Web Authorization Protocol
Internet-Draft
Intended status: Informational
Expires: 5 January 2026

A. Parecki
Okta
D. Fett
J. Heenan
Authlete
4 July 2025

OAuth 2.0 Client ID Prefix
draft-parecki-oauth-client-id-prefix-00

Abstract

This specification defines the concept of a Client Identifier Prefix to enable Authorization Servers and Clients to use more than one mechanism to obtain and validate Client metadata.

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://drafts.aaronpk.com/oauth-client-id-prefix/draft-parecki-oauth-client-id-prefix.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-parecki-oauth-client-id-prefix/.

Discussion of this document takes place on the Web Authorization Protocol Working Group 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/aaronpk/oauth-client-id-prefix.

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

A Client Identifier is used by an OAuth 2.0 Client to identify itself to an Authorization Server. The Client Identifier is used in the Authorization Request and various other places throughout OAuth flows. In ecosystems where more than one method of obtaining and validating Client metadata is used, it is necessary to indicate unambiguously which method is used. This specification defines a structure for Client Identifiers that includes a prefix indicating the Client Identifier Prefix.

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.

3. Client Identifier Prefix

This specification defines the concept of a Client Identifier Prefix that indicates how an Authorization Server is supposed to interpret the Client Identifier and associated data in the process of Client identification, authentication, and authorization. The Client Identifier Prefix enables deployments of this specification to use different mechanisms to obtain and validate metadata of the Client beyond the scope of [RFC6749].

The Client Identifier Prefix is a string that MAY be communicated by the Client in a prefix within the client_id parameter in the Authorization Request. A fallback to pre-registered Clients as in [RFC6749] or a default Client Identifier Prefix is in place as a default mechanism in case no Client Identifier Prefix was provided. A certain Client Identifier Prefix may require the Client to sign the Authorization Request as means of authentication and/or pass additional parameters and require the Authorization Server to process them.

3.1. Syntax

In the client_id Authorization Request parameter and other places where the Client Identifier is used, the Client Identifier Prefixes are prefixed to the usual Client Identifier, separated by a : (colon) character:

<client_id_prefix>:<orig_client_id>

Here, <client_id_prefix> is the Client Identifier Prefix and <orig_client_id> is an identifier for the Client within the namespace of that prefix. See Section 3.3 for Client Identifier Prefixes defined by this specification.

Authorization Servers MUST use the presence of a : (colon) character and the content preceding it to determine whether a Client Identifier Prefix is used. If a : character is present, and the content preceding it is a recognized and supported Client Identifier Prefix value, the Authorization Server MUST interpret the Client Identifier according to the given Client Identifier Prefix. The Client Identifier Prefix is defined as the string before the (first) : character. If the Authorization Server does not support the Client Identifier Prefix, the Authorization Server MUST refuse the request.

For example, an Authorization Request might contain client_id=client_attestation:example-client to indicate that the client_attestation Client Identifier Prefix is to be used and that within this prefix, the Client can be identified by the string example-client.

Note that the Client may need to determine which Client Identifier Prefixes the Authorization Server supports prior to sending the Authorization Request in order to ensure the client’s preferred prefix is supported.

3.2. Fallback for Unrecognized Client ID Prefixes

If a : character is not present in the Client Identifier, the Authorization Server MUST treat the Client Identifier as referencing a pre-registered client. This is equivalent to the [RFC6749] default behavior, i.e., the Client Identifier needs to be known to the Authorization Server in advance of the Authorization Request. The Client metadata is pre-registered using [RFC7591] or through out-of-band mechanisms.

For example, if an Authorization Request contains client_id=example-client, the Authorization Server would interpret the Client Identifier as referring to a pre-registered client.

If a : character is present in the Client Identifier but the value preceding it is not a recognized and supported Client Identifier Prefix value, the Authorization Server MAY treat the Client Identifier as having a default Client Identifier Prefix.

For example, an Authorization Request containing a client_id value of https://client.example.com/metadata.json could be interpreted by the Authorization Server as referring to a Client ID Metadata Document [I-D.draft-parecki-oauth-client-id-metadata-document], with the default Client Identifier Prefix being client-id-metadata-document.

From this definition, it follows that pre-registered clients MUST NOT contain a : character preceded immediately by a supported Client Identifier Prefix value in the first part of their Client Identifier.

3.2.1. Example

Deployments that use https URLs as client IDs and that have only one way to resolve client metadata from the URL, MAY use only the full https URL as the client ID. If there is only one way to resolve client metadata then there is no ambiguity in which metadata retrieval method to use, and are not susceptible to client identifier mixup attacks as described in Section 6.1.

For example, an authorization server using only the Client ID Metadata Document [I-D.draft-parecki-oauth-client-id-metadata-document] method to retrieve client metadata MAY accept client IDs such as:

https://client.example.com/metadata.json

This results in this non-normative example of an authorization request:

```
GET /authorize?
  response_type=code
  &client_id=https%3A%2F%2Fclient.example.org%2Fmetadata.json
  &redirect_uri=https%3A%2F%2Fclient.example.org%2Fcallback
  &code_challenge=GdE4nqBrwRxQfN2Y8fq3rrYk_kkpwg6tQ74J94-2nHw
  &code_challenge_method=S256
  &scope=write
```

3.3. Defined Client Identifier Prefixes

This specification defines the following Client Identifier Prefixes, followed by the examples where applicable:

* redirect_uri: This value indicates that the Client Identifier (without the prefix redirect_uri:) is the Client's Redirect URI (or Response URI when Response Mode direct_post is used). The Authorization Request MUST NOT be signed. The Client MAY omit the redirect_uri Authorization Request parameter. Example Client Identifier: redirect_uri:https%3A%2F%2Fclient.example.org%2Fcb.

* client_id_metadata_document: This value indicates that the Client Identifer (without the prefix client_id_metadata_document:) is the client's Client ID Metadata Document [I-D.draft-parecki-oauth-client-id-metadata-document].

* https: This Client Identifier Prefix MUST NOT be registered.

4. Example

The following is a non-normative example of an authorization request with the client_id_metadata_document Client ID Prefix:

```
GET /authorize?
  response_type=code
  &client_id=client_id_metadata_document:https%3A%2F%2Fclient.example.org%2Fmetadata.json
  &redirect_uri=https%3A%2F%2Fclient.example.org%2Fredirect
  &code_challenge=GdE4nqBrwRxQfN2Y8fq3rrYk_kkpwg6tQ74J94-2nHw
  &code_challenge_method=S256
  &scope=write
```

5. Authorization Server Metadata

Authorization servers that publish Authorization Server Metadata ([RFC8414]) MUST include the following properties to indicate support for client ID prefixes as described in this specification.

client_id_prefixes_supported: REQUIRED. A JSON array of strings indicating the registered client ID prefixes supported by this authorization server.

6. Security Considerations

6.1. Client Identifier Mixups

Confusing Clients using a Client Identifier Prefix with those using none can lead to various mixup attacks. Therefore, Authorization Servers MUST always use the full Client Identifier, including the prefix if provided, within the context of the Authorization Server or its responses to identify the client. This refers in particular to places where the Client Identifier is used in [RFC6749] as well as in any artifacts such as the aud claim of JWT access tokens [RFC9068].

7. IANA Considerations

7.1. OAuth Authorization Server Metadata Registry

The following authorization server metadata value is defined by this specification and (TBD) registered in the IANA "OAuth Authorization Server Metadata" registry established in OAuth 2.0 Authorization Server Metadata [RFC8414].

* Metadata Name: client_id_prefixes_supported

* Metadata Description: A JSON array of strings indicating the client ID prefixes supported by the authorization server.

* Change Controller: IETF

* Specification Document: Section 5 of [[this specification]]

8. References

8.1. Normative References

[DID-Core] "DID Core", 19 July 2022, <https://www.w3.org/TR/did-core/>.

[OpenID.Federation] Hedberg, R., Jones, M.B., Solberg, A.., Bradley, J., Marco, G. D., and V. Dzhuvinov, "OpenID Federation 1.0", 17 May 2024, <https://openid.net/specs/openid-federation-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>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[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>.

[RFC7515] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Signature (JWS)", RFC 7515, DOI 10.17487/RFC7515, May 2015, <https://www.rfc-editor.org/rfc/rfc7515>.

[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>.

8.2. Informative References

[I-D.draft-parecki-oauth-client-id-metadata-document] Parecki, A. and E. Smith, "OAuth Client ID Metadata Document", Work in Progress, Internet-Draft, draft-parecki-oauth-client-id-metadata-document-02, 9 January 2025, <https://datatracker.ietf.org/doc/html/draft-parecki-oauth-client-id-metadata-document-02>.

[OpenID] Sakimura, N., Bradley, J., Jones, M., Medeiros, B. de., and C. Mortimore, "OpenID Connect Core 1.0", 15 December 2023, <https://openid.net/specs/openid-connect-core-1_0.html>.

[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>.

[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>.

Acknowledgments

The authors would like to thank the following people for their contributions and reviews of this specification:

Brian Campbell, Emelia Smith.

Authors' Addresses

Aaron Parecki
Okta
Email: aaron@parecki.com

Daniel Fett
Authlete

Email: mail@danielfett.de

Joseph Heenan
Authlete
Email: joseph@heenan.me.uk