OAuth 2.0 Attestation-Based Client Authentication

Tobias Looker, Paul Bastian

A Refresher - Motivation

- Environments that *public* clients are operating/deployed in increasingly have primitives that can be used for client authentication examples such as:
  - App Attest on iOS for Native iOS applications
  - Play Integrity on Android for Native Android applications
- The question is how to appropriately use these capabilities to allow clients to authenticate with an authorization server?

Progress Update

Two proposals for discussion today

1. Proposal to use DPoP for the PoP syntax, instead of defining our own.
2. Proposal to support a header based syntax for the communication of the attestation to support other non-client authentication usecases such as DCR or presentation to an RS in a protected resource call.

Current Draft Token Request

```
POST /token HTTP/1.1
Host: as.example.com
Content-Type: application/x-www-form-urlencoded

grant_type=authorization_code&
code=n0esc3NRze7LTCu7iYzS6a5acc3f0ogp4&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3A
client-assertion-type%3Ajwt-client-attestation&
client_assertion=eyJhbGciOiJSUzI1NiIsImtpZCI6IjIyIn0.
eyJpc3Mi[...omitted for brevity...].
cC4hiUPo[...omitted for brevity...]~
eyJzI1NiIsImtphbGciOimtpZCI6IjIyIn0.
IjIyIn0[...omitted for brevity...].
iOiJSUzI1[...omitted for brevity...]
```

New assertion type

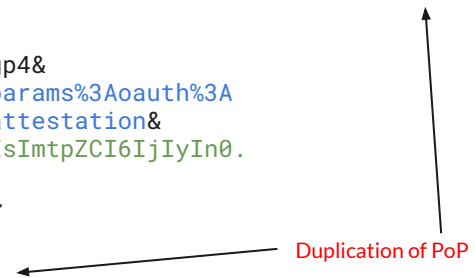
Two JWTs concatenated via a '~' character

- Client Attestation
- Client Attestation PoP

Current Draft Token Request + DPoP

```
POST /token HTTP/1.1
Host: as.example.com
Content-Type: application/x-www-form-urlencoded
DPoP:
eyJ0eXAiOiJkcG9wK2p3dCIsImp3ayI6eyJhbGciOiJFUzI1NiIsImNydiI6IlAtMjU2Iiwia3R5IjoiRUMiLCJ4IjoiaThReW03NFRNUHVLQXV
KUGlZczFSZlVsYTVjemNxelVobEpmRHNMdzd0NCIsInkiOiJGQjlUY2ZmeVZDSEpFQjJjejc4NTE2MUE0SmxlTkh2cG44bXhHRldZMlNjIn0sIm
FsZyI6IkVTMjU2In0.eyJqdGkiOiIzNTc2ODI5Ny1kZWM1LTQ2ZjYtODVlNS1iNzU4MjE2YWI1ZmYiLCJodG0iOiJQT1NUIiwiaHR1IjoiaHR0c
HM6Ly9hcy5leGFtcGxlL3Rva2VuIiwiaWF0IjoxNzAwODEyODAwLCJub25jZSI6ImV5SjdTX3pHLmV5SkgwLVouSFg0dy03diJ9.5VuDrkd8RhM
Raps_AzJBs2p-_UXXWT4dVHITBHiQxe31GeDq81otnIh3HBQN8_XjS1diHPq1tti1pn55eZdI5g

grant_type=authorization_code&
code=n0esc3NRze7LTCu7iYzS6a5acc3f0ogp4&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3A
client-assertion-type%3Ajwt-client-attestation&
client_assertion=eyJhbGciOiJSUzI1NiIsImtpZCI6IjIyIn0.
eyJpc3Mi[...omitted for brevity...].
cC4hiUPo[...omitted for brevity...]~
eyJzI1NiIsImtphbGciOimtpZCI6IjIyIn0.
IjIyIn0[...omitted for brevity...].
iOiJSUzI1[...omitted for brevity...]
```



Could we instead better reuse DPoP?

- Many use cases for this draft we expect to also use DPoP bound access tokens, the current draft would lead to more implementation complexity than would be ideal as there would need to be two PoP's in a token request *likely for the same key*.
- DPoP already defines a fit for purpose proof of possession syntax including nonce management, which our draft is re-inventing (client attestation pop).
- Realisation that DPoP's PoP syntax can be used without requiring the AS to issue DPoP bound access tokens.
- The draft also offers a potential way to authenticate the DPoP key back to the client, which is something that DPoP left out of scope but would be advantageous for some implementations see Issue #69, therefore meaning how this draft works with DPoP is important to optimise.

Could we reuse Attestations outside Client Authentication?

- At IETF Prague people asked to use this mechanism
  - At other endpoints, e.g. at RS
  - Not for Client Authentication, e.g. instead use DCR
- In essence is there any reason we would not consider shifting this attestation information into a header(s)?

New Proposed Token Request using Header + DPOP

```
POST /token HTTP/1.1
Host: as.example.com
Content-Type: application/x-www-form-urlencoded
DPoP:
eyJ0eXAiOiJkcG9wK2p3dCIsImp3ayI6eyJhbGciOiJFUzI1NiIsImNydiI6IlAtMjU2Iiwia3R5IjoiRUMiLCJ4IjoiaThReW03NFRNUHVLQXV
KUGlZczFSZlVsYTVjemNxelVobEpmRHNMdzd0NCIsInkiOiJGQjlUY2ZmeVZDSEpFQjJjejc4NTE2MUE0SmxlTkh2cG44bXhHRldZMlNjIn0sIm
FsZyI6IkVTMjU2In0.eyJqdGkiOiIzNTc2ODI5Ny1kZWM1LTQ2ZjYtODVlNS1iNzU4MjE2YWI1ZmYiLCJodG0iOiJQT1NUIiwiaHR1IjoiaHR0c
HM6Ly9hcy5leGFtcGxlL3Rva2VuIiwiaWF0IjoxNzAwODEyODAwLCJub25jZSI6ImV5SjdTX3pHLmV5SkgwLVouSFg0dy03diJ9.5VuDrkd8RhM
Raps_AzJBs2p-_UXXWT4dVHITBHiQxe31GeDq81otnIh3HBQN8_XjS1diHPq1tti1pn55eZdI5g
OAuth-Client-Attestation: eyJhbGciOiJSUzI1NiIsImtpZCI6IjIyIn0.eyJpc3Mi[...omitted for brevity...].
cC4hiUPo[...omitted for brevity...]

grant_type=authorization_code&
code=n0esc3NRze7LTCu7iYzS6a5acc3f0ogp4&
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3A
client-assertion-type%3Ajwt-client-attestation
```

Client Attestation PoP via DPoP syntax

Client Attestation via new header

Present if the attestation information communicated in the header is being used for client authentication.

Discussion

The main trade off with this proposal is that the DPoP key and the Client instance key MUST be the same. Which can viewed as both a useful simplification or a constraint.

Does the working group have an opinion on this? Are there practical use cases where the DPoP key and Client instance key need to be different? Or is the complexity it creates more hassle than it is worth?

This issue is time sensitive, given a major interoperability event in late April plans to use the draft and plans to also require DPoP bound access tokens.

What are the necessary JWT claims in this spec?

- Interesting features of Client Attestation in most cases is about the assurance of the (possibly hardware-protected) keys
  - Level of Assurance (see PR #51) -> possibly rename to attackPotentialResistance
  - Type of hardware key used
  - Type of user Authentication used
- Which of these things belong into this spec or other specs profiling this mechanism?

Links

Datatracker -> https://datatracker.ietf.org/doc/draft-ietf-oauth-attestation-based-client-auth

Git Repository -> https://github.com/vcstuff/draft-ietf-oauth-attestation-based-client-auth

Latest Editor Copy -> https://vcstuff.github.io/draft-ietf-oauth-attestation-based-client-auth/draft-ietf-oauth-attestation-based-client-auth.html

Questions?