

OAuth 2 Simplified

Sun, Jul 29, 2012 9:30am -07:00



90% Unlimited Downloads Choose from Over 300,000 Vectors, Graphics & Photos.

This post describes OAuth 2 in a simplified format to help developers and service providers implement the protocol.

The [OAuth 2 spec](#) itself leaves many decisions up to the implementor. Instead of describing all possible decisions that need to be made to successfully implement OAuth 2, this post makes decisions that are appropriate for most implementations.

Table of Contents

- [Roles](#): Applications, APIs and Users
- [Creating an App](#)
- [Authorization](#): Obtaining an access token
 - [Web Server Apps](#)
 - [Browser-Based Apps](#)
 - [Mobile Apps](#)
 - [Others](#)
- [Making Authenticated Requests](#)
- [Differences from OAuth 1.0](#)
 - [Authentication and Signatures](#)
 - [User Experience and Alternative Authorization Flows](#)
 - [Performance at Scale](#)
- [Resources](#)

Roles

The Third-Party Application: "Client"

The client is the application that is attempting to get access to the user's account. It needs to get permission from the user before it can do so.

The API: "Resource Server"

The resource server is the API server used to access the user's information.

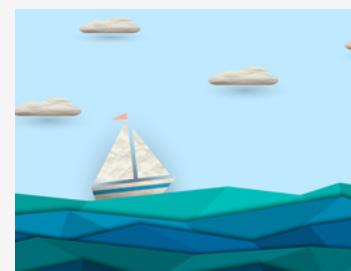
The User: "Resource Owner"

The resource owner is the person who is giving access to some portion of their account.

Creating an App

Before you can begin the OAuth process, you must first register a new app with the service. When registering a new app, you usually register basic information such as application name, website, a logo, etc. In addition,

ads via Carbon



90% Unlimited
Downloads Choose from
Over 300,000 Vectors,
Graphics & Photos.

ads via Carbon

you must register a redirect URI to be used for redirecting users to for web server, browser-based, or mobile apps.

Redirect URIs

The service will only redirect users to a registered URI, which helps prevent some attacks. Any HTTP redirect URIs must be protected with TLS security, so the service will only redirect to URIs beginning with "https". This prevents tokens from being intercepted during the authorization process.

Client ID and Secret

After registering your app, you will receive a client ID and a client secret. The client ID is considered public information, and is used to build login URLs, or included in Javascript source code on a page. The client secret **must** be kept confidential. If a deployed app cannot keep the secret confidential, such as Javascript or native apps, then the secret is not used.

Authorization

The first step of OAuth 2 is to get authorization from the user. For browser-based or mobile apps, this is usually accomplished by displaying an interface provided by the service to the user.

OAuth 2 provides several "grant types" for different use cases. The grant types defined are:

- **Authorization Code** for apps running on a [web server](#)
- **Implicit** for [browser-based](#) or [mobile apps](#)
- **Password** for logging in with a [username and password](#)
- **Client credentials** for [application access](#)

Each use case is described in detail below.

Web Server Apps

Web server apps are the most common type of application you encounter when dealing with OAuth servers. Web apps are written in a server-side language and run on a server where the source code of the application is not available to the public.

Authorization

Create a "Log In" link sending the user to:

```
https://oauth2server.com/auth?response_type=code&
  client_id=CLIENT_ID&redirect_uri=REDIRECT_URI&scope=photos
```

The user sees the authorization prompt



An application would like to connect to your account

The app **Sample App** by Aaron Parecki would like the ability to access your basic information and photos.

Allow **Sample App** access?

Deny

Allow

If the user clicks "Allow," the service redirects the user back to your site with an auth code

```
https://oauth2client.com/cb?code=AUTH_CODE_HERE
```

Your server exchanges the auth code for an access token

```
POST https://api.oauth2server.com/token
grant_type=authorization_code&
code=AUTH_CODE_HERE&
redirect_uri=REDIRECT_URI&
client_id=CLIENT_ID&
client_secret=CLIENT_SECRET
```

The server replies with an access token

```
{
  "access_token": "RsT5OjbzRn430zqMLgV3Ia"
}
```

or if there was an error

```
{
  "error": "invalid_request"
}
```

Security: Note that the service must require apps to pre-register their redirect URIs.

Browser-Based Apps

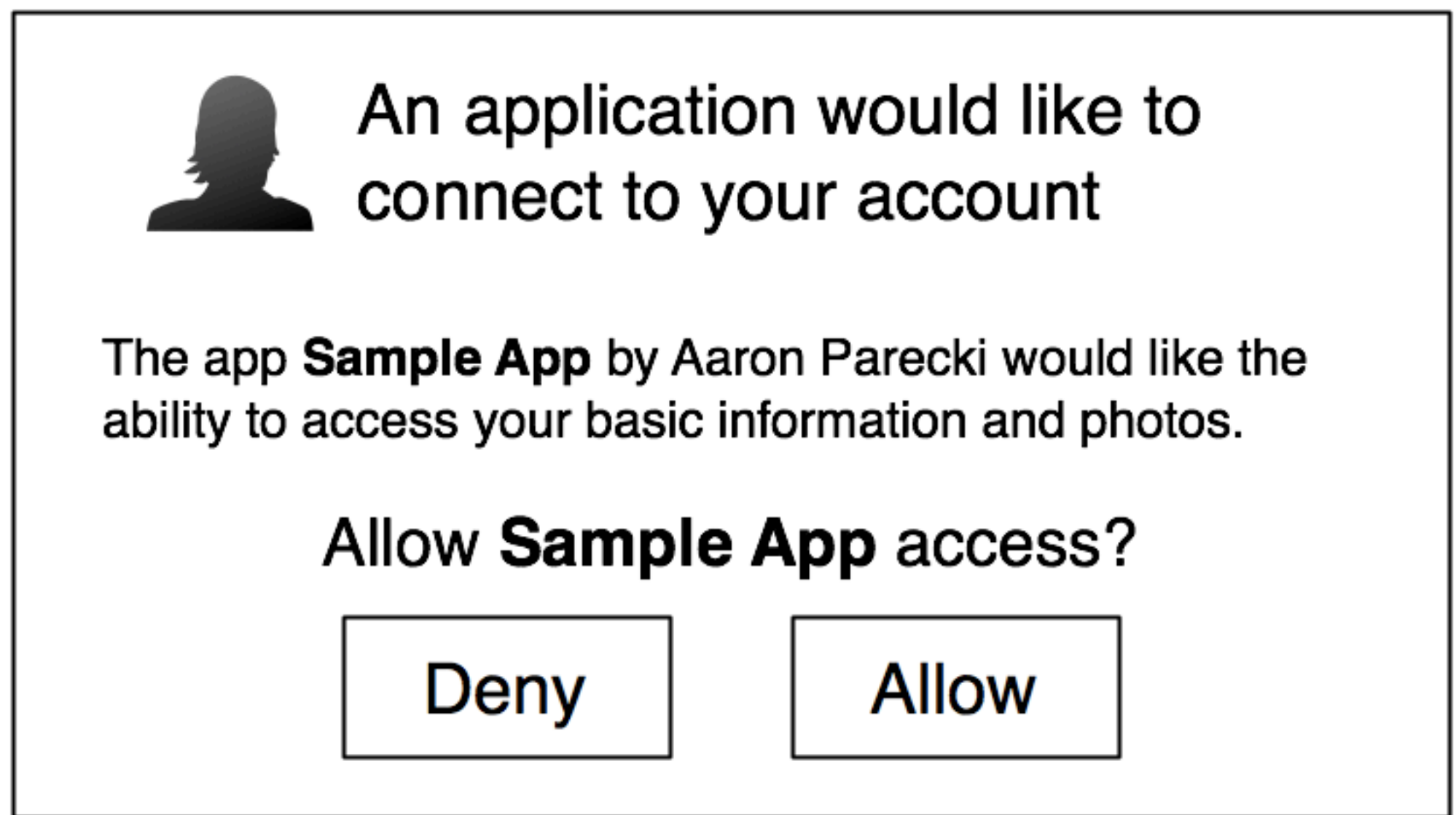
Browser-based apps run entirely in the browser after loading the source code from a web page. Since the entire source code is available to the browser, they cannot maintain the confidentiality of their client secret, so the secret is not used in this case.

Authorization

Create a "Log In" link sending the user to:

```
https://oauth2server.com/auth?response_type=token&
  client_id=CLIENT_ID&redirect_uri=REDIRECT_URI&scope=photos
```

The user sees the authorization prompt



If the user clicks "Allow," the service redirects the user back to your site with an access token

```
https://oauth2client.com/cb#token=ACCESS_TOKEN
```

That's it, there's no other steps! At this point, some Javascript code can pull out the access token from the fragment (the part after the #) and begin making API requests.

If there was an error, you will instead receive an error code in the URI fragment, such as:

```
https://oauth2client.com/cb#error=access_denied
```

Mobile Apps

Like browser-based apps, mobile apps also cannot maintain the confidentiality of their client secret. Because of this, mobile apps must also use an OAuth flow that does not require a client secret.

Authorization

Create a "Log in" button sending the user to either the native app of the service on the phone, or a mobile web page for the service. On iPhone, apps can register a custom URI protocol such as "facebook://" so the native Facebook app is launched whenever a URL with that protocol is visited. On Android, apps can register URL matching patterns which will launch the native app if a URL matching the pattern is visited.

iPhone

If the user has the native Facebook app installed, direct them to the following URL:

```
fbauth2://authorize?response_type=token&client_id=CLIENT_ID
&redirect_uri=REDIRECT_URI&scope=email
```

In this case, your redirect URI looks like `fb00000000://authorize` where the protocol is "fb" followed by your app's client ID. This means your app must be registered to open URLs with that protocol.

Android or Others

If the user does not have the Facebook iPhone app, or for other devices, you can launch a mobile browser to the standard web authorization URL.

```
https://facebook.com/dialog/oauth?response_type=token
&client_id=CLIENT_ID&redirect_uri=REDIRECT_URI&scope=email
```

The user will see the authorization prompt



After clicking "Okay", the user will be redirected back to your application by a URL like

```
fb00000000://authorize#token=ACCESS_TOKEN
```

Your mobile application can parse out the access token from the URI and begin using it to make API requests.

Others

Password

OAuth 2 also provides a "password" grant type which can be used to exchange a username and password for an access token directly. Since this obviously requires the application to collect the user's password, it should only be used by apps created by the service itself. For example, the native Twitter app could use this grant type to log in on mobile or desktop apps.

To use the password grant type, simply make a POST request like the following:

```
POST https://api.oauth2server.com/token
grant_type=password&
username=USERNAME&
password=PASSWORD&
client_id=CLIENT_ID
```

The server replies with an access token in the same format as the other grant types.

Note, the client secret is not included here under the assumption that most of the use cases for password grants will be mobile or desktop apps, where the secret cannot be protected.

Application access

In some cases, applications may wish to update their own information such as their website URL or application icon, or they may wish to get statistics about the users of the app. In this case, applications need a way to get an access token for their own account, outside the context of any specific user. OAuth provides the `client_credentials` grant type for this purpose.

To use the client credentials grant type, make a POST request like the following:

```
POST https://api.oauth2server.com/token
grant_type=client_credentials&
client_id=CLIENT_ID&
client_secret=CLIENT_SECRET
```

You will get an access token response in the same format as the other grant types.

Making Authenticated Requests

Now that you have an access token, you can make requests to the API. You can quickly make an API request using cURL as follows:

```
curl -H "Authorization: Bearer RsT5OjbzRn430zqMLgV3Ia" \
https://api.oauth2server.com/1/me
```

That's it! Make sure you always send requests over HTTPS and never ignore invalid certificates. HTTPS is the only thing protecting requests from being intercepted or modified.

Differences from OAuth 1.0

OAuth 1.0 was largely based on existing proprietary protocols such as Flickr's "FlickrAuth" and Google's "AuthSub". The result represented the best solution based on actual implementation experience. However, after several years of working with the protocol, the community learned enough to rethink and improve the protocol in three main areas where OAuth 1.0 proved limited or confusing:

Authentication and Signatures

The majority of developers' confusion and annoyance with OAuth 1.0 was due to the cryptographic requirements of signing requests with the client ID and secret. Losing the ability to easily copy and paste cURL examples made it much more difficult to get started quickly.

OAuth 2 recognizes this difficulty and replaces signatures with requiring HTTPS for all communications between browsers, clients and the API.

User Experience and Alternative Authorization Flows

OAuth includes two main parts, obtaining an access token, and using the access token to make requests. OAuth 1.0 works best for desktop web browsers, but fails to provide a good user experience for native desktop and mobile apps or alternative devices such as game or TV consoles.

OAuth 2 supports a better user experience for native applications, and supports extending the protocol to provide compatibility with future device requirements.

Performance at Scale

As larger providers started to use OAuth 1.0, the community quickly realized the protocol does not scale well. Many steps require state management and temporary credentials, which require shared storage and are difficult to synchronize across data centers. OAuth 1.0 also requires that the API server has access to the application's ID and secret, which often breaks the architecture of most large providers where the authorization server and API servers are completely separate.


OAuth 2 supports the separation of the roles of obtaining user authorization and handling API calls. Larger providers needing this scalability are free to implement it as such, and smaller providers can use the same server for both roles if they wish.

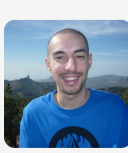
Resources


Credit: Some content adapted from hueniverse.com.


More information is available on oauth.net


Sun, Jul 29, 2012 9:30am -07:00 [#oauth](#) [#oauth2](#) [#standards](#) [#web](#) [#authentication](#)
6 replies 76 mentions

- 

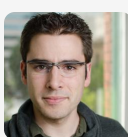
[Avdi Grimm](#) avdi.org
As guides to OAuth go, this one doesn't suck pocket.co/sckvF
Sun, Jun 8, 2014 10:38am -07:00 (via brid-gy.appspot.com)
- 

[David Montesdeoca](#) twitter.com/backpackerhh
#OAuth 2 Simplified: aaronparecki.com/articles/2012/...
Sun, Jun 8, 2014 10:46am -07:00 (via brid-gy.appspot.com)
- 

[channa ly](#) www.channaly.info
@aaronpk yes sure. it is really helpful thanks for sharing that.
Wed, Sep 3, 2014 9:34am -07:00 (via brid-gy.appspot.com)
- 

[Aaron Parecki](#) aaronparecki.com
@luxagraf OAuth 2 is simple enough you don't need a library, I have a simple guide here <https://aaronparecki.com/2012/07/29/2/oauth2-simplified>
Mon, Jul 11, 2016 12:17pm -07:00
- 

[Scott Gilbertson](#) luxagraf.net
@aaronpk It's simple enough if someone is around for redirects and auth codes, harder when it's an automated script.
Mon, Jul 11, 2016 12:28pm -07:00 (via brid-gy.appspot.com)



Aaron Parecki aaronparecki.com

@luxagraf Yeah, you'll want to set up a page that can generate an access token for you. Some services like GitHub provide this built in.

Mon, Jul 11, 2016 12:31pm -07:00 (via brid-gy.appspot.com)

Other Mentions

[stackoverflow.com](#)

Thu, Feb 18, 2016 1:06pm -07:00

[stackoverflow.com](#)

Mon, Feb 22, 2016 1:30am -07:00

[www.sitepoint.com](#)

Fri, Apr 15, 2016 6:32am -07:00

[stackoverflow.com](#)

Thu, May 12, 2016 10:18am -07:00

[4gophers.ru](#)

Fri, Jun 24, 2016 2:54pm -07:00

[www.sdk.cn](#)

Mon, Apr 18, 2016 8:04pm -07:00

[stackoverflow.com](#)

Sun, Oct 2, 2016 10:47am -07:00

[security.stackexchange.com](#)

Wed, Jun 15, 2016 12:11am -07:00

[stackoverflow.com](#)

Wed, Jul 20, 2016 7:52pm -07:00

Posted by [cristanmeijer](#) blogmobile.itude.com/author/cristanmeijer

Checklist for building an API

Apr 8

Posted by cristanmeijer ...

Tue, Apr 8, 2014 1:29am -07:00

[chenlu4.sinaapp.com](#)

Programming an OAuth2 Client App in PHP

参考: http://20missionglass.tumblr.com/post/60787835108/programming-an-oauth2-client-app-in-php

OAuth2 Simplified: http://aaronparecki.com/articles/2012/07/29/1/oauth2-simplified ...

Wed, May 7, 2014 3:57am -07:00

[www.kaleidos.net](#)

[Stateless Authentication with api rest](#)

When we talk about authentication for api rest, almost everyone tends to think about oauth1 or oauth2 and their variants defined by service providers. It's true that there also other auth systems such as token, openid, etc, but they are not as widely used in comparison with oauth.

What do you think about them? Are they all truly stateless?

How authentication works? ...

Tue, Jun 17, 2014 7:32am -07:00



Aaron Parecki aaronparecki.com

@rob_levine I'm curious what your current issues are! Would love to make sure it's addressed in the book. Also check out my basic OAuth 2.0 tutorial here if you haven't yet http://aaronparecki.com/articles/2012/07/29/1/oauth2-simplified

Wed, Jun 25, 2014 1:37pm -07:00

[mesotheliomacare.xyz](#)

[Oauth Introduction](#)

introduction of a touch environment with Premiere Pro CC ... Easy Access: Primetime's support for the OAUTH 2.0 protocol makes it incredibly easy for consumers to access their favorite pay-TV content. Pay-TV providers can enable frictionless TV ...

Send Large Files Email File too large to send via email? Send File makes sending your big files easy. Send photos to family, music to friends, documents to businesses and more. MUSKOGEE, Okla. — A silver alert has been issued for 78-year-old ...

Fri, Jun 27, 2014 6:05am -07:00

www.zescience.com

[As an Oauth2 service provider, how do I validate which app is passing me the user token?](#)

I have a pretty good handle on Oauth2 having connected to several services as a client. Now, I'm working on implementing a service that clients will connect to with Oauth2.

It seems wrong to just accept any valid user access token without validating that the app that is giving it to me is in fact the app the I first gave the token to. In the Oauth2 spec, and in several articles including this one, the topic isn't discussed. Is there a standard way for the client to tell me who they are, ...

Wed, Jul 30, 2014 10:51am -07:00

www.pearltrees.com

Wed, Aug 20, 2014 8:37am -07:00



[Aaron Parecki](#) aaronparecki.com

@channaly Thanks! You might also like my written version of the talk: <http://aaronparecki.com/articles/2012/07/29/1/oauth2-simplified#oauth2>

Wed, Sep 3, 2014 9:31am -07:00

doanswers.biz

[@fontface not operative on server though on localhost](#)

Q1.@fontface not operative on server though on localhost

I dont know since my fontface isn't working. It works on my web developer though when i uploaded it to my web server, it does not arrangement notwithstanding same formula and office we attempted controlling firebug to wizz in a problem though to not avail.

@font-face { font-family: 'FuturaCondensedMedium'; src: url ; src: local , url format , url format , url format ; } physique { background: #ffffff; ...

Mon, Sep 8, 2014 12:07pm -07:00

curiouser.cheshireeng.com

Digression: Using OAuth 2.0 at WordPress.com

10

Wednesday ...

Wed, Sep 10, 2014 11:19am -07:00

ziesman.id-gyan.biz

[ASP.NET MVC – Forms Auth vs OAuth 2.0](#)

(adsbygoogle = window.adsbygoogle || []).push({});

Q1.ASP.NET MVC – Forms Auth vs OAuth 2.0

I am exploring a luck of controlling OAuth 2.0 in destiny projects.

Wed, Sep 10, 2014 9:59pm -07:00

mintknow.biz

[oAuth2 server. Should we have unparalleled or opposite endpoints for opposite extend type](#)

(adsbygoogle = window.adsbygoogle || []).push({});

Q1.oAuth2 server. Should we have unparalleled or opposite endpoints for opposite extend type

According a essay oauth2 Simplified it could be 4 extend form for a oAuth server.

Fri, Sep 12, 2014 1:51pm -07:00

inhumanbean.com

OAuth 2 Simplified – Aaron Parecki

Tue, Nov 11, 2014 9:34am -07:00

raymondhlee.wordpress.com

Implementing OAuth2 with Spring Security

December 21, 2014

Leave a comment ...

Sat, Dec 20, 2014 8:10pm -07:00

www.quora.com

Fri, Feb 6, 2015 9:30am -07:00

www.noblebuffalo.com

[Portico API](#)

Sidwell’s flagship web product, Portico, is growing in so many ways. In additional to the original Silverlight control, I’ve developed a mobile website, and I am about to develop a whole application containing several “widgets” using ESRI’s Web Application Builder. We also have an Advanced Print Preview and some custom reports that access the backend. Portico has become so much more than a website — it’s a development platform — and I certainly will not predict the next ...

Mon, Mar 2, 2015 10:51am -07:00

dluat.com

[Session integration, is this approach secure?](#)

(adsbygoogle = window.adsbygoogle || []).push({});

A user logs in using default Laravel authentication, which puts an encrypted cookie in the browser, and saves the session in the database. The user moves to a classic asp page, where I check the cookie value, get the hash, and call the laravel app back passing the session id hash. ...

Sun, Apr 19, 2015 11:31pm -07:00

freemymind.wordpress.com

[OAuth 2.0 for Native Mobile Apps and Browser Apps](#)

There’s great deal of confusion when it comes to implementing API authorization using OAuth 2.0 The reasons are varied partly from the different options the specification itself provides and partly due to confusion of implementing it on a Mobile App. This being the case with myself, I thought of writing here on how to approach to make a selection based on your requirements. A basic point to understand is OAuth 2.0 doesn’t specify Authentication. It is actually an Authorization framework. ...

Wed, Apr 29, 2015 10:16am -07:00

7wins.eu

Wed, Apr 29, 2015 11:23pm -07:00

nodejs.clbo.dk

[#10 Authendication / Authorization](#)

Time: November 5, 2015 (week 45) 08:30 – 13:20
Morbi tempor lectus at nulla varius, vel consequat ex pellentesque. Vestibulum suscipit consequat erat, sed ullamcorper enim suscipit vel. Quisque tristique consectetur mauris at fringilla. Vivamus porta tristique mauris non vestibulum. Sed nisl arcu, pulvinar malesuada dui et, lobortis bibendum dolor. Duis aliquam viverra orci in eleifend. Nam magna arcu, auctor sed fringilla et, finibus non quam. Vivamus tempus enim sapien, eu accumsan dui ...

Sun, May 3, 2015 4:50pm -07:00

onehundred15.wordpress.com

Fri, Jun 19, 2015 9:26pm -07:00

dexpage.com

Thu, Jun 25, 2015 9:47am -07:00

dexpage.com

Fri, Jul 3, 2015 3:03am -07:00

dexpage.com

Wed, Jul 8, 2015 5:06pm -07:00

dexpage.com

Wed, Jul 8, 2015 7:03pm -07:00

unity3dassets.com

Mon, Jul 13, 2015 4:15pm -07:00

dexpage.com

Wed, Jul 15, 2015 2:19pm -07:00

fewfice.com

Wed, Aug 26, 2015 1:58am -07:00

blog.explodingads.com

1 – Oauth2 simplifiedBy blog.explodingads.com on August 28, 2015 in Startups

By [pietromenna](#)
1 point, 0 comments ...
Fri, Aug 28, 2015 9:37am -07:00

[emoo www.emoo.eu/author/emoo](http://www.emoo.eu/author/emoo)

[How authorization works with APIs](#)

Recently I've been busy engaged on content material for some API documentation workshops. One of many subjects I needed to cowl within the workshop is how authentication and authorization work with APIs. That is in all probability one of many extra difficult features of APIs, so I'd wish to get as a lot suggestions as potential about this part.

Authentication and authorization overview

Earlier than customers could make requests together with your API, they'll often have to register for some ...

Sat, Sep 5, 2015 9:26am -07:00



[Tiffany Lien](#) www.linkedin.com/in/tiffanylien

The most straight-forward description of OAuth: ow.ly/Sfo6i #tech

Tue, Sep 15, 2015 11:50am -07:00 (via [brid-gy.appspot.com](#))



[Joshua Ellis](#) www.jellis.com.au

The most simplistic #oauth2 explanation I've seen by @aaronpk aaronparecki.com/articles/2012/... Awesome job.

Wed, Sep 23, 2015 3:11am -07:00 (via [brid-gy.appspot.com](#))



[れこ](#) leko.jp

読了 OAuth 2 Simplified ift.tt/1FqFICu ift.tt/1Ffge0c

Wed, Sep 23, 2015 7:50am -07:00 (via [brid-gy.appspot.com](#))



[Amir S](#) twitter.com/momenticus

Brief and straight forward explanation of oauth-2: aaronparecki.com/articles/2012/...

Thu, Oct 15, 2015 2:58pm -07:00 (via [brid-gy.appspot.com](#))



[Alex Maldonado](#) twitter.com/OOCoder

Excellent explanation of OAuth2 protocol and the flow details. A better job than Google/Facebook IMHO :). @aaronpk buff.ly/1MubTVV

Tue, Oct 27, 2015 2:39pm -07:00 (via [brid-gy.appspot.com](#))

[Oleg Dulin](#) cdn.wikinewstech.com/author/oleg-dulin

[IDG Contributor Network: OAuth 2.0: The protocol at the center of the universe](#)

Social Privacy Authentication

Most apps today require authentication.

Credit: Thomas Ulrich / Pixabay ...

Tue, Nov 24, 2015 10:23am -07:00

[Oleg Dulin](#) cdn.wikinewstech.com/author/oleg-dulin

[IDG Contributor Network: OAuth 2.0: The protocol at the center of the universe](#)

Social Privacy Authentication

Most apps today require authentication.

Credit: Thomas Ulrich / Pixabay ...

Wed, Nov 25, 2015 2:23am -07:00

[Oleg Dulin](#) cdn.wikinewstech.com/author/oleg-dulin

[IDG Contributor Network: OAuth 2.0: The protocol at the center of the universe](#)

Social Privacy Authentication

Most apps today require authentication.

Credit: Thomas Ulrich / Pixabay ...

Wed, Nov 25, 2015 3:17am -07:00

questions.techjaffa.info

[As an OAuth2 service provider, how do I validate which app is passing me the user token?](#)

Question In ,

I have a pretty good handle on OAuth2 having connected to several services as a client. Now, I'm working on implementing a service that clients will connect to with OAuth2.

It seems wrong to just accept any valid user access token without validating that the app that is giving it to me is in fact the app that I first gave the token to. In the OAuth2 spec, and in several articles including this one, the topic isn't discussed. Is there a standard way for the client to tell ...

Thu, Nov 26, 2015 6:33am -07:00

questions.techjaffa.info

[How do I implement an OAuth2 Authorization Code Flow in Web Api using OWIN Middleware?](#)


Question In ,

I'm trying to create a simple proof of concept OAuth enabled application but am stuck on the authorization code implementation.

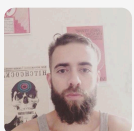
Everywhere I read seems like it goes in one way or another, never actually using the authorization code flow. I've been using the following resources for information:

<https://tools.ietf.org/html/draft-ietf-oauth-v2-31> ...


Thu, Nov 26, 2015 3:17pm -07:00



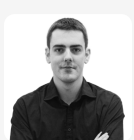
[M@ matthew-sinclair.com](#)
“OAuth 2 Simplified” [bit.ly/1LGJU2h](#) #bookmarked
Sun, Nov 29, 2015 9:21am -07:00 (via [brid-gy.appspot.com](#))




[literally an egg yoshuawuyts.com](#)
If you want to know more about OAuth, this is probably the article you're looking for - [aaronparecki.com/articles/2012/...](#)
Tue, Dec 15, 2015 6:22pm -07:00 (via [brid-gy.appspot.com](#))



[Hernan Marano about.me/herchugm](#)
Finally, an introduction to #oauth that can be understood [aaronparecki.com/articles/2012/...](#) by @aaronpk
Thu, Dec 17, 2015 10:22am -07:00 (via [brid-gy.appspot.com](#))



[Nicolas Grancher www.pbunk.com](#)
OAuth 2 Simplified - [aaronparecki.com/articles/2012/...](#)
Mon, Dec 21, 2015 12:17am -07:00 (via [brid-gy.appspot.com](#))




[Nicolas Grancher www.pbunk.com](#)
OAuth 2 Simplified - [aaronparecki.com/articles/2012/...](#)
Mon, Dec 21, 2015 12:17am -07:00 (via [brid-gy.appspot.com](#))


[by Oleg Dulin www.wikinewstech.com/us/author/oleg-dulin](#)
IDG Contributor Community: OAuth 2.zero: The protocol on the middle of the universe
Tue, Dec 22, 2015 9:11am -07:00

[by Oleg Dulin www.wikinewstech.com/us/author/oleg-dulin](#)
IDG Contributor Community: OAuth 2.zero: The protocol on the middle of the universe
Tue, Dec 22, 2015 7:42pm -07:00

[by Oleg Dulin www.wikinewstech.com/us/author/oleg-dulin](#)
IDG Contributor Community: OAuth 2.zero: The protocol on the middle of the universe
Wed, Dec 23, 2015 4:38am -07:00

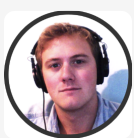


[Tuan Nguyen blog.tuannguyena.com](#)
OAuth 2 Simplified - Aaron Parecki [aaronparecki.com/articles/2012/...](#)
Mon, Dec 28, 2015 7:36pm -07:00 (via [brid-gy.appspot.com](#))




[admin php.javacss.space/author/admin](#)
Unable to get JSON response from cURL

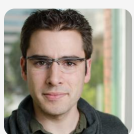
when hitting url https://api.instagram.com/oauth/authorize?client_id=&redirect_uri=http://localhost/instagram_flow/result.php&response_type=code&scope=basic+comments+relationships+likes
in browser response as:
{“code”: 400, “error_type”: “oauthexception”, “error_message”: “you must include valid client_id, response_type, , redirect_uri parameters”} ...
Mon, Jan 11, 2016 11:48am -07:00




[Ryan Wilson-Perkin ryanwilsonperkin.com](#)
Your simplified article on OAuth2 was incredibly easy to understand, thanks for writing it![aaronparecki.com/articles/2012/...](#)
Wed, Jan 13, 2016 2:11pm -07:00 (via [brid-gy.appspot.com](#))



[Ryan Wilson-Perkin ryanwilsonperkin.com](#)
@aaronpk Your simplified article on OAuth2 was the first I found easy to understand, thanks for writing it!
[aaronparecki.com/articles/2012/...](#)
Wed, Jan 13, 2016 2:13pm -07:00 (via [brid-gy.appspot.com](#))



[Aaron Parecki aaronparecki.com](#)
@rwilsonperkin Thanks! Glad you found it useful!
Wed, Jan 13, 2016 2:14pm -07:00 (via [brid-gy.appspot.com](#))



[cssphp.space/author](#)
Client secret for Django oauth –

i using django oauth toolkit , django rest oauth authentication mobile app. accessing protected resource client id , secret of app required . should store client secret. storing in apk unsafe can decompiled. obfuscation can reverse engineered . whats best , safe way serve client secret app.

it isn’t extremely of import maintain client id hidden, right not save client secret somewhere in app. exposing compromise security. ...

Wed, Jan 13, 2016 3:19pm -07:00



[Martin](#) [twitter.com/mm001bask](#)

[aaronparecki.com/articles/2012/...](#)

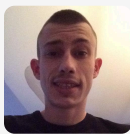
Sat, Jan 30, 2016 5:44am -07:00 (via [brid-gy.appspot.com](#))



[Marcin Gryszko](#) [grysz.com](#)

OAuth2 Simplified, Aaron Parecki [buff.ly/1pEYXXF](#)

Tue, Mar 22, 2016 3:10am -07:00 (via [brid-gy.appspot.com](#))



[Robin Chalas](#) [github.com/chalasr](#)

#OAuth2 Simplified (As much as it can be) [aaronparecki.com/2012/07/29/2/o...](#)

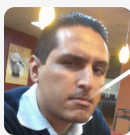
Wed, Apr 6, 2016 4:08pm -07:00 (via [brid-gy.appspot.com](#))



[dan mcweeney](#) [blog.danmcweeney.com](#)

Nice rundown on the spec. Good primer reference. [aaronparecki.com/2012/07/29/2/o...](#)

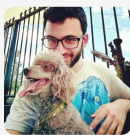
Sun, May 29, 2016 4:20pm -07:00 (via [brid-gy.appspot.com](#))



[Luis Angel Lopez](#) [twitter.com/lopezchairez](#)

@aaronpk finally understood when to use an #oauth2 server #php #laravel #laracasts [goo.gl/n4bebM](#)

Thu, Jun 30, 2016 1:02pm -07:00 (via [brid-gy.appspot.com](#))



[Adam Chalmers](#) [adamchalmers.github.io](#)

This is the best OAuth2 explainer I've found: [aaronparecki.com/2012/07/29/2/o...](#)

Thu, Jul 14, 2016 7:18pm -07:00 (via [brid-gy.appspot.com](#))



[ISRA](#) [twitter.com/IsraMtinajero](#)

OAuth 2 Simplified[aaronparecki.com/2012/07/29/2/o...](#)

Wed, Jul 27, 2016 9:49am -07:00 (via [brid-gy.appspot.com](#))



[Sec News Bot](#) [twitter.com/SecNewsBot](#)

Hacker News - OAuth 2 Simplified (2012) [ift.tt/24TC4Qk](#)

Wed, Jul 27, 2016 3:13pm -07:00 (via [brid-gy.appspot.com](#))



[Klaus Peter Laube](#) [www.klauslaube.com.br](#)

OAuth 2 Simplified [aaronparecki.com/2012/07/29/2/o...](#)

Thu, Aug 18, 2016 11:51am -07:00 (via [brid-gy.appspot.com](#))



[Ilker Temir](#) [www.linkedin.com/in/ilkertemir](#)

Confused by #oauth2? Read @aaronpk's "OAuth 2 Simplified", confusion will evaporate quickly. [aaronparecki.com/2012/07/29/2/o...](#)

Sun, Aug 21, 2016 6:37pm -07:00 (via [brid-gy.appspot.com](#))



[Juan](#) [soundcloud.com/elxavit0/sets](#)

Revisiting this really nice writeup on OAuth2 --> [aaronparecki.com/articles/2012/...](#) #oauth #oauth2 #standards #web #authentication

Tue, Aug 23, 2016 6:51am -07:00 (via [brid-gy.appspot.com](#))



[willemodendaal](#) [medium.com/the-curious-coder](#)

A nice simple explanation of oath2 scenarios: [aaronparecki.com/2012/07/29/2/o...](#)

Wed, Aug 24, 2016 11:27pm -07:00 (via [brid-gy.appspot.com](#))



[Florent Dupont](#) [florent-dupont.blogspot.fr](#)

Oauth2 simplified [aaronparecki.com/2012/07/29/2/o...](#)

Mon, Sep 26, 2016 1:31pm -07:00 (via [brid-gy.appspot.com](#))

Posted in [/articles](#)



Hi, I'm Aaron Parecki, co-founder of [IndieWebCamp](#). I maintain [oauth.net](#), and am the editor of the [Webmention specification](#).

I've been [tracking my location](#) since 2008, and write down everything I [eat](#) and [drink](#). I've [spoken](#) at conferences around the world about [owning your data](#), [OAuth](#), [quantified self](#), and explained [why R is a vowel](#).

 [IndieWebCamp](#) Founder

 [W3C](#) Editor

