-mls-addl-creds

Richard Barnes
MLS @ IETF 117
# Credentials in RFC 9420

<table>
<thead>
<tr>
<th>Value</th>
<th>Name</th>
<th>R</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0000</td>
<td>RESERVED</td>
<td>-</td>
<td>RFC 9420</td>
</tr>
<tr>
<td>0x0001</td>
<td>basic</td>
<td>Y</td>
<td>RFC 9420</td>
</tr>
<tr>
<td>0x0002</td>
<td>x509</td>
<td>Y</td>
<td>RFC 9420</td>
</tr>
</tbody>
</table>
## Additional Credential Types

<table>
<thead>
<tr>
<th>Value</th>
<th>Name</th>
<th>Recommended</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0003</td>
<td>userinfo-vc</td>
<td>Y</td>
<td>RFC XXXX</td>
</tr>
<tr>
<td>0x0004</td>
<td>multi</td>
<td>Y</td>
<td>RFC XXXX</td>
</tr>
<tr>
<td>0x0005</td>
<td>weak-multi</td>
<td>Y</td>
<td>RFC XXXX</td>
</tr>
</tbody>
</table>
UserInfo Verifiable Credentials

Based on [Verifiable Credentials](https://www.w3.org/2018/credentials/v1) framework

Certificate-like (attrs, pubkey, sig)...

... but managed via [OpenID Connect](https://openid.net)

Main technical challenge is comparing JWKs (did:jwk) to MLS public keys

Draft implementation in MLSpp

```
JWT Payload:
{
  "vc": {
    "@context": [
      "https://www.w3.org/2018/credentials/v1"
    ],
    "type": [
      "VerifiableCredential",
      "UserInfoCredential"
    ],
    "credentialSubject": {
      "id": "did:jwk:eyJrdeHk101JFQyIsInVzZSI6InNpZyIsImNydiI6IlAtMjU2Iiwi
eCI6InFpR0tMdhSSm1KU19BT1FwV89IWEzYXNzZWZ6d1B3RiVvVzZtWkJ3
dncI5IjolA4bnl1THBKN5wmlaekNWS21HMFRGZXFQTwymZOT1VR
0F12ZUdkay1smFeZyI6IkVTrmjuZ2In0",
      "sub": "E46288761001",
      "name": "Jane Doe",
      "given_name": "Jane",
      "family_name": "Doe",
      "preferred_username": "j.doe",
      "email": "janedoe@example.com",
      "picture": "http://example.com/janedoe/me.jpg",
      "phone_number": "+1 202 555 1212"
    }
  },
  "iat": 1667575982,
  "exp": 1668180782,
  "iss": "https://server.example.com"
}
```
Multi-Credentials

Use cases:
- Multiple sources of attributes (e.g., communications provider + enterprise)
- Heterogeneous credential support

Multiple credentials all “bless” the LeafNode signature key by signing over it

Variants:
- “Multi” - Everyone must support all credentials in the multi-cred
- “Weak-multi” - Everyone must support at least one credential in the multi-cred

```c
struct {
    CipherSuite cipher_suite;
    Credential credential;
    SignaturePublicKey credential_key;

    /* SignWithLabel(., "CredentialBindingTBS", CredentialBindingTBS) */
} CredentialBinding

struct {
    CredentialBinding bindings<V>;
} MultiCredential;

struct {
    CredentialBinding bindings<V>;
} WeakMultiCredential;
```
Status & Questions

draft-00, but really draft-01 (we changed the name due to adding multi-cred)

Draft implementation in MLSpp

Interest in adopting in the MLS WG?

Other credential types folks are interested in?