OAuth Use Cases

draft-zeltsan-use-cases-oauth-00

Abstract

This document lists the OAuth use cases. The document’s objective is to identify the use cases that will be a base for deriving the OAuth requirements. The provided list is based on the Internet-Drafts and other OAuth-related documents that have been discussed by the participants of the oauth working group.

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 November 19, 2010.

Copyright Notice

Copyright (c) 2010 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 (http://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 ................................................. 3
2. Notational Conventions ...................................... 3
3. OAuth use cases .............................................. 3
   3.1. Use cases of the draft The OAuth 2.0 Protocol ........ 3
       3.1.1. User-Agent Flow .................................. 3
       3.1.2. Web Server Flow .................................. 4
       3.1.3. Device Flow ...................................... 5
       3.1.4. Username and Password Flow ....................... 6
       3.1.5. Client Credentials Flow ........................... 7
       3.1.6. Assertion Flow ................................... 7
   3.2. Use case of the draft Using OAuth for Recursive Delegation ............................................. 8
   3.3. Use cases of the draft Evaluating OAuth’s suitability for SIP authentication ............................. 9
       3.3.1. Establishment of an MSRP session ................. 9
       3.3.2. Gateway for browser-based VoIP applets ........ 10
   3.4. Use case Access Token Exchange ......................... 11
   3.5. Use case for signature with asymmetric secret .... 12
4. Authors of the use cases ..................................... 13
5. Security Considerations ..................................... 13
6. IANA Considerations ......................................... 13
7. Acknowledgements ........................................... 13
8. References .................................................. 13
   8.1. Normative References ................................. 13
   8.2. Informative References ............................... 14
Author’s Address ............................................... 14
1. Introduction

The need for documenting the OAuth use cases was discussed at the oauth WG virtual meetings, on the group’s mailing list, and at the IETF 77. This Internet-Draft describes such use cases. The objective of the draft is to initiate discussion that will lead to defining a set of the use cases that the OAuth specifications should support. The following section provides the abbreviated descriptions of the use cases. The descriptions include the pointers to the sources of the use cases.

2. Notational Conventions

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 [RFC2119].

3. OAuth use cases

This section lists the major use cases that have been discussed by the oauth WG.

3.1. Use cases of the draft The OAuth 2.0 Protocol

[draft-ietf-oauth-v2]

3.1.1. User-Agent Flow

Description:
A Client application residing in a user-agent accesses protected resource on user’s behalf

(This use case covers the one specified in section Rich App Profile of [draft-hardt-oauth])

Pre-conditions:

- Client is an application residing in a user-agent, typically implemented in a browser using a scripting language such as JavaScript
- Client is capable of interacting with the user-agent
- Client provides redirection URL to authorization server for redirecting the user-agent after authorization by the end user
o Client’s authentication by authorization server is based on the user-agent’s same origin policy (client cannot keep client secret confidential)

o Client does not need to be an HTTP server

o End user’s registration with authorization Server for identification and authentication is required

o End user’s authentication to authorization server is required but not specified

o User-agent is a web browser capable of executing a server-provided script to extract an access token from the response of the authorization server. User-agent passes the access token to the client

Involved entities:

o Resource owner: end user

o Web Server that stores a protected resource

o Authorization server

o Client

o User-agent

3.1.2. Web Server Flow

Description:
A client, which is a part of the web server application, accesses protected resource on user’s behalf. User does not need to share authentication credentials with the client.

(This use case covers the one specified in section Web App Profile of [draft-hardt-oauth] and in [RFC5849])

Pre-conditions:

o Client is a Web application capable of interacting with the user-agent (typically a browser) and receiving requests from the authorization server (i.e., capable of acting as HTTP server)

o Client’s registration with authorization server for identification and authentication is required
o Client provides its callback URL to the authorization server

o Client authentication to authorization server is required; it is based on a client identifier and shared secret

o End user’s registration with authorization server for identification and authentication is required

o End user’s authentication to authorization server is required but not specified

Involved entities:

o Resource owner: end user

o Server that stores a protected resource

o Authorization server

o Client

3.1.3. Device Flow

Description:
A Client executing on a device that does not have an easy data-entry method accesses a protected resource on user’s behalf.

(This use case covers the one described in the message submitted to OAUTH-WG list by Brent Goldman, Facebook on 3/11/2010)

Pre-conditions:

o A Client is running on a device that does not have an easy data-entry method (e.g. game consoles or media server). The client is able to poll the authorization server until the end user completes the authorization process

o Client’s registration with authorization server for identification is required

o Client’s interaction with user-agent requires end user’s involvement (i.e., manual data entry by the end user)

o End user’s registration with authorization server for identification and authentication is required

o End user’s authentication to authorization server is required but not specified
The End user has a separate access to a user-agent on another computer or device (e.g. home computer, a laptop, or a smartphone)

Involved entities:

- Resource owner: end user
- Server that stores a protected resource
- Authorization server
- Client
- User-agent

### 3.1.4. Username and Password Flow

**Description:**
A trusted client accesses the end user’s resource on her or his behalf using the end user’s username and password. The client does not need to store the end user’s username and password permanently.

(This use case covers those specified in [draft-dehora-farrell-oauth-accesstoken-creds] and in section Client Account and Password Profile of [draft-hardt-oauth])

**Pre-conditions:**

- A client is capable of obtaining the username and password from the end user
- Client’s registration with authorization server for identification and authentication is required
- Client’s authentication to authorization server is based on client identifier and client secret
- Client’s authorization is based on its knowledge of the end user’s username and password
- Client is not required to be an HTTP server (i.e., does not need to do HTTP redirection)
- End user’s registration with authorization server for identification and authentication is required
- End user’s authentication to authorization server is based on username and password
3.1.5. Client Credentials Flow

Description:
A client that is not acting on behalf of a separate resource owner accesses a protected resource

Pre-conditions:

- Client has established a client identifier and a client secret (that assumed to be of high-entropy since it does not need to be memorized by an end user)
- Client’s registration with authorization server for identification and authentication is required
- Client’s authentication to the authorization server is based on the client identifier and shared secret
- End user involvement is not needed

Involved entities:
- Server that stores a protected resource
- Authorization server
- Client

3.1.6. Assertion Flow

Description:
A client exchanges an existing assertion (e.g., SAML assertion) for an access token, which it uses to access the protected resource. The
client could be the resource owner or could be acting on the resource owner’s behalf

(This use case covers the one specified in section Assertion Profile of [draft-hardt-oauth])

Pre-conditions:

- Client has obtained an assertion from an asserting entity
- Client’s registration with authorization server is not required
- End user involvement is not needed
- Authorization server is able to validate the assertion issued by the asserting entity

Involved entities:

- Asserting entity
- Authorization server
- Client
- Server that stores a protected resource

3.2. Use case of the draft Using OAuth for Recursive Delegation

[draft-vrancken-oauth-redelegation]

3.2.1. Content sharing

Description:
Enable organizing and sharing information among the end users. For example, a Web user (resource owner) can grant data access to a pre-defined set of users. This can be done with the use of a special OAuth client - content manager - which serves as a proxy between the end-users and the Web servers that host the resources related to the project. The content manager allows a user (the owner of the resources) to specify a set of the resources related to a project (e.g., by tagging) and a set of the clients and their access rights in respect to the resources. The content manager may also enable searching of the related materials.

Pre-conditions:
First client is a client that has been authorized by the resource owner - an end user - to grant access rights for the protected resource to the second client

Second client is a client that wishes to get authorization of the first client to get access to the protected resource

Clients’ registration with Web server for identification and authentication is required

Clients’ authentication to authorization (Web) server is required and based on their identifiers and shared secrets

Clients need to be the HTTP servers (i.e., be capable of HTTP redirection)

End user’s authentication to Web server is required but not specified

Involved entities:

- Resource owner: end user
- First client
- Second client
- Server that stores a protected resource

3.3. Use cases of the draft Evaluating OAuth’s suitability for SIP authentication

[draft-beck-oauth-sip-eval]

3.3.1. Establishment of an MSRP session

Description:
A client that implements text chat using asynchronous HTTP requests accesses on behalf of an end user her or his protected resources on a SIP server

Pre-conditions:

- The resource owner is an end user that has a SIP account and uses chat service that is implemented as asynchronous HTTP requests

- Protected resource is data associated with end user’s SIP account on her or his SIP server, which is capable to participate in OAuth
A client is a Web application that implements text chat service using asynchronous HTTP requests, implements also the OAuth client, and is capable of translating the chat-related HTTP requests into SIP and MSRP messages.

Client’s registration with authorization server for identification and authentication is required.

Client’s authentication by authorization server is required.

Client is an HTTP server (i.e., it is capable of redirecting end user’s user-agent).

Involved entities:

- Resource owner
- SIP server that stores protected resource
- Client

3.3.2. Gateway for browser-based VoIP applets

Description:
A client that implements a VoIP client as a browser applet accesses the end user’s SIP resources on her or his SIP server and on the end user’s behalf.

Pre-conditions:

- Resource owner is an end user that has a SIP account and uses VoIP Web service that is implemented as a browser applet.
- SIP server that is capable to participate in OAuth exchange and stores data associated with end user’s SIP account.
- A client is a Web application that implements VoIP as a browser applet, implements also the OAuth client, and is capable of translating the Web applets messages into SIP and RTP.
- Client’s registration with authorization server for identification and authentication is required.
- Client’ authentication by authorization server is required.
3.4. Use case Access Token Exchange

(submitted to OAuth-WG list by Torsten Lodderstedt, Deutsche Telekom on 4/6/2010)

Description:
A client performs on an end user’s behalf multi-layered services (i.e., the services where one service requires execution of another service that requires authentication)

Pre-conditions:

- Authorization server is capable of issuing access tokens to the client and multiple and dependent Web services
- Client’s authentication by authorization server is required
- Authorization of (N-1)th Web service for execution of Nth service is based on the content of the access token obtained by the client
- Requirements to the registration of the client and the dependent Web services are to be defined
- Requirements to the end user’s registration with client are to be defined
- Requirements for the end user’s authentication by the client are to be defined

Involved entities:

- Resource owner
- Client that initiates the first Web service
- First Web service that requires execution of the second Web service
Nth Web service that is a service that requires execution of the (N-1)th Web service

Authorization server

Servers that store the protected resources

3.5. Use case for signature with asymmetric secret

(submitted to OAUTH-WG list by Eran Hammer-Lahav, Yahoo! on 2/18/2010)

Description:
A client, that does not have a shared secret with a server storing a protected resource, accesses the protected resource on end user’s behalf

Pre-conditions:

o A client does not share a secret with a server that stores a protected resource, but has a public key

o client’s registration with authorization server for identification and authentication is not required

o Client’s authentication by the Web server is required and based on the public key cryptography

o Web Server that stores a protected resource has access to the client’s public key and associated information

o Requirements to the End user’s involvement are to be determined

o Requirements to the client being an HTTP server are to be determined

Involved entities:

o Resource owner: end User

o Client

o Server that stores a protected resource
4. Authors of the use cases

The major contributors of the use cases are as follows:

W. Beck, Deutsche Telekom AG
G. Brail, Sonoa Systems
B. de hOra
B. Eaton, Google
S. Farrell, NewBay Software
Y. Goland, Microsoft
B. Goldman, Facebook
E. Hammer-Lahav, Yahoo!
D. Hardt
R. Krikorian, Twitter
T. Lodderstedt, Deutsche Telekom
E. Maler, PayPal
D. Recordon, Facebook
L. Shepard, Facebook
A. Tom, Yahoo!
B. Vrancken, Alcatel-Lucent
Z. Zeltsan, Alcatel-Lucent

5. Security Considerations

TBD

6. IANA Considerations

This Internet Draft includes no request to IANA.

7. Acknowledgements

The author thanks Igor Faynberg and Hui-Lan Lu for their invaluable help with preparing this document.

8. References

8.1. Normative References

8.2. Informative References


Author’s Address

Zachary Zeltsan
Alcatel-Lucent
600 Mountain Avenue
Murray Hill, New Jersey
USA

Phone: +1 908 582 2359
Email: Zachary.Zeltsan@alcatel-lucent.com