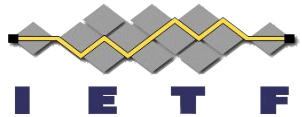
OAuth Status ~~Attestations~~ Assertions

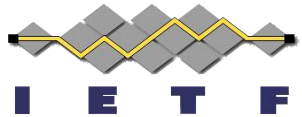
draft-demarco-oauth-status-assertions

Giuseppe De Marco, Orie Steele, Francesco Marino

IETF Interim meeting
June 11, 2024



What's the Status Assertion?



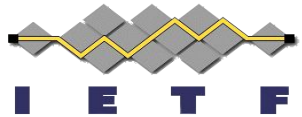
1. It is a signed artifact, JWT or CWT, demonstrating the validity of a digital credential, ensuring it hasn't been revoked.

2. It is issued by the **same Issuer of the Digital Credential**.

3. Holder obtains it automatically, and present it along with the related Credential to the Relying Party.

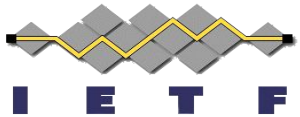
4. Allows:
   a. To not renew a Credentials requiring LoA High (not achievable using a refresh token).
   b. To avoid any interaction between the Relying Party and the Credential Issuers making untraceable the RP that uses a particular Credential.
   c. To not allow an RP to continuously monitor the status of a Credential, outside of the User authentication scopes.

Example: *A Digital Credential supporting Status Assertions*




```
{
    "vct": "https://credentials.example.com/identity_credential",
    "given_name": "John",
    …
    "birthdate": "1990-01-01",
    "is_over_18": true,
    "is_over_21": true,
    "is_over_65": false,
    "status": {
       "status_assertion": {
           "credential_hash_alg": "sha-256",   // the hash must be produced on the issuer signed part of the credential
       }
    }
}
```

Example: *Holder Requesting a status attestation*



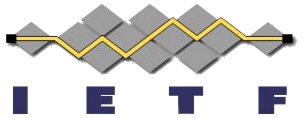
Request

```
POST /status HTTP/1.1
Host: issuer.example.org
Content-Type: application/json

{
  "status_assertion_requests" : [request-object-cred-A, request-object-cred-B]
}
```

Response

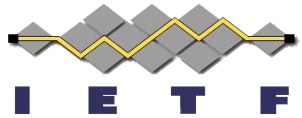
```
HTTP/1.1 200 Ok
Content-Type: application/json

{
  "status_assertion_responses": [status-assertion-cred-A, status-assertion-error]
}
```

Example: *Status Assertion Request Object*




```
{
     "alg": "ES256",
     "typ": "status-assertion-request+jwt"
}
.
{
     "iss": "0b434530-e151-4c40-98b7-74c75a5ef760",
     "aud": "https://issuer.example.org/status-assertion-endpoint",
     "iat": 1698744039,
     "exp": 1698830439,
     "jti": "6f204f7e-e453-4dfd-814e-9d155319408c",
     "credential_hash": $hash-about-the-Issuer-Signed-JWT
     "credential_hash_alg": "sha-256"
}
```

Example: *Status Assertion Object*




```
{
     "alg": "ES256",
     "typ": "status-assertion+jwt",
     "kid": $ISSUER-JWKID
}
.
{
     "iss": "https://issuer.example.org",
     "iat": 1504699136,
     "exp": 1504785536,
     "credential_hash": $CREDENTIAL-HASH,
     "credential_hash_alg": "sha-256",
     "cnf": {
         "jwk": {...}
     }
}
```

Next Steps

- Credential revocation status checks must be tailored to different use cases; a one-size-fits-all approach is not feasible:
  - OAuth Status List are good when the RP must periodically check the status of a credential, eg: service providing goods or services on a periodic basis. Another eg: Wallet Instance Attestation checks from the Credential Issuers (that otherwise should revoke the credentials issued at LoA High)
  - OAuth Status Assertions for all the other cases, eg: Private-RP must not be able to continuously monitor the Credential status outside of the user's authentication
- Adopt?