OAuth Roadmap

Hannes Tschofenig, Derek Atkins
Good Progress with Our Documents

<table>
<thead>
<tr>
<th>Draft Reference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>draft-ietf-oauth-assertions-07</td>
<td>Assertion Framework for OAuth 2.0</td>
</tr>
<tr>
<td>draft-ietf-oauth-dyn-reg-01</td>
<td>OAuth Dynamic Client Registration Protocol</td>
</tr>
<tr>
<td>draft-ietf-oauth-json-web-token-05</td>
<td>JSON Web Token (JWT)</td>
</tr>
<tr>
<td>draft-ietf-oauth-jwt-bearer-03</td>
<td>JSON Web Token (JWT) Bearer Token Profiles for OAuth 2.0</td>
</tr>
<tr>
<td>draft-ietf-oauth-revocation-01</td>
<td>Token Revocation</td>
</tr>
<tr>
<td>draft-ietf-oauth-saml2-bearer-15</td>
<td>SAML 2.0 Bearer Assertion Profiles for OAuth 2.0</td>
</tr>
<tr>
<td>draft-ietf-oauth-use-cases-03</td>
<td>OAuth Use Cases</td>
</tr>
<tr>
<td>draft-ietf-oauth-v2-threatmodel-08</td>
<td>OAuth 2.0 Threat Model and Security Considerations</td>
</tr>
</tbody>
</table>
Specifications

• We are making progress in the working group.
• Where do we go next?
• Luckily we have various proposals sitting around.
UX


- Extensions to allow client to request particular user experience at authorization endpoint

- Partially incorporated into OIDC
  - Display (page, popup, etc)
  - Prompt (added by OIDC)
  - Preferred Locale (was “language”), in Request Object
JSON Based Request Object

- Authorization Request currently is done only by HTTP query parameters.
- Use of “request” param to send JWS version of the parameters.
- Use of “request_uri” param to send the reference to the JWS.
  - This is useful when complex authorization query is being issued.
- Make it possible to sign/sign+encrypt the request.
- Used in OpenID Connect and has been very stable for couple of years now.
  - [http://openid.net/specs/openid-connect-messages-1_0.html#OpenID_Request_Object](http://openid.net/specs/openid-connect-messages-1_0.html#OpenID_Request_Object)
- [draft-sakimura-oauth-requrl-03](https://tools.ietf.org/html/draft-sakimura-oauth-requrl-03)
Hyperlinked OAuth

- Insert JSON Hyperlink to OAuth responses
  - Provides metadata about resources where the token can be used
  - [http://nat.sakimura.org/2012/08/29/hal-](http://nat.sakimura.org/2012/08/29/hal-)

```json
{
  "_links": {
    "self": {
      "href": "/tokens?code=asdfasdfs"
    },
    "userinfo": {
      "href": "/userinfo{?id_token,scope,access_token,schema}",
      "Authorize": "{token_type} {access_token}",
      "templated": true
    }
  },
  "access_token": "SlAV32hkKG",
  "token_type": "Bearer",
  "refresh_token": "8xLOxBtZp8",
  "expires_in": 3600,
  "id_token": "eyJ0...NiJ9.eyJ1c...I6IjIifX0.DeWt4Qu...ZXso"
}
```
Chained Tokens

• In the wild implementations by DT, MITRE, AOL (sortof)
• Resource Server presents Access Token to Authorization Server in exchange for another Access Token to access a second Resource Server
  – With potentially reduced scope
Alternate Serialization

- Serialization of Token Endpoint response in XML and Form Parameters

```xml
<oauth type="object">
  <access_token type="string">2YotnFZFEjr1zCsicMWpAA</access_token>
  <token_type type="string">example</token_type>
  <expires_in type="number">3600</expires_in>
  <refresh_token type="string">tGzv3JOkF0XG5Qx2TIKWIA</refresh_token>
  <example_parameter type="string">example_value</example_parameter>
</oauth>
```
Client Instance

  – Expired 5/2011

• Human-readable instance-specific client meta information presented at Authorization Endpoint
  – One client_id, many instances

• Could be extended to machine-readable instance_id
Device Flow

  - Expired 1/2011
- Method for limited-input devices to get an access token & refresh token
  - Was carved out of OAuth2 Core
- Relies on backend polling and out-of-band secret conveyance
Multiple Tokens

• Don’t want the client to send the same access token to multiple resource servers, still want single access grant, reduce network round trips
• Method of getting multiple access tokens because different resource servers require different access tokens
  – Not method to get different types of tokens
  – Method to get distinct instances of one type of token
• Possible solutions:
  – Uber refresh token w/downscoped access tokens
  – Scope syntax to indicate need for different tokens
  – Multiple authorization codes (implemented by DT)
  – Token translation by OIDC
  – UMA’s permissions/scope stuff
Token Introspection

• Method for token holder to fetch information about token from the AS
  – Usually the RS talking to the AS
  – Usually protected by extra client credentials
• Existing implementations by Ping, MITRE, AOL
• http://www.ietf.org/mail-archive/web/oauth/current/msg08607.html
RS->AS connection

• Generic methods for RS know what a token is good for
  – Token introspection, structured tokens

• Methods for connecting an RS to an AS (and vice versa)
UMA

- UMA is fundamentally a use case
- Technology could be broken/distilled into multiple using components
  - Introduce the RS to the AS
    - Have to tell the AS which RS the Client wants to use
    - User has to authorize it
Implementations

• [http://oauth.net/code/](http://oauth.net/code/) lists implementations for OAuth 1.0, and OAuth 2.0.

• It does, however, not aim to verify whether the implementations actually relate to the OAuth 2.0 specifications (and how much).

• Is there value in listing open source reference implementations?
Interoperability Testing Events

• OpenID Connect does perform interop tests (see http://osis.idcommons.net/wiki/OC3:OpenID_Connect_Interop_3) but the scope is about OpenID Connect rather than OAuth itself.

• Other forms of tests are also possible. SCIM has also set-up test servers
  – http://www.simplecloud.info/#complianceTest
  – http://code.google.com/p/scimproxy/

• Would further interoperability tests improve the quality of OAuth 2.0 implementations?
Best Current Practices

• Various IIW discussions help to share best current practices but those do not get documented properly and reviewed.

• Examples:
  – ID tokens vs. Access Tokens
  – Embedded browser vs. custom URI schemes
  – Various security practices
Education and Information Sharing

• There is an increased interest to hear more about OAuth.

• Information sharing in form of presentations or articles.

• For examples:
  – Presentation to NIST related to their GreenButton initiative:
    http://collaborate.nist.gov/twiki-sggrid/bin/view/SmartGrid/OpenESPIDevelopmentProjectWalkthroughs
  – Article in German computer magazine by Torsten:
    http://www.heise.de/ix/inhalt/2012/10/4/

• Does it make sense to collect slides and other presentation material at a single place?
Next Steps

• Chairs will work with volunteers to prepare an interop event.

• We will work with volunteers to collect
  – Slides, papers, and other resources that will be linked via the main working group page.
  – Update WG Wiki page to list available open source implementations.

• We are going to encourage more dissemination efforts and will help to review.