Web Authorization Protocol
Internet-Draft
Intended status: Informational
Expires: 1 January 2026

J. Lombardo
AWS
A. Babeanu
IndyKite

30 June 2025

OAuth 2.0 client extension claims
draft-lombardo-oauth-client-extension-claims-02

Abstract

This specification defines new claims for JWT profiled access tokens [RFC9068] so that resource providers can benefit from more granular information about the client: its authentication methods as well as the grant flow and the grant flow extensions used as part of the issuance of the associated tokens.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://identitymonk.github.io/draft-lombardo-oauth-client-extension-claims/draft-lombardo-oauth-client-extension-claims.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-lombardo-oauth-client-extension-claims/.

Discussion of this document takes place on the Web Authorization Protocol mailing list (mailto:oauth@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/oauth/. Subscribe at https://www.ietf.org/mailman/listinfo/oauth/.

Source for this draft and an issue tracker can be found at https://github.com/identitymonk/draft-lombardo-oauth-client-extension-claims.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction 3
2. Conventions and Definitions 4
3. JWT Access Token Client Extensions Data Structure 4
3.1. Client Flow Information Claims 4
3.2. Client Authentication Information Claims 5
4. Authorization Server Metadata 6
5. Requesting a JWT Access Token with Client Extensions 6
6. Validating JWT Access Tokens with Client Extensions 6
7. Security Considerations 6
7.1. Validation Of Token 6
7.2. Processing Of Claims Defined By This Document 6
7.3. Security Best Practice 7
8. IANA Considerations 7
8.1. OAuth Grant Type Registration 8
8.1.1. authorization_code grant type 8
8.1.2. implicit grant type 8
8.1.3. password grant type 8
8.1.4. client_credentials grant type 9
8.1.5. refresh_token grant type 9
8.1.6. jwt-bearer grant type 9
8.1.7. saml2-bearer grant type 9
8.1.8. token-exchange grant type 9
8.1.9. device_code grant type 9
8.1.10. ciba grant type 10
8.2. OAuth Grant Extension Type Registration 10

8.2.1. pkce grant extension type 10
8.2.2. dpop grant extension type 10
8.2.3. wpt grant extension type 10
8.2.4. rar grant extension type 10
8.2.5. par grant extension type 11
8.2.6. jar grant extension type 11
8.3. OAuth Token Endpoint Authentication Methods Registration 11
8.3.1. jwt-bearer token endpoint authentication method . . . 11
8.3.2. jwt-svid token endpoint authentication method 11
8.3.3. wit token endpoint authentication method 11
8.3.4. txn_token token endpoint authentication method . . . 12
8.4. Claims Registration 12
8.4.1. gty claim definition 12
8.4.2. cxt claim definition 12
8.4.3. ccr claim definition 12
8.4.4. cmr claim definition 13
9. References 13
9.1. Normative References 13
9.2. Informative References 14
Acknowledgments 16
Authors' Addresses 16

1. Introduction

Resource providers need information about the subject, the action, the resource, and the context involved in the request in order to be able to determine properly if a resource can be disclosed. This decision may also involve the help of a Policy Decision Point (PDP).

When accessed with a JWT profiled OAuth2 Access Token [RFC9068] presented as a bearer token [RFC6750], a resource provider receives mainly information about the subject in the form of: - The sub claim, - Any user profile claim set by the Authorization Server if applicable, - Any Authentication Information claims like the user class of authentication (acrclaim) or user method of authentication (claim amr [RFC8176]) - Any Authorization Information if applicable

The resource provider has very little information about the client, mainly in the form of the client_id [RFC8693] claim. It falls short in several important circumstances, for instance, in [FAPI2.0-Security-Profiles] or [hl7.fhir.uv.smart-app-launch] regulated APIs when they require peculiar client authentication mechanisms to be enforced or transaction specific details to be present in the token.

This document defines 4 new claims allowing to describe with more precise information the client and metadata on how it interacted with the authorization server during the issuance of the access token. It respects description of how to encode access tokens in JWT format.

The process by which the client interacts with the authorization server is out of scope.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

JWT access token: An OAuth 2.0 access token encoded in JWT format and complying to the requirements described in [RFC9068].

This specification uses the terms "access token", "authorization server", "authorization endpoint", "authorization request", "client", "protected resource", and "resource server" defined by "The OAuth 2.0 Authorization Framework" [RFC6749].

3. JWT Access Token Client Extensions Data Structure

The following claims extend the [RFC9068] access token payload data structure:

3.1. Client Flow Information Claims

The claims listed in this section MUST be issued and reflect grant type and extensions used with authorization server as part of the authorization request from the client. Their values are dynamic across all access tokens that derive from a given authorization response to reflect the elements used in the process that lead to their issuance.

gty: REQUIRED - defines the OAuth2 authorization grant type the client used for the issuance of the access token. String that is an identifier for an OAuth2 Grant type. Values used in the gty Claim MUST be from those registered in the IANA Grant Type Reference Values registry TODO established by this RFC and referencing, without being limited to, values established through section 2. of [RFC7591], section 2.1 of [RFC8693], and section 4 of [OpenID.CIBA].

cxt: REQUIRED - defines the list of extensions the client used in

conjunction with the OAuth2 authorization grant type used for the issuance of the access token. For example but not limited to: Proof Key for Code Exchange by OAuth Public Clients (or PKCE) as defined in [RFC7636], Demonstrating Proof of Possession (or DPoP) as defined in [RFC9449]. JSON array of strings that are identifiers for extensions used. Values used in the cxt Claim MUST be from those registered in the IANA Client Context Reference Values registry TODO established by this RFC and referencing, without being limited to, values established through section 2 of [RFC8414], and Section 5.1 of [RFC9449].

3.2. Client Authentication Information Claims

The claims listed in this section MAY be issued and reflect the types and strength of client authentication in the access token that the authentication server enforced prior to returning the authorization response to the client. Their values are fixed and remain the same across all access tokens that derive from a given authorization response, whether the access token was obtained directly in the response (e.g., via the implicit flow) or after obtaining a fresh access token using a refresh token. Those values may change if an access token is exchanged for another via an [RFC8693] procedure in order to reflect the specificities of this request.

ccr: OPTIONAL - defines the authentication context class reference the client satisfied when authenticating to the authorization server. An absolute URI or registered name from future RFC SHOULD be used as the ccr value; registered names MUST NOT be used with a different meaning than that which is registered. Parties using this claim will need to agree upon the meanings of the values used, which may be context specific.

cmr: OPTIONAL - defines the authentication methods the client used when authenticating to the authorization server. String that is an identifier for an authentication method used in the authentication of the client. For instance, a value might indicate the usage of private JWT as defined in [RFC7521] and [RFC7523] or HTTP message signature as defined in [RFC9421] . The cmr value is a case-sensitive string. Values used in the cmr Claim SHOULD be from those registered in the IANA OAuth Token Endpoint Authentication Methods Values registry [IANA.oauth-parameters_token-endpoint-auth-method] defined by [RFC7591]; parties using this claim will need to agree upon the meanings of any unregistered values used, which may be context specific.

4. Authorization Server Metadata

The following authorization server metadata parameters are introduced as an extension of [RFC8414], in order to describe the server's capabilities.

support_client_extentison_claims: Boolean parameter indicating to clients and resource servers whether the authorization server will return the extension claims described in this document.

Note: that the non presence of support_client_extentison_claims is sufficient for the client to determine that the server is not capable and therefore will not return the extension claims described in this RFC.

5. Requesting a JWT Access Token with Client Extensions

This specification does not change how clients interacts with authorization servers.

An authorization server supporting this specification MUST issue a JWT access token with client extensions claims described in this RFC in response to any authorization grant defined by [RFC6749] and subsequent extensions meant to result in an access token and as along as the authorization server support this capability.

6. Validating JWT Access Tokens with Client Extensions

This specification follows the requirements of the section 4 of [RFC9068].

7. Security Considerations

7.1. Validation Of Token

The JWT access token data format described here is the same as JWT access token defined by [RFC9068].

7.2. Processing Of Claims Defined By This Document

Any processor, client or resource server, MUST only process claims described in this document that it understands.

If a processor does not understand a claim described in this document or its value, it SHOULD ignore it.

7.3. Security Best Practice

The security current best practices described in [RFC9700] MUST be applied.

8. IANA Considerations

The following registration procedure is used for the registry established by this specification.

Values are registered on a Specification Required [RFC8126] basis after a two-week review period on the oauth-ext-review@ietf.org mailing list, on the advice of one or more Designated Experts. However, to allow for the allocation of values prior to publication of the final version of a specification, the Designated Experts may approve registration once they are satisfied that the specification will be completed and published. However, if the specification is not completed and published in a timely manner, as determined by the Designated Experts, the Designated Experts may request that IANA withdraw the registration.

Registration requests sent to the mailing list for review should use an appropriate subject (e.g., "Request to register JWT profiled OAuth2 Access Token client extensions: example").

Within the review period, the Designated Experts will either approve or deny the registration request, communicating this decision to the review list and IANA. Denials should include an explanation and, if applicable, suggestions as to how to make the request successful. The IANA escalation process is followed when the Designated Experts are not responsive within 14 days.

Criteria that should be applied by the Designated Experts includes determining whether the proposed registration duplicates existing functionality, determining whether it is likely to be of general applicability or whether it is useful only for a single application, and whether the registration makes sense.

IANA must only accept registry updates from the Designated Experts and should direct all requests for registration to the review mailing list.

It is suggested that multiple Designated Experts be appointed who are able to represent the perspectives of different applications using this specification, in order to enable broadly-informed review of registration decisions. In cases where a registration decision could be perceived as creating a conflict of interest for a particular Expert, that Expert should defer to the judgment of the other Experts.

The reason for the use of the mailing list is to enable public review of registration requests, enabling both Designated Experts and other interested parties to provide feedback on proposed registrations. The reason to allow the Designated Experts to allocate values prior to publication as a final specification is to enable giving authors of specifications proposing registrations the benefit of review by the Designated Experts before the specification is completely done, so that if problems are identified, the authors can iterate and fix them before publication of the final specification.

8.1. OAuth Grant Type Registration

This specification registers the following grant type in the [IANA.oauth-parameters] OAuth Grant Type registry.

8.1.1. authorization_code grant type

Type name: authorization_code

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.2. implicit grant type

Type name: implicit

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.3. password grant type

Type name: password

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.4. client_credentials grant type

Type name: client_credentials

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.5. refresh_token grant type

Type name: refresh_token

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.6. jwt-bearer grant type

Type name: urn:ietf:params:oauth:grant-type:jwt-bearer

Change controller: IETF

Specification document(s): section 2. of [RFC7591] and section 6 of [I-D.parecki-oauth-identity-assertion-authz-grant]

8.1.7. saml2-bearer grant type

Type name: urn:ietf:params:oauth:grant-type:saml2-bearer

Change controller: IETF

Specification document(s): section 2. of [RFC7591]

8.1.8. token-exchange grant type

Type name: urn:ietf:params:oauth:grant-type:token-exchange

Change controller: IETF

Specification document(s): section 2.1. of [RFC8693]

8.1.9. device_code grant type

Type name: urn:ietf:params:oauth:grant-type:device_code

Change controller: IETF

Specification document(s): section 3.4. of [RFC8628]

8.1.10. ciba grant type

Type name: urn:openid:params:grant-type:ciba

Change controller: IETF

Specification document(s): section 4. of [OpenID.CIBA]

8.2. OAuth Grant Extension Type Registration

This specification registers the following grant extension type in the [IANA.oauth-parameters] OAuth Grant Extension Type registry.

8.2.1. pkce grant extension type

Type name: pkce

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC7636]

8.2.2. dpop grant extension type

Type name: dpop

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC9449]

8.2.3. wpt grant extension type

Type name: wpt

Change controller: IETF

Specification document(s): This RFC as a reference to section 4.2 of [I-D.ietf-wimse-s2s-protocol]

8.2.4. rar grant extension type

Type name: rar

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC9396]

8.2.5. par grant extension type

Type name: par

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC9126]

8.2.6. jar grant extension type

Type name: jar

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC9101]

8.3. OAuth Token Endpoint Authentication Methods Registration

This specification registers additional token endpoint authentication methods in the [IANA.oauth-parameters] OAuth Token Endpoint Authentication Methods registry.

8.3.1. jwt-bearer token endpoint authentication method

Type name: jwt-bearer

Change controller: IETF

Specification document(s): This RFC as a reference to [RFC7591] and [I-D.parecki-oauth-identity-assertion-authz-grant]

8.3.2. jwt-svid token endpoint authentication method

Type name: jwt-svid

Change controller: IETF

Specification document(s): This RFC as a reference to SPIFFE JWT-SVID (https://github.com/spiffe/spiffe/blob/main/standards/JWT-SVID.md)

8.3.3. wit token endpoint authentication method

Type name: wit

Change controller: IETF

Specification document(s): This RFC as a reference to

[I-D.ietf-wimse-s2s-protocol]

8.3.4. txn_token token endpoint authentication method

Type name: txn_token

Change controller: IETF

Specification document(s): This RFC as a reference to [I-D.ietf-oauth-transaction-tokens]

8.4. Claims Registration

Section X.Y of this specification refers to the attributes gty, cxt, ccr, and cmr to express client metadata JWT access tokens. This section registers those attributes as claims in the [IANA.jwt] registry introduced in [RFC7519].

8.4.1. gty claim definition

Claim Name: gty

Claim Description: OAuth2 Grant Type used

Change Controller: IETF

Specification Document(s): Section X.Y of this document

8.4.2. cxt claim definition

Claim Name: cxt

Claim Description: Client Extensions used on top of the OAuth2 Grant Type

Change Controller: IETF

Specification Document(s): Section X.Y of this document

8.4.3. ccr claim definition

Claim Name: ccr

Claim Description: Client Authentication Class Reference

Change Controller: IETF

Specification Document(s): Section X.Y of this document

8.4.4. cmr claim definition

Claim Name: cmr

Claim Description: Client Authentication Methods Reference

Change Controller: IETF

Specification Document(s): Section X.Y of this document

9. References

9.1. Normative References

[IANA.oauth-parameters]
IANA, "OAuth Parameters",
<https://www.iana.org/assignments/oauth-parameters>.

[OpenID.CIBA]
Rodriguez, G. F., Ed., Walter, F., Ed., Nennker, A., Ed., Tonge, D., Ed., and B. Campbell, Ed., "OpenID Connect Client-Initiated Backchannel Authentication Flow - Core 1.0", September 2021, <https://openid.net/specs/openid-client-initiated-backchannel-authentication-core-1_0.html>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC6749] Hardt, D., Ed., "The OAuth 2.0 Authorization Framework", RFC 6749, DOI 10.17487/RFC6749, October 2012, <https://www.rfc-editor.org/rfc/rfc6749>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/rfc/rfc7519>.

[RFC7591] Richer, J., Ed., Jones, M., Bradley, J., Machulak, M., and P. Hunt, "OAuth 2.0 Dynamic Client Registration Protocol", RFC 7591, DOI 10.17487/RFC7591, July 2015, <https://www.rfc-editor.org/rfc/rfc7591>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8414] Jones, M., Sakimura, N., and J. Bradley, "OAuth 2.0 Authorization Server Metadata", RFC 8414, DOI 10.17487/RFC8414, June 2018, <https://www.rfc-editor.org/rfc/rfc8414>.

[RFC8628] Denniss, W., Bradley, J., Jones, M., and H. Tschofenig, "OAuth 2.0 Device Authorization Grant", RFC 8628, DOI 10.17487/RFC8628, August 2019, <https://www.rfc-editor.org/rfc/rfc8628>.

[RFC8693] Jones, M., Nadalin, A., Campbell, B., Ed., Bradley, J., and C. Mortimore, "OAuth 2.0 Token Exchange", RFC 8693, DOI 10.17487/RFC8693, January 2020, <https://www.rfc-editor.org/rfc/rfc8693>.

[RFC9068] Bertocci, V., "JSON Web Token (JWT) Profile for OAuth 2.0 Access Tokens", RFC 9068, DOI 10.17487/RFC9068, October 2021, <https://www.rfc-editor.org/rfc/rfc9068>.

9.2. Informative References

[FAPI2.0-Security-Profiles]
Fett, D. D., Ed., Tonge, D., Ed., and J. Heenan, Ed., "FAPI 2.0 Security Profile", n.d., <https://openid.net/specs/fapi-2_0-security-02.html>.

[hl7.fhir.uv.smart-app-launch]
"HL7 FHIR SMART App Launch", n.d., <https://www.hl7.org/fhir/smart-app-launch/app-launch.html#obtain-authorization-code>.

[I-D.ietf-oauth-transaction-tokens]
Tulshibagwale, A., Fletcher, G., and P. Kasselman, "Transaction Tokens", Work in Progress, Internet-Draft, draft-ietf-oauth-transaction-tokens-05, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-oauth-transaction-tokens-05>.

[I-D.ietf-wimse-s2s-protocol]
Campbell, B., Salowey, J., Schwenkschuster, A., and Y. Sheffer, "WIMSE Workload to Workload Authentication", Work in Progress, Internet-Draft, draft-ietf-wimse-s2s-protocol-05, 19 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-wimse-s2s-protocol-05>.

[I-D.parecki-oauth-identity-assertion-authz-grant]
Parecki, A., McGuinness, K., and B. Campbell, "Identity Assertion Authorization Grant", Work in Progress, Internet-Draft, draft-parecki-oauth-identity-assertion-authz-grant-04, 20 June 2025, <https://datatracker.ietf.org/doc/html/draft-parecki-oauth-identity-assertion-authz-grant-04>.

[IANA.jwt] IANA, "JSON Web Token (JWT)", <https://www.iana.org/assignments/jwt>.

[IANA.oauth-parameters_token-endpoint-auth-method]
IANA, "OAuth Token Endpoint Authentication Methods", <https://www.iana.org/assignments/oauth-parameters>.

[RFC6750] Jones, M. and D. Hardt, "The OAuth 2.0 Authorization Framework: Bearer Token Usage", RFC 6750, DOI 10.17487/RFC6750, October 2012, <https://www.rfc-editor.org/rfc/rfc6750>.

[RFC7521] Campbell, B., Mortimore, C., Jones, M., and Y. Goland, "Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants", RFC 7521, DOI 10.17487/RFC7521, May 2015, <https://www.rfc-editor.org/rfc/rfc7521>.

[RFC7523] Jones, M., Campbell, B., and C. Mortimore, "JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants", RFC 7523, DOI 10.17487/RFC7523, May 2015, <https://www.rfc-editor.org/rfc/rfc7523>.

[RFC7636] Sakimura, N., Ed., Bradley, J., and N. Agarwal, "Proof Key for Code Exchange by OAuth Public Clients", RFC 7636, DOI 10.17487/RFC7636, September 2015, <https://www.rfc-editor.org/rfc/rfc7636>.

[RFC8176] Jones, M., Hunt, P., and A. Nadalin, "Authentication Method Reference Values", RFC 8176, DOI 10.17487/RFC8176, June 2017, <https://www.rfc-editor.org/rfc/rfc8176>.

[RFC8705] Campbell, B., Bradley, J., Sakimura, N., and T. Lodderstedt, "OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens", RFC 8705, DOI 10.17487/RFC8705, February 2020, <https://www.rfc-editor.org/rfc/rfc8705>.

[RFC9101] Sakimura, N., Bradley, J., and M. Jones, "The OAuth 2.0 Authorization Framework: JWT-Secured Authorization Request (JAR)", RFC 9101, DOI 10.17487/RFC9101, August 2021, <https://www.rfc-editor.org/rfc/rfc9101>.

[RFC9126] Lodderstedt, T., Campbell, B., Sakimura, N., Tonge, D., and F. Skokan, "OAuth 2.0 Pushed Authorization Requests", RFC 9126, DOI 10.17487/RFC9126, September 2021, <https://www.rfc-editor.org/rfc/rfc9126>.

[RFC9396] Lodderstedt, T., Richer, J., and B. Campbell, "OAuth 2.0 Rich Authorization Requests", RFC 9396, DOI 10.17487/RFC9396, May 2023, <https://www.rfc-editor.org/rfc/rfc9396>.

[RFC9421] Backman, A., Ed., Richer, J., Ed., and M. Sporny, "HTTP Message Signatures", RFC 9421, DOI 10.17487/RFC9421, February 2024, <https://www.rfc-editor.org/rfc/rfc9421>.

[RFC9449] Fett, D., Campbell, B., Bradley, J., Lodderstedt, T., Jones, M., and D. Waite, "OAuth 2.0 Demonstrating Proof of Possession (DPoP)", RFC 9449, DOI 10.17487/RFC9449, September 2023, <https://www.rfc-editor.org/rfc/rfc9449>.

[RFC9700] Lodderstedt, T., Bradley, J., Labunets, A., and D. Fett, "Best Current Practice for OAuth 2.0 Security", BCP 240, RFC 9700, DOI 10.17487/RFC9700, January 2025, <https://www.rfc-editor.org/rfc/rfc9700>.

Acknowledgments

The authors wants to acknowledge the support and work of the following individuals: George Fletcher (Practical Identity), Christopher Langton (Vulnetix).

The authors wants also to recognize the trail blazers and thought leaders that created the ecosystem without which this draft proposal would not be able to solve customer pain points and secure usage of digital services, especially without being limited to: Vittorio Bertocci†, Brian Campbell (Ping Identity), Justin Richer (MongoDB), Aaron Parecki (Okta), Pieter Kasselman (SPRL), Dr Mike Jones (Self-Issued Consulting, LLC), Dr Daniel Fett (Authlete).

Authors' Addresses

Jean-Franois Lombardo
AWS
Canada

Email: jeffsec@amazon.com

Alexandre Babeanu
IndyKite
Canada
Email: alex.babeanu@indykite.com