity": Periodicity
        }
}
```

SecureDelegation object over MI

- uCDN is delegating HTTPS delivery to dCDN, and it needs to convey information about how delegation is enforced.
- We propose two datamodel options that allows the uCDN to describe the « secure delegation » information to a dCDN.

- **1. SecureDelegation object defined as a top level object**
  - Define a top level object that can be exchanged to configure Secure Delegation
  - This is done just once for all paths and domains of the CDN Interconnection
  - Currently, this method doesn't exist in RFC8006, and thus requires a new SecureDelegation object.
- **2. SecureDelegation Extension to PathMetaData**
  - Define metadata extension to the PathMetaData that already exists in RFC8006
  - This method involves the definition of the delegation metadata for each path URL of the delegated entity (dCDN)

```
SecureDelegationMetadata
{
   "generic-metadata-type": "MI.SecuredDelegation"
   "generic-metadata-value":
   {
       "timewindow": TimeWindowACL,
       "methods": Array of DelegationMethods,
       "pathpattern": Array of PathPattern,
       "delegatedDomain": Array of HostMatch,
   }
 }
```

```
PathMetadata:
{
     "metadata": [
         {
         "generic-metadata-type": "MI.SecureDelegation"
         "generic-metadata-type": {
             "methods ": Array of DelegationMethods}
         }]
}
```

Examples

- **1. SecureDelegation object defined as a top level object**

SecureDelegationMetadata

```
{
        "generic-metadata-type": "MI.SecuredDelegation"
        "generic-metadata-value":
        {
            "timewindow": {start: 12932132,
end:23023944},
            "methods": ["MI.AcmeStarDelegationMethod"],
            "pathpattern": [["path-pattern": {
                "pattern": "/movies/*",
                "case-sensitive": true}]
            "delegatedDomain": «  »,
        }
}
```

- **2. SecureDelegation Extension to PathMetaData**

PathMatch:

```
{
    "path-pattern": {
        "pattern": "/movies/*",
        "case-sensitive": true},
    "path-metadata": {
        "type": "MI.PathMetadata",
        "href":
"https://metadata.ucdn/video.example.com/movies"}
}
```

PathMetadata:

```
{
    "metadata": [
        {
        "generic-metadata-type": "MI.SecureDelegation"
        "generic-metadata-type": {
            "methods ": [“MI.AcmeStarDelegationMethod”]}
        }]
}
```

Pros and Cons for option 1 and 2

- Option 1: SecureDelegation object defined as a top level object
  - ☺ Easy extensions : domain, new methods
  - ☹ Extends the CDNI metadata model

- Option 2: delegation metadata in Path Metadata
  - ☺ Path granularity
  - ☹ “limited to path”
  - ☹ Requires to repeat delegation metadata for each path

Other areas for consideration

- Identify other needs on CDNI interfaces for supporting HTTPS delegation
  - Purge, force cert renewal, …
- Discuss other delegation solutions for CDNI
  - Lurk, OOB, …

Thank you

STAR call-flow in CDNI