

# Zend\_Authentication\_Adapter\_Oauth - Jason Austin

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

## Zend Framework: Zend\Authentication\Adapter\Oauth Component Proposal

<b>Proposed Component Name</b>	Zend\Authentication\Adapter\Oauth
<b>Developer Notes</b>	<a href="http://framework.zend.com/wiki/display/ZFDEV/Zend\Authentication\Adapter\Oauth">http://framework.zend.com/wiki/display/ZFDEV/Zend\Authentication\Adapter\Oauth</a>
<b>Proposers</b>	Jason Austin
<b>Zend Liaison</b>	TBD
<b>Revision</b>	1.0 - 17 November 2010: Initial Draft. 1.1 - 1 December 2010: Conversion to Zend\Authentication\Adapter\Oauth (wiki revision: 18)

## Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
9. Class Skeletons

## 1. Overview

Zend\Authentication\Adapter\Oauth is an authentication adapter designed to authenticate to third party OAuth providers such as Twitter, Facebook, etc.

## 2. References

## 3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** conform to the requirements of the Zend\Authentication\Adapter component (class Zend\Authentication\Adapter\Oauth implements Zend\Authentication\Adapter).
- The Zend\Authentication\Adapter\Oauth::authenticate() method **will not** return Zend\Authentication\Result::SUCCESS unless a valid OAuth access token is obtained from the configured source
- This adapter **will not** throw exceptions for conditions that may occur during normal operation with a properly configured adapter (i.e. authentication failure). All such exceptions will be caught in the adapter's authenticate method and translated into an appropriate Zend\Authentication\Result::FAILURE response.

- This adapter **will** return `Zend\Authentication\Result::FAILURE` if access was not granted by the end user from the OAuth provider (i.e. they deny access to the application)
- This adapter **will** require configuration of an OAuth consumer via the existing `Zend\Oauth` component
- This adapter **will** take advantage of `Zend\Session\Namespace` to save state while the user is taken away to the OAuth providers site, then back to the application
- This adapter **will** provide ability to change the `Zend\Session\Namespace` option
- This adapter **will** provide a mechanism to pass the OAuth return parameters (typically passed in the `$_GET` var) to the adapter for `Zend\Oauth` acquisition of an access token

## 4. Dependencies on Other Framework Components

- `Zend_Oauth`
- `Zend_Session`

## 5. Theory of Operation

The component will be used as an instance of `Zend\Authentication\Adapter`, implementing the `authenticate()` method. What makes this component unique is its need to leave the user's application to authenticate with a third-party OAuth provider, then return back to a state within the application.

The workflow would be something like:

- Create an instance of `Zend\Oauth\Consumer`, specifically containing 'consumerKey', 'consumerSecret', 'siteUrl' and 'callbackUrl'
- Instantiate `Zend\Authentication\Adapter\Oauth`, passing the options array to the constructor as well as the instance of `Zend\Oauth\Consumer`
- Call the `authenticate` method

```
<?php

$options = array(
    'consumerKey'    => '2342kljv23onv34ij34',
    'consumerSecret' => '231kjalkjoi408afklj4afseaf',
    'callbackUrl'    => 'http://mywebsite.com/login',
    'siteUrl'        => 'http://twitter.com/oauth'
);

use Zend\Oauth\Consumer as OAuthConsumer;
$consumer = new OAuthConsumer($options);

use Zend\Authentication\Adapter\Oauth as OAuthAdapter;
$adapter = new OAuthAdapter();
$adapter->setConsumer($consumer);

$result = $auth->authenticate($adapter);

?>
```

- Authenticate method would acquire a request token
- Store the request token in a `Zend\Session` object
- Redirect the user to the OAuth provider to authenticate
- User will allow or deny access from twitter's website
- OAuth provider will redirect the user back to the application through the `callbackUrl` passed in the options
- Obtain an access token based on the request token stored in the `Zend\Session` object
- Unset the `Zend\Session` object
- Return a `key=>value` array of the body returned from the acquisition of the Access Token

Sample code can be found here: [http://github.com/jfaustin/Zend\\_Auth\\_Adapter\\_Oauth](http://github.com/jfaustin/Zend_Auth_Adapter_Oauth)

## 6. Milestones / Tasks

- Milestone 1: [DONE] Prototype
- Milestone 2: Create documentation necessary to use the component
- Milestone 3: Working prototype checked into the incubator
- Milestone 4: Create Unit Tests

## 7. Class Index

- Zend\Authentication\Adapter\Oauth

## 8. Use Cases

## 9. Class Skeletons

```
use Zend\Authentication\Adapter as AuthenticationAdapter,  
  
class Oauth implements AuthenticationAdapter  
{  
    public function __construct($options = array, Zend\Oauth\Consumer = null)  
    {  
    }  
  
    public function authenticate()  
}
```

]]></ac:plain-text-body></ac:macro>  
]]></ac:plain-text-body></ac:macro>