

Authorize a SharpBox based DropBox application

Table of Contents

What is a DropBox application.....	2
Why does a user authorize an application.....	2
The authorization process at a glance	3
How to implement DropBox application authorization with SharpBox	3
Generate a request token and receive the authorization URL	3
Visit the authorization URL via Webbrowser control	4
Exchange the request token into an access token	4
Summary	4



What is a DropBox application

DropBox allows independent software vendors (ISVs) creating applications based on their RESTful API. This helps to build applications which are using DropBox as storage backend. Every application which want to get access to a DropBox needs to be registered at DropBox and needs to be verified from the DropBox team as well. This kind of software is called a DropBox Application and will be identified with a consumer key and a consumer secret.

Why does a user authorize an application

The user wants to have control over all workflows which accessing their personal data, so every application needs to be accepted from the user at minimum one time. On the other side with this registration the user gets a way at DropBox to reject the registration later to prevent access if your application is doing something wrong. Having access to the application as self is not necessary to prevent access in the future.



[Back to home](#)

[Account Info](#) [Account Settings](#) [My Computers](#) [Referral Status](#) [My Apps](#)

My Apps

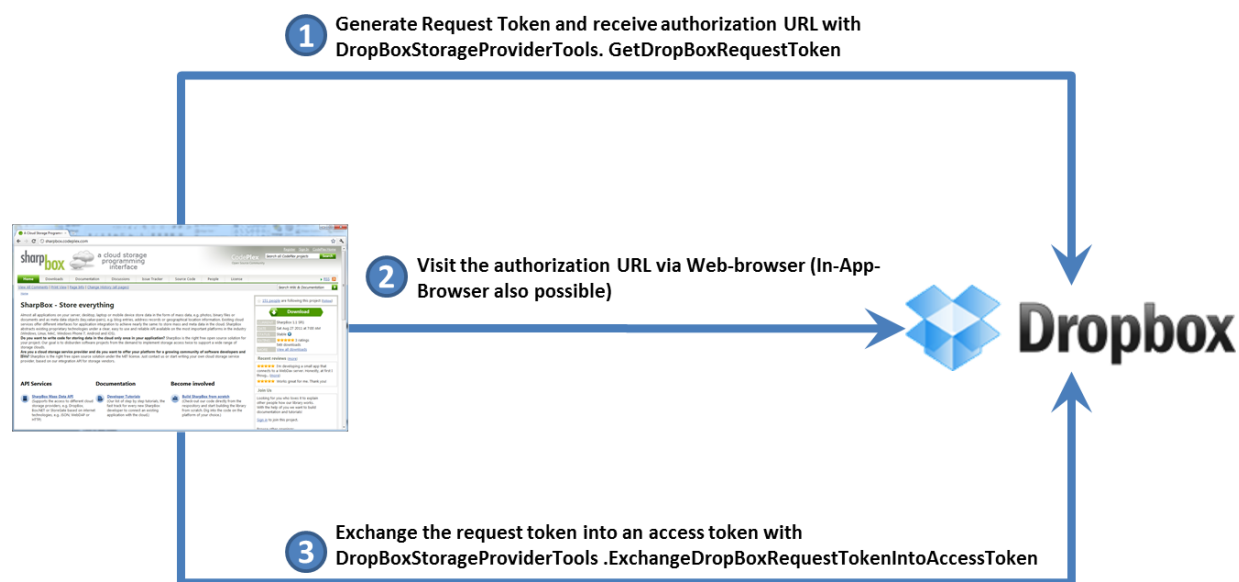
You have given these apps access to your Dropbox account.

App Name	Publisher	Access Type	
Documents To Go	DataViz	Full Dropbox	×
SharpBoxBeta3	SharpBoxBeta3	Full Dropbox	×
deitest	deitest	No access - App folder deleted	×
SmartCode VNC Manager	SmartCode Solutions	Full Dropbox	×
Cloud Infrastructure Backup	app.limit	Full Dropbox	×
iThoughtsHD	Craig Scott	Full Dropbox	×

Document Name: Authorize a SharpBox based DropBox application
Document Type: Tutorial
Author: Dirk Eisenberg (dirk dot Eisenberg at gmail dot com)
License: MIT as part of the SharpBox Project

The authorization process at a glance

The DropBox API is based on the [oAuth](#) protocol which needs an issued security token from DropBox to get access to the users cloud storage. During this process DropBox offers a special WebPage where the user needs to allow the access of your application which will be identified by application key and application secret. At the end of the process your application will get a security token (which contains key and secret as well) which is valid for 10 years. This token can be stored on a cache and has to be reused for reconnects to the cloud storage. The follow figure shows the process in detail:



How to implement DropBox application authorization with SharpBox

SharpBox offers for the DropBox [oAuth](#) based authentication and authorization process special APIs which are only necessary to drive the described steps. This set of APIs are implemented in the class [DropBoxStorageProviderTools](#). All other APIs are cloud storage vendor independent and can be used for all storage provider in the same way. The following list illustrates the steps to implement an authorization process which results with an issues access token.

Generate a request token and receive the authorization URL

In the first phase a request token has to be generated based on the consumer key and consumer secret issued by DropBox in your developer account interface. The interface will be available after registration under <http://www.dropbox.com/developers>



```
// get the config of dropbox
DropBoxConfiguration config =
CloudStorage.GetCloudConfigurationEasy(nSupportedCloudConfigurations.DropBox) as
DropBoxConfiguration;

// set your own callback which will be called from DropBox after successful
// authorization
config.AuthorizationCallBack = new Uri(_AuthorizationCallbackUri);

// create a request token
DropBoxRequestToken requestToken =
DropBoxStorageProviderTools.GetDropBoxRequestToken(config,
    <<YOUR CONSUME KEY>>,
    <<YOUR CONSUMER SECRET>>);
```

Visit the authorization URL via Webbrowser control

In phase 2 the generated authorization URL has to be visited in a web browser control or in an inner frame of your ASP.NET webpage. The URL will be returned from SharpBox with the following code fragment:

```
// call the authorization url via WebBrowser Plugin
String AuthorizationUrl =
DropBoxStorageProviderTools.GetDropBoxAuthorizationUrl(config, requestToken);
```

Exchange the request token into an access token

The last phase converts the request token (which is only be usable for the authorization process) into an issued access token which can be stored on a local cache and has to be reused during login phase when the user comes back with your application. The following code performs the exchange process:

```
// create the access token
return DropBoxStorageProviderTools.ExchangeDropBoxRequestTokenIntoAccessToken(config,
    <<YOUR CONSUMER KEY>>, <<YOUR CONSUMER SECRET>>, requestToken);
```

Summary

After this tutorial it should be possible to implement an authorization process for DropBox applications aligned with the terms of use. This process is also approved by DropBox and will ensure that the generated application can be approved for public use. We propose to send DropBox a video or a demo account to your application, this speeds up the approval process!

Document Name: Authorize a SharpBox based DropBox application
Document Type: Tutorial
Author: Dirk Eisenberg (dirk dot Eisenberg at gmail dot com)
License: MIT as part of the SharpBox Project