OAuth Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 07, 2014

M. Kelly
Stateless
N. Sakimura, Ed.
Nomura Research Institute

November 03, 2013

JSON Metadata for OAuth Responses 1.0
draft-sakimura-oauth-meta-03

Abstract

This specification defines an extensible metadata member that may be inserted into the OAuth 2.0 responses to assist the clients to process those responses. It is expressed as a member called "_links" that is inserted as the top level member in the responses. It will allow the client to learn where the members in the response could be used and how, etc. Since it is just a member, any client that does not understand this extension should not break and work normally while supporting clients can utilize the metadata to its advantage.

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 http://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 May 07, 2014.

Table of Contents

1. Introduction

Although OAuth 2.0 [RFC6749] has been known for its REST friendliness, OAuth itself is not RESTful, as it heavily relies on out-of-band information to drive the interactions. This situation can be eased by hypertext-enabling the JSON responses through the introduction of a member that represents such hypertext and other metadata. To achieve this, this specification introduces a top level member "_links" that represents various link relationships and other metadata.

2. Requirements

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. JSON Meta Object

A JSON Meta Object uses the format described in [RFC4627] and is intended to be inserted into a JSON document to express some of the metadata associated with it as "_links" member.

The value of the "_links" member is a JSON object that expresses link relations ("rel"), which in turn holds an object with "href" and other members or an array of such objects.

Following non-normative schematic example should help envisage what it would look like. (Note: line-wraps are for display purpose only.)

```
{
  "_links":{
    "self":{"href":"https://example.com/token?code=123"},
    "userinfo":
      {
        "href":"https://example.com/user/{user_id}",
        "Authorize":"{token_type} {access_token}"
      }
  },
  "token_type":"Bearer",
  "access_token":"aCeSsToKen"
}
```

Here, we have "_links" member that expresses various "relations" such as "self" and "userinfo", which is a resource type of OpenID Connect's Userinfo endpoint. Each relationships has either a link relations object or an array of link relations objects as its value. The link relations objects holds various members such as "href". They are explained in the next section.

3.1. _links Member

"_links" member holds exactly one object that contains the following members with relation as the "string" defined in [RFC4627]. The "string" SHOULD be a link relation type that is either defined in the IANA registry defined in Web Linking (Web Linking) or a URI that describes the relation.

Each relation member holds exactly one object or one array, whose elements are objects. Each object has following members, which are all optional.

3.1.1. href

The value of the "href" member is a URI Template [RFC6570] that the relation points to. The values for template parameters SHOULD be taken from the value of the top-level members in the including JSON object whose "string" matches the template variable name.

3.1.2. Authorize

The HTTP Authorize header defined in Hypertext Transfer Protocol -- HTTP/1.1 [RFC2616] to be used when accessing the resource identified by href. It is templated in exactly the same syntax as in URI Template [RFC6570] except that it is applied to the Authorization request header than the URI.

3.1.3. content-type

The content-type to be used when the parameters are sent to the URL.

[todo] Locate the proper reference and name for content transfer encodings.

e.g., "application/x-www-form-urlencoded", "multipart/form-data", "application/json".

4. Application to the OAuth 2.0 Token Endpoint Responses

To create the Section 3 should be used in the token endpoint responses of the OAuth 2.0 Authorization Framework [RFC6749], following relations SHOULD be included.

4.1. Successful Responses

In the case of the Successful Response described in section 5.1. of [RFC6749], the following member SHOULD be present in the value of the "_links" member described in _links Member (Section 3.1) of this specification.

4.1.1. self

An object with the following members.

href REQUIRED. The URI that resulted in this response.

4.1.2. describedby

An object with the following members.

href REQUIRED. The value is one of the following URIs: "http://tools.ietf.org/html/rfc6749#section-4.1.4" (Access Token Response of Authorization Code Grant), "http://tools.ietf.org/html/rfc6749#section-4.3.3" (Access Token Response of Resource Owner Password Credentials Grant), "http://tools.ietf.org/html/rfc6749#section-4.4.3" (Access Token Response of Client Credentials Grant). [[editor's note. Add Assertion Flows as well.]]

4.1.3. Protected Resources

Each protected resources MUST provide a unique Relation Name by either registering to the Link Relation Type Registry defined in section 6.2 of [RFC5988]or providing an absolute URI that provides a collision registant name. The value is an array of objects that has the following members.

href REQUIRED. The URI template that describes the request to the resource as described in href (Section 3.1.1).

content-type OPTIONAL. As described in content-type (Section 3.1.3).

Authorize OPTIONAL. HTTP Authorization header to be sent when accessing the resource. This is described in Authorize (Section 3.1.2). If this member is not available, then the client SHOULD access the expanded "href" value to obtain the Authorization header response to learn what authorization scheme it should use.

4.2. Error Responses

In the case of the Error Response described in section 5.2. of [RFC6749], the folloing member SHOULD be present.

4.2.1. self

An object with the following members.

href REQUIRED. The URI that resulted in this response.

4.2.2. describedby

An object with the following members.

href REQUIRED. The value is "http://tools.ietf.org/html/rfc6749#section-5.2".

5. IANA Considerations

5.1. Link Type Registration

Pursuant to [RFC5988], the following link type registrations [[will be]] registered by mail to link-relations@ietf.org.

5.1.1. OAuth 2 Registrations

The secition 3 of the OAuth 2.0 Authorization Framework [RFC6749] defines two endpoints that may be discovered through this specification. These are the user Authorization Endpoint and the Token Endpoint.

5.1.1.1. Authorization Endpoint

- Relation Name: oauth2-authorize
- Descritpion: An OAuth 2.0 Authorization Endpoint specified in section 3.1 of [RFC6749]
- Reference: [RFC6749]

5.1.1.2. Token Endpoint

- Relation Name: oauth2-token
- Description: An OAuth 2.0 Token Endpoint specified in section 3.2 of [RFC6749].
- Refeence: [RFC6749]

6. Security Considerations

6.1. href tampering

Unless integrity protected channel is used, an attacker may be able to tamper the value of the href thereby causing the receiver of the JSON response to send a request to the URL under the attacker's control with potentially confidential information contained in the parameters. To mitigate this risk, an integrity protected channel such as TLS protected channel should be used.

7. Acknowledgements

This specification borrows heavily from [HAL]. The Link type registration is taken from [oauth-lrdd].

[todo]

8. Document History

-02

- Added Mike Kelly as an author.
- xref fix.
- Introduced "operations" as in draft-ietf-scim-api-00#section-3.5.
- Updated the informative reference to HAL.
- Added description to OAuth Token Endpoint hrefs.
- Added content-type to the example.
- Added Area and Working Group.

-01

- Some format changes, reference fix, and typo fixes.
- Changed 'items' to 'elements' to match the JSON terminology.

-00

- Initial Draft

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC5988] Nottingham, M., "Web Linking", RFC 5988, October 2010.

[RFC6570] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", RFC 6570, March 2012.

[RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.

9.2. Informational References

[HAL] Kelly, M., "JSON Hypermedia API Language", February 2013.

[RFC4627] Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", RFC 4627, July 2006.

[oauth-lrdd] Mills, W., "Link Type Registrations for OAuth 2", October 2012.

Authors' Addresses

Mike Kelly
Stateless

Email: mike@stateless.co

Nat Sakimura (editor)
Nomura Research Institute

Email: sakimura@gmail.com