OAuth Working Group B. Campbell
Internet-Draft J. Bradley
Intended status: Standards Track Ping Identity
Expires: February 4, 2019 H. Tschofenig
Arm Limited
August 3, 2018

Resource Indicators for OAuth 2.0
draft-ietf-oauth-resource-indicators-00

Abstract

This straw-man specification defines an extension to The OAuth 2.0 Authorization Framework that enables the client and authorization server to more explicitly to communicate about the protected resource(s) to be accessed.

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/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 4, 2019.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
  1.1. Requirements Notation and Conventions 3
  1.2. Terminology 3
2. Resource Parameter 3
3. IANA Considerations 5
  3.1. OAuth Parameters Registration 5
    3.1.1. Registry Contents 5
  3.2. OAuth Extensions Error Registration 6
    3.2.1. Registry Contents 6
4. Security Considerations 6
5. References 7
  5.1. Normative References 7
  5.2. Informative References 7
Appendix A. Acknowledgements 8
Appendix B. Document History 8
Authors' Addresses 8

1. Introduction

Several years of deployment and implementation experience with OAuth 2.0 [RFC6749] has uncovered a need, in some circumstances, for the client to explicitly signal to the authorization sever where it intends to use the access token it is requesting.

Knowing which resource server will process the access token enables the authorization server to construct the token as necessary for that entity. Properly encrypting the token (or content within the token) to a particular resource server, for example, requires knowing which resource server will receive and decrypt the token. Furthermore, various resource servers oftentimes have different requirements with respect to the data contained in, or referenced by, the token and knowing the resource server where the client intends to use the s token allows the the authorization server to mint the token accordingly.

Specific knowledge of the intended recipient(s) of the access token also helps facilitate improved security characteristics of the token itself. Bearer tokens, currently the only defined type of OAuth access token, allow any party in possession of a token to get access to the associated resources. To prevent misuse, two important security assumptions must hold: bearer tokens must be protected from disclosure in storage and in transit and the access token must only be valid for use at a specific resource server and for a specific

scope. When the authorization server is informed of the resource server that will process the access token, it can restrict the intended audience of that token such that it cannot be used at other resource servers. Section 5.2 of OAuth 2.0 Authorization Framework: Bearer Token Usage [RFC6750] prescribes including the token's intended recipients within the token to prevent token redirect.

Scope, from Section 3.3 of OAuth 2.0 [RFC6749], sometimes is overloaded to convey the location or identity of the resource server, however, doing so isn't always feasible or desirable. Scope is typically about what access is being requested rather than where that access will be redeemed (e.g. "email", "user:follow", "user_photos", and "channels:read" are a small sample of scope values in use).

A means for the client to signal to the authorization sever where it intends to use the access token it's requesting is important and useful. A number of implementations and deployments of OAuth 2.0 have already employed proprietary parameters toward that end. This specification aims to provide a standardized and interoperable alternative to the proprietary approaches going forward.

1.1. Requirements Notation and Conventions

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 RFC 2119 [RFC2119].

1.2. Terminology

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

2. Resource Parameter

The client may indicate the resource server(s) for which it is requesting an access token by including the following parameter in the request.

resource

OPTIONAL. The value of the "resource" parameter indicates a resource server where the requested access token will be used. It MUST be an absolute URI, as specified by Section 4.3 of[RFC3986], and MUST NOT include a query or fragment component. If the authorization server fails to parse the provided value or does not

consider the resource server acceptable, it MUST reject the request and provide an error response with the error code "invalid_resource". Multiple "resource" parameters may be used to indicate that the issued token is intended to be used at multiple resource servers.

When an access token will be returned from the authorization endpoint, the "resource" parameter is used in the authorization request to the authorization endpoint as defined in Section 4.2.1 of OAuth 2.0 [RFC6749]. An example of an authorization request where the client tells the authorization server that it wants a token for use at "https://rs.example.com/" is shown in Figure 1 below.

```
GET /as/authorization.oauth2?response_type=token
   &client_id=s6BhdRkqt3&state=laeb
   &redirect_uri=https%3A%2F%2Fclient%2Eexample%2Ecom%2Fcb
   &resource=https%3A%2F%2Frs.example.com%2F HTTP/1.1
Host: authorization-server.example.com
```

Figure 1: Protected Resource Request

When the access token is returned from the token endpoint, the request parameter is included in the token request to the token endpoint. Sections 4.1.1, 4.3.1, 4.4.2, 4.5 and 6 of OAuth 2.0 [RFC6749] define requests to the token endpoint with different grant types. An example of a token request, using a refresh token, where the client tells the authorization server that it wants a token for use at "https://rs.example.com/" is shown in Figure 2 below.

```
POST /as/token.oauth2 HTTP/1.1
Host: authorization-server.example.com
Authorization: Basic czZCaGRSa3F0Mzpoc3FFelFsVW9IQUU5cHg0RlNyNHlJ
Content-Type: application/x-www-form-urlencoded

grant_type=refresh_token
&refresh_token=4LTC8lb0acc6Oy4esc1Nk9BWC0imAwH
&resource=https%3A%2F%2Frs.example.com%2F
```

Figure 2: Protected Resource Request

The "resource" parameter indicates the physical location of resource server, typically as an https URL, where the client intends to use the requested access token. This enables the authorization server to apply policy as appropriate for the resource, such as determining the type and content of the token to be issued, if and how the token is to be encrypted, and applying appropriate audience restrictions to the token.

The client SHOULD provide the most specific URI that it can for the set of resources or API it intends to access. In practice a client will know a base URI for the resource server application that it interacts with, which is appropriate to use as the value of the "resource" parameter. The client SHOULD use the base URI for the API unless specific knowledge of resource server dictates the client use a shorter path. For example, the value "https://rs.example.com/" would be used for a resource server that is the exclusive application on that host, however, if the resource server is one of many applications on that host, something like "https://rs.example.com/application/" would be used. Another example, for an API like SCIM [RFC7644] that has multiple endpoints such as "https://rs.example.com/scim/Users", "https://rs.example.com/scim/Groups", and "https://rs.example.com/scim/Schemas" The client should use "https://rs.example.com/scim/" as the resource so that the issued access token is valid for all the endpoints of the SCIM API.

The authorization server SHOULD audience restrict the access token to the resource server(s) indicated by the "resource" parameter. Audience restrictions can be communicated in JSON Web Tokens [RFC7519] with the "aud" claim and the top-level member of the same name provides the audience restriction information in a Token Introspection [RFC7662] response. The authorization server may use the exact "resource" value as the audience or it may map from that value to a more general URI or abstract identifier for the resource server.

The requested resource pertains to the access token that is the expected result of the request and not to the underlying access granted by the resource owner.

3. IANA Considerations

3.1. OAuth Parameters Registration

This specification registers the following value in the IANA "OAuth Parameters" registry [IANA.OAuth.Parameters] established by [RFC6749].

3.1.1. Registry Contents

- Parameter name: resource
- Parameter usage location: authorization request, token request
- Change controller: IESG
- Specification document(s): Section 2 of [[this specification]]

3.2. OAuth Extensions Error Registration

This specification registers the following error in the IANA "OAuth Extensions Error Registry" [IANA.OAuth.Parameters] established by [RFC6749].

3.2.1. Registry Contents

- Error name: invalid_resource
- Error usage location: implicit grant error response, token error response
- Related protocol extension: resource parameter
- Change controller: IESG
- Specification document(s): Section 2 of [[this specification]]

4. Security Considerations

An access token that is audience restricted to a resource server, which obtains the token legitimately, cannot be used to access resources on behalf of the resource owner at other resource servers. The "resource" parameter enables a client to indicate the resource server where the requested access token will be used, which in turn enables the authorization server to apply the appropriate audience restrictions to the token.

Some Resource servers may host user content or be multi-tenant. In order to avoid attacks that might confuse a client into sending a AT to a user controlled resource it is important to use the a specific resource URI including path and not use just a host with no path. This will cause any AT issued for accessing the user controlled resource to have a invalid audience if replayed against the legitimate resource API.

Although multiple occurrences of the "resource" parameter may be included in a request, using only a single "resource" parameter is encouraged. A bearer token that has multiple intended recipients (audiences) can be used by any one of those recipients at any other. Thus, a high degree of trust between the involved parties is needed when using access tokens with multiple audiences. Furthermore an authorization server may be unwilling or unable to fulfill a token request with multiple resources.

[[TODO: I continue to question the value of allowing multiple resources vs the functional and security complexity that comes with doing so. Writing the preceding paragraph just underscores that concern. So just noting it here.]]

5. References

5.1. Normative References

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

[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/info/rfc2119>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

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

5.2. Informative References

[I-D.draft-tschofenig-oauth-audience]
Tschofenig, H., "OAuth 2.0: Audience Information", draft-tschofenig-oauth-audience (work in progress), February 2013.

[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/info/rfc6750>.

[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/info/rfc7519>.

[RFC7644] Hunt, P., Ed., Grizzle, K., Ansari, M., Wahlstroem, E., and C. Mortimore, "System for Cross-domain Identity Management: Protocol", RFC 7644, DOI 10.17487/RFC7644, September 2015, <https://www.rfc-editor.org/info/rfc7644>.

[RFC7662] Richer, J., Ed., "OAuth 2.0 Token Introspection", RFC 7662, DOI 10.17487/RFC7662, October 2015, <https://www.rfc-editor.org/info/rfc7662>.

Appendix A. Acknowledgements

The following individuals contributed to discussions relating to and giving rise to this draft specification:

George Fletcher, Hans Zandbelt, Justin Richer, Michael Jones, Nat Sakimura, Phil Hunt, Sergey Beryozkin, and Anthony "no go" Nadalin.

Appendix B. Document History

[[to be removed by the RFC Editor before publication as an RFC]]

-00

- First version of the working group document. A replica of draft-campbell-oauth-resource-indicators-02.

Authors' Addresses

Brian Campbell
Ping Identity

Email: brian.d.campbell@gmail.com

John Bradley
Ping Identity

Email: ve7jtb@ve7jtb.com

Hannes Tschofenig
Arm Limited

Email: hannes.tschofenig@gmx.net