draft-ietf-oauth-jwt-introspection-response-02

Vladimir Dzhuvinov, Torsten Lodderstedt

IETF-105
Mar 25 2019, Prague

What is it?

- Proposes an additional JWT-based response type for Token Introspection (RFC 7662)

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

 {
   "sub": "Z5O3upPC88QrAjx00dis",
   "aud": "https://protected.example.net/resource",
   "extension_field": "twenty-seven",
   "scope": "read write dolphin",
   "iss": "https://server.example.com/",
   "active": true,
   "exp": 1419356238,
   "iat": 1419350238,
   "client_id": "l238j323ds-23ij4",
   "username": "jdoe"
  }
```

```
HTTP/1.1 200 OK
Content-Type: application/jwt

eyJraWQiOiIxIiwiYWxnIjoiUlMyNTYifQ.eyJzdWIiOiJaNU8zdXBQQzg4UXJBa
ngwMGRpcyIsImF1ZCI6Imh0dHBzOlwvXC9wcm90ZWN0ZWQuZXhhbXBsZS5u
ZXRcL3Jlc291cmNlIiwiZXh0ZW5zaW9uX2ZpZWxkIjoidHdlbnR5LXNldmVuIiwic2
NvcGUiOiJyZWFkIHdyaXRlIGRvbHBoaW4iLCJpc3MiOiJodHRwczpcL1wvc2Vyd
mVyLmV4YW1wbGUuY29tXC8iLCJhY3RpdmUiOnRydWUsImV4cCI6MTQxOTM
1NjIzOCwiaWF0IjoxNDE5MzUwMjM4LCJjbGllbnRfaWQiOiJsMjM4ajMyM2RzLTI
zaWo0IiwidXNlcm5hbWUiOiJqZG9lIn0.HEQHf05vqVvWVnWuEjbzUnPz6JDQVR
69QkxgzBNq5kk-sK54ieg1STazXGsdFAT8nUhiiV1f_Z4HOKNnBs8TLKaFXokhA
0MqNBOYI--2unVHDqI_RPmC3p0NmP02Xmv4hzxFmTmpgjSy3vpKQDihOjhwN
Bh7G81JNaJqjJQTRv_1dHUPJotQjMK3k8_5FyiO2p64Y2VyxyQn1VWVlgOHlJw
hj6BaGHk4Qf5F8DHQZ1WCPg2p_-hwfINfXh1_buSjxyDRF4oe9pKy6ZB3ejh9qI
Mm-WrwltuU1uWMXxN6eS6tUtpKo8UCHBwLWCHmJN7KU6ZojmaISspdS23IEL
Alyw
```

Why?

- **High assurance level use cases** such as payments & electronic signing require signed access tokens due to auditing/non-repudiation requirements
- Use of structured access tokens not always possible
  - Integrated authorization for multiple services (e.g. sign a contract + initiate corresponding payment) operated by different providers
  - Cannot use single JWT carrying all necessary data (privacy)
- Token Introspection (RFC 7662) better fits but currently lacks signed responses

Changes since IETF-103 (-02)

- Updated references, e.g. to RFC 8414

Open Issue

- Justin Richer suggested to change the draft into general mechanism to enable JWT responses at any OAuth endpoint
- I asked the WG for their opinion (on the list):
    - Mike Jones: *“Let's do one thing well. Not create something that's extra-complicated without any clear use cases for doing so.”*
    - Justin Richer: *“I think it makes sense to have something that all of the OAuth JSON-spouting endpoints (introspection, token, revocation, registration, discovery) can use to universally put out signed and/or encrypted JWTs instead using the same mechanism.”*

Q & A