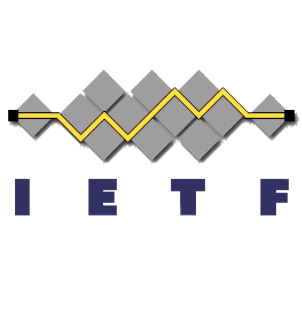
Mutual TLS Profiles for OAuth Clients

Brian Campbell
John Bradley
Nat Sakimura
Torsten Lodderstedt

IETF 98
Chicago
March 2017

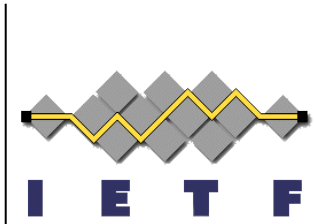
https://tools.ietf.org/html/draft-campbell-oauth-mtls-00

What is it?



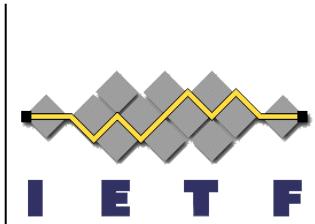
- Mutual TLS client authentication to the token endpoint
- Mutual TLS sender constrained protected resources access

Why Bother?



- Mutual TLS client authentication is something that's been done in practice for OAuth but we've never had a spec for it
- Mutual TLS sender constrained resources access binds access tokens to the client certificate so they can't be (re)played or used by any other entity
- Banks "need" these for server to server API use cases being driven by new open banking regulations

How it Works



- MTLS client authentication to the token endpoint
  - TLS connection from client to token endpoint is established or reestablished with mutual X509 certificate authentication
  - Client includes the "client_id" HTTP request parameter in all requests to the token endpoint
  - Trust model intentionally left open
- Mutual TLS sender constrained resource access
  - Associate a hash of the certificate with the access token
  - TLS connection from client to resource is also mutually authenticated
    - The protected resource matches certificate from TLS connection to the certificate hash in the access token
  - New JWT Confirmation Method
    - X.509 Certificate SHA-256 Thumbprint Confirmation Method: x5t#S256

```
{
  "iss": "https://server.example.com",
  "aud": "https://resource.example.org",
  "sub": "ty.webb@example.com",
  "exp": "1493726400",
  "nbf": "1493722800",
  "cnf":{
    "x5t#s256": "bwcK0esc3ACC3DB2Y5_lESsXE8o9ltc05O89jdN-dg2"
  }
}
```

Next Steps

- Consider adoption as a WG document!?!
- Get the band back together in Prague…