
TubePress Documentation

Release

TubePress LLC

August 12, 2014

1	Introduction	3
1.1	Features	3
1.2	How to Get TubePress	8
1.3	Getting Help	9
1.4	License	9
2	User Manuals	11
2.1	TubePress for WordPress	11
2.2	TubePress Manual for PHP	33
2.3	TubePress for Wix	49
2.4	Official TubePress Add-Ons	57
3	Reference	77
3.1	Options	77
3.2	Index	111

TubePress adds dynamic and interactive video features to websites. It's highly configurable and designed for website builders and users of any skill level. A typical use of TubePress is to *embed a video gallery* into a web page.

TubePress is available both as a downloadable package and via several cloud-hosted, zero-installation services. The software is largely open sourced, and it's designed and maintained by [TubePress LLC](#), a software company based in Pittsburgh, Pennsylvania.

Introduction

1.1 Features

TubePress allows website builders of any skill level to add dynamic video features to their site. Click a feature below to learn more about what TubePress can do for your site.

1.1.1 Video Galleries

Creating video galleries is how TubePress made a name for itself. The software lets you easily embed beautiful, highly-functional thumbnail galleries throughout you site, and you complete control over the entire process. A live demo of a TubePress gallery can be seen [here](#).

Coles Garden wedding



- 1
- 2
- 3
- 4
- 5
- next »



Hotel Zaza, St. Anne's, The Corinthian wedding
6:24



Perini Ranch wedding film
4:37



Mayo Hotel wedding
4:38



Coles Garden 8mm wedding film
4:55



Norman wedding video
5:38



Skelly Lodge wedding
4:38



Mayo Hotel, First United Methodist wedding
6:06



Coles Garden wedding
4:43



Skirvin Hotel wedding
4:39



Dominion House wedding
4:59



Engagement shoot, Love Story film at Boone Pickens Stadium
7:59



Five Oaks Lodge wedding
4:36

- 1
- 2
- 3
- 4
- 5
- next »

The process to create a video gallery is always the same ...

1. **Choose which videos to display.** Out of the box, TubePress supports a multitude of YouTube and Vimeo video sources. More video providers are on the way!
2. **Configure TubePress to display those videos.** Check the *user manual for your environment* for details on how to do this.
3. **Add the gallery to a website.** Again this process is different depending on your environment, so please refer to your specific *user manual*.

Video Galleries from Multiple Sources

Note: This feature is currently only available with the downloadable TubePress Pro. It is not yet available on the cloud-hosted TubePress services, but that functionality is coming soon.

With TubePress Pro, you can create galleries that are composed of video from multiple video sources. TubePress can even combine videos from multiple video providers - for instance, you can easily have YouTube and Vimeo videos intermingled in the same gallery!

As an example, say that you want to create a single gallery containing videos from the following sources, all combined into a single gallery:

- the YouTube playlist F5D324185EE73FEC
- uploads from YouTube user bbc
- uploads from YouTube user funnyordie
- YouTube search for the exact term “spirit of truth”
- the Vimeo album 140484

This is easy to accomplish with TubePress Pro. The trick is to use + in some of your shortcode options to combine the sources. For the example above, the TubePress shortcode might look like this:

```
[tubepress mode="playlist + user + tag + vimeoAlbum" playlistValue="F5D324185EE73FEC"
  userValue="bbc + funnyordie" tagValue="'spirit of truth'" vimeoAlbumValue="140484"]
```

The videos from all 5 sources will be collected and seamlessly assembled into a single TubePress gallery. You can combine any number of gallery sources with any number of parameters to each source.

Please keep in mind the following “gotchas”

- The `resultsPerPage` option will apply to each source, so expect each gallery page to contain up to `resultsPerPage` x `<number of sources>` videos per page. In the example above, assuming `resultsPerPage` was set to 10, there would be 40 videos per page (10 videos per page x 4 sources).
- It is strongly recommended to enable the TubePress API cache, as TubePress has to perform one network operation for each video source
- As the user pages through the gallery, some of the sources will drop off quietly. For instance, if one video source has 1,000 videos and another has 10 videos, the second page of the gallery won't contain any of the latter source's videos.

1.1.2 Single Video Embeds

Embedding a single video into a website is easy and common. Nearly every video provider includes an “embed” option for each video that provides an HTML `<iframe>` that can be added to any website. If you're reading this, you've probably done this countless times.

TubePress improves the embedding process by

1. Abstracting away the HTML. No need to copy and paste a long and complicated `<iframe>` snippet.
2. Allowing you to control the embedded video player (e.g. colors, behavior, size, etc).
3. Bringing in all of the video's meta information (length, description, author, keywords, etc). You can even have fine-grained control over how the meta information is presented; it's just simple HTML.

Help for Addicted Doctors - Dr. Rodney Hough on CBS 21



vs.



3:29

Author: 3hough

Keywords: doctors drug addiction pennsylvania physicians health

Category: People & Blogs

Number of ratings: 1

Average rating: 5.00

View count: 5,809

Date posted: Sep 11, 2009

My father, Dr. Rodney Hough, interviewed by CBS back in 2007 regarding his recovery from addiction to pain pills. This aired on CBS 21 News, which serves central Pennsylvania, in February 2008.

To embed a single video with TubePress, you simply need to give TubePress the ID of the video you'd like to embed. For instance, if you want to embed the YouTube video with ID J51kfdN5aA, you would use the following shortcode:

```
[tubepress video="J51kfdN5aA"]
```

1.1.3 Interactive Searching

On this page

- [Introduction](#)
- [Search Input](#)
- [Search Results](#)
- [Ajax-Enabled Search Results](#)

Introduction

TubePress provides an end-to-end solution for allowing your users to interactively search YouTube or Vimeo, all without leaving your site. The image below is a simple example showing an interactive search input along with a few results.

pittsburgh steelers

1 2 3 4 5 ... 49 50 next »

 <p>Pittsburgh Steelers: Unstoppable 2:49 Views: 527,776 ES Posthumus - Unstoppable is the song. Check out my</p>	 <p>2010- 2011 Pittsburgh Steelers- "Black And Yellow" 4:09 Views: 1,103,816 A tribute video to the city of</p>	 <p>2011-12 Pittsburgh Steelers Hype Up Video 7:17 Views: 9,191 Just a little something for this upcoming Pittsburgh Steeler</p>	 <p>Madden NFL 12 Gameplay - Pittsburgh Steelers vs Cleveland Browns HD 4:46 Views: 31,284</p>
 <p>Madden 11: Kilo Replay ft. Mike Wallace and the Pittsburgh Steelers (Madden NFL 11) Sport 10:15 Views: 2,080</p>	 <p>(Episode 3) Pittsburgh Steelers 1st & 10 2009 AFC Championship: Baltimore vs. Pittsburgh 8:05 Views: 467</p>	 <p>Pittsburgh Steelers Big Hits 3:40 Views: 149 some of the biggest hits from the steelers must watch</p>	 <p>2 & Some Change - 7/13/11 (Pittsburgh Steelers' James Harrison Sounds Off) Sports 5:05 Views: 4,981</p>

Like everything else with TubePress, interactive searching has loads of options and is fully customizable to suit your needs. To use the feature, you'll need to invoke TubePress twice:

1. Once for displaying the search input and
2. Once more for displaying the user's search results

Let's go over each invocation in detail.

Search Input

Searching wouldn't be very fun if you didn't provide an easy way for your users to search. With TubePress, it's really easy:

```
[tubepress output="searchInput"]
```

When invoked on your page, this shortcode will print out a search input box along with a submit button. If you'd like to customize the HTML of either of these, you can edit your theme's HTML template at `search/search_input.tpl.php`.

By default, the search form will submit back to the same page. If you want the results to show up on another page, just set `searchResultsUrl` (most users shouldn't need to do this).

```
[tubepress output="searchInput" searchResultsUrl="http://mysite.com/videos.php"]
```

Now that you have a search input form, you'll need to invoke TubePress somewhere else to display the search results.

Search Results

Displaying search results is also easy. Add the following shortcode wherever on the page you'd like the search results to show up:

```
[tubepress output="searchResults" searchProvider="vimeo"]
```

The `searchProvider` option can be set to either `youtube` or `vimeo`, and this determines if the gallery will display YouTube videos or Vimeo videos.

This shortcode will also display a placeholder gallery if the user isn't searching, but you can make it invisible until it has search results by setting `searchResultsOnly`:

```
[tubepress output="searchResults" searchProvider="youtube" searchResultsOnly="true"]
```

To configure the resulting thumbnail gallery of search results, simply add any TubePress options to this shortcode.

Ajax-Enabled Search Results

With TubePress Pro, search results are displayed asynchronously (i.e. without a full page load). It's also a bit easier to set up than standard searching because you only need to invoke TubePress once on the page. The only catch is you also have to identify the DOM element where you want the search results to show up:

```
[tubepress output="ajaxSearchInput" searchResultsDomId="#somediv" searchProvider="vimeo"]
```

For the `searchResultsDomId` option value, just use any jQuery selector that uniquely identifies a DOM element that can accept HTML. Like the standard search input, you can customize the HTML for the search input elements by editing your theme's HTML template at `search/ajax_search_input.tpl.php`.

1.2 How to Get TubePress

There are two independent installation paths for TubePress:

1. One of two **downloadable** software packages:
 - [TubePress for WordPress](#)
 - TubePress Pro (for *WordPress* and/or *any PHP site*)
2. One of two **cloud-hosted** TubePress installations:
 - [TubePress Express](#)
 - [TubePress for Wix](#)

Either installation path provides nearly identical features, but there are a number of drawbacks and advantages to each:

	Cloud-hosted TubePress	Downloadable TubePress
Required technical skills	None	Basic PHP knowledge ³
Upgrades and patches	Automatic	Manual or Automatic ⁴
Installation time	None	1 - 5 minutes, usually
Extra/Premium features	All Included automatically	Must be installed manually
Ability to modify source	None	Full access
Website limitations	None - works on <i>any</i> site	Web server must support PHP 5.2+
Pricing	30-day free trial	Free and premium versions available

¹Basic PHP knowledge is only required if using TubePress *outside* of WordPress

²The free version of TubePress can be auto-updated when used in WordPress 3.7 or higher.

1.3 Getting Help

The best place to get assistance with TubePress is the [community discussion forum](#). TubePress LLC has knowledgeable staff that monitors and responds to new posts in the forums.

1.4 License

The core of TubePress, which comprises a very large percentage of the overall codebase, is open source and released under the [Mozilla Public License 2.0 \(MPL 2.0\)](#). You can browse [the source code for TubePress on GitHub](#).

The premium products sold by TubePress LLC are built as add-ons to TubePress, and these products are under a proprietary, closed-source license.

³Basic PHP knowledge is only required if using TubePress *outside* of WordPress

⁴The free version of TubePress can be auto-updated when used in WordPress 3.7 or higher.

User Manuals

Below you can find tailored user manuals for using TubePress, and its related software, in various environments.

2.1 TubePress for WordPress

TubePress is available to use as a WordPress plugin, which makes it a breeze to use on any WordPress site.

While the WordPress plugin is available 100% for free, we recommend the premium version of the plugin: **TubePress Pro**. TubePress Pro includes all the features of the free TubePress plugin, plus the following additional capabilities:

- Priority technical support (i.e. you jump to the front of the line)
- Use in WordPress templates (outside post or page content)
- Multiple video sources in a single, unified gallery (e.g. YouTube + Vimeo in the same gallery)
- Auto-next playback (when a video ends, start playback of the next video in the gallery)
- Full HTTPS capability
- Extended controls for the YouTube video player
- Highest-resolution thumbnails from YouTube and Vimeo
- Asynchronous paging through gallery pages (video playback does not stop)
- Asynchronous interactive video search
- Play videos with Fancybox or Tinybox

2.1.1 Manual Contents

Install / Upgrade

This page details how to install, upgrade, and uninstall TubePress when used as a WordPress plugin.

On this page

- Requirements
- Install
 - Free Version
 - TubePress Pro
 - Vimeo API Setup
 - Optional TubePress Pro Installations
 - * JW Player (version 5.x)
 - * Shadowbox.js
- Upgrade
 - Free Version
 - TubePress Pro
- Uninstall

Requirements

It's a good idea to first make sure that your web server meets the minimum requirements to run TubePress. You will need:

- PHP 5.2 or higher
- [PHP DOM extension](#) (enabled by default in PHP, only required for using TubePress with YouTube)
- [PHP PCRE extension](#) built with PCRE version 7.1 or higher (enabled in most PHP installations)

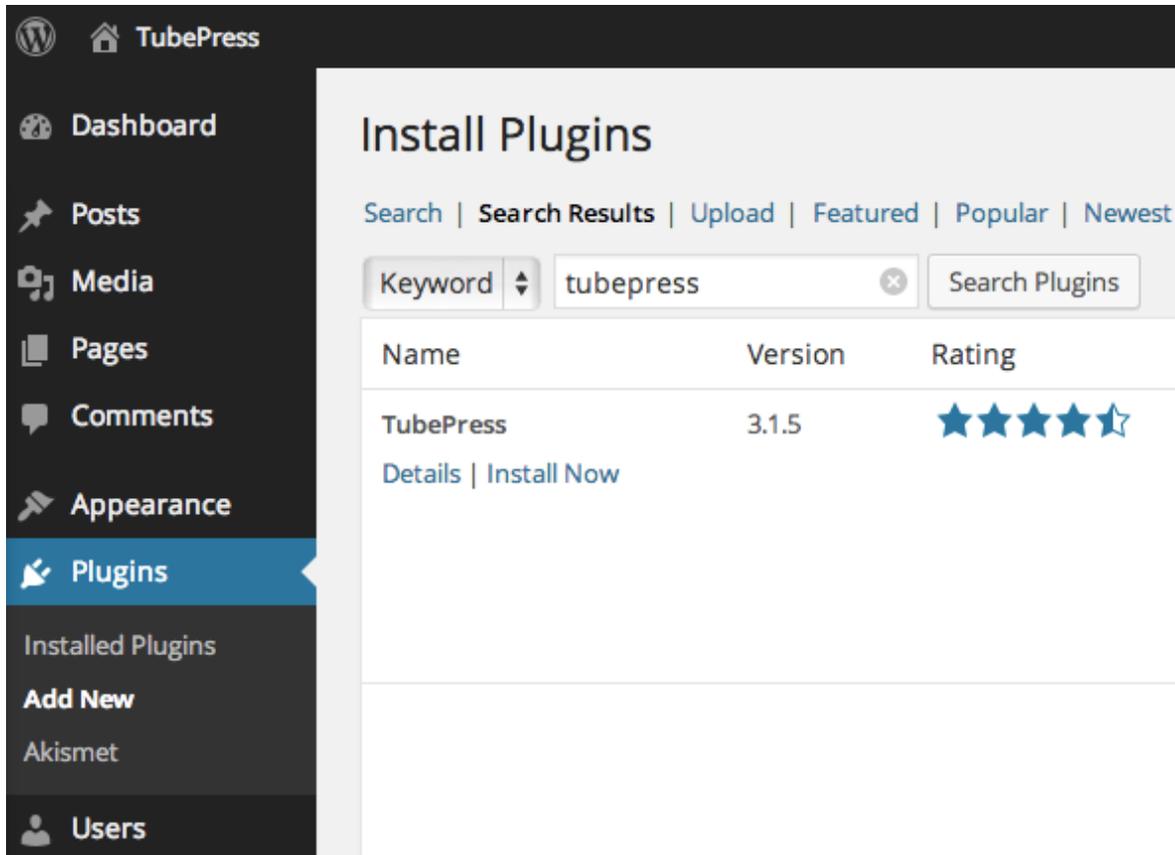
Not sure if your server meets these requirements? You should either

1. Ask your hosting provider (good idea to include a link to this page!)
2. Install a WordPress plugin that prints out system information. Here are some examples:
 - [WordPress phpinfo\(\)](#)
 - [WP-ServerInfo](#)

Install

Free Version Installing the free WordPress plugin is simple. The easiest method is from your WordPress dashboard.

1. Follow [these instructions](#) to perform the plugin installation and activation. You can search for `tubepress` and it should come up as the first result.



2. *Vimeo users only.* Perform the *one-time Vimeo API setup*.

If the dashboard installation method is not available, or if you'd prefer to install the plugin manually, here's how to do it.

1. Download the plugin from its [page on wordpress.org](#).
2. Follow [these instructions](#) to finish the manual installation and activation.
3. *Vimeo users only.* Perform the *one-time Vimeo API setup*.

TubePress Pro Installation of TubePress Pro into a WordPress environment must be performed manually, since TubePress Pro is *open* but not *public* source code.

1. If you haven't already, [purchase a TubePress Pro license](#).
2. Download TubePress Pro from [here](#).
3. Follow [these instructions](#) to finish the manual installation and activation of the plugin.
4. *Vimeo users only.* Perform the *one-time Vimeo API setup*.
5. *Optional.* [Install support for Shadowbox.js and/or JW Player 5](#).

Vimeo API Setup

Note: If you aren't using Vimeo, you can safely skip this section.

In order to use TubePress with Vimeo, you are required to obtain a "consumer key" and "consumer secret" from Vimeo. This is a one-time setup required by Vimeo. Thankfully it's easy to create these credentials and supply them to TubePress.

1. [Log in to Vimeo](#), if you haven't already.
2. [Click here](#) to create a new Vimeo app. You can fill in anything for the fields. Below is a sample form that's filled out. Then click the *Create App* button.

vimeo developer();

APIs Player

My Advanced API Apps /

Create a new App

App Name *

App Description *

When users authenticate via your app, this is what they'll see.

App URL *

Your app's publicly accessible home page where users can get more information or download your app.

App Logo URL

The URL of a logo to show with your app when authenticating. Displayed at 100x100 px.

App Callback URL

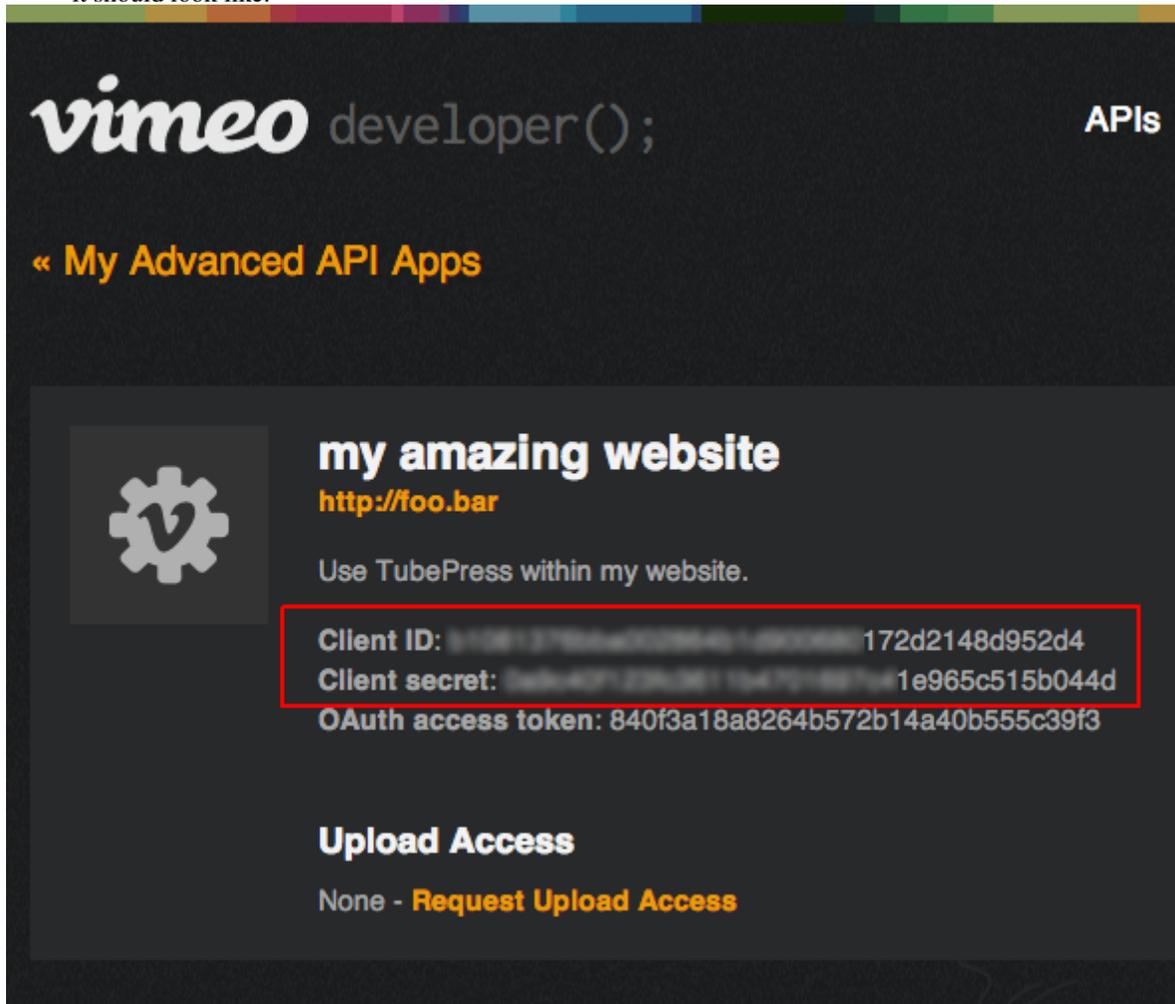
This is the URL we'll send a user back to when authenticating.

I agree that my application does not violate the [Vimeo API License Agreement](#) or the [Vimeo](#)

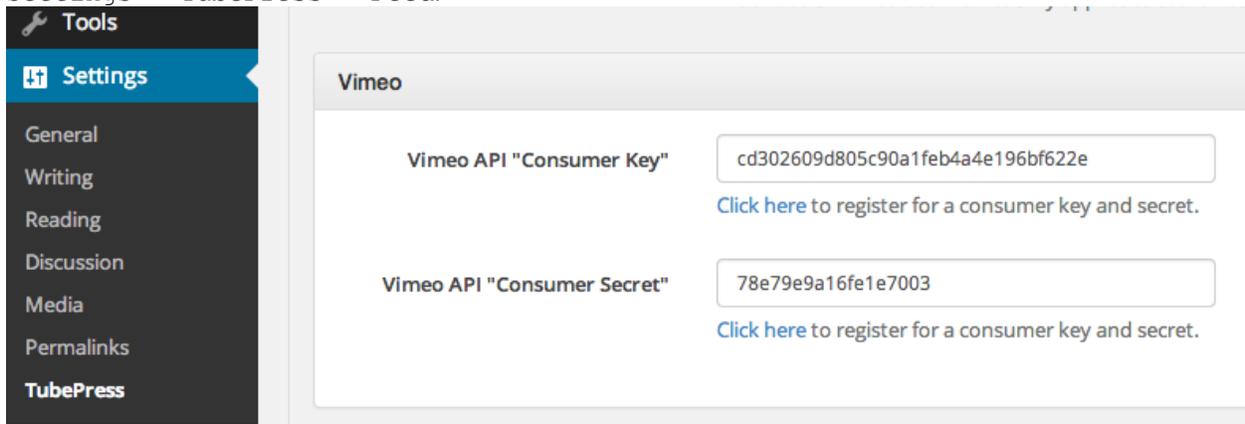
[Terms of Service](#) or WordPress

* Required

3. Make a note of the “Client ID” and “Client secret” of your new app. See the red box below for an idea of what it should look like.



That's it! You now simply need to provide these credentials to TubePress. You can do this from WP Admin > Settings > TubePress > Feed.



Make sure not to mix up the “key” and the “secret”!

Optional TubePress Pro Installations Due to licensing restrictions, TubePress Pro does not ship with every library that the free WordPress plugin includes. However it's very easy to install these optional libraries manually. This section provides installation instructions.

JW Player (version 5.x) TubePress can use [JW Player 5.10](#) to play YouTube videos. We are working to integrate JW Player 6.x, but this feature is not yet ready for production.

1. Download the file `player.swf` from [here](#).
2. Upload the file to your TubePress Pro installation at
`<tubepress_home>/src/main/web/flash/longtail/player.swf`.

Shadowbox.js

1. Download Shadowbox.js from [here](#). The image below shows the required options when configuring your download. **You may select additional checkboxes (though not recommended), but make sure that at least the settings shown are chosen.**

Shadowbox.js

[Home](#)[Download](#)[Support](#)[Usage](#)[Options](#)[API](#)[FAQ](#)

This page allows you to create a custom build of Shadowbox that includes only the features you need for your project. Compiling all of the necessary source files into one improves performance and

Choose the [adapter](#) you'll be using:

Select the player(s) for the types of content you want to display:

- Images
- External sites and pages
- Inline and/or dynamic HTML
- Flash (swf files)
- Flash video (flv files)
- QuickTime
- Windows Media Player

Choose the language support you need:

- Include support for using CSS selectors to select links

The current version of Shadowbox is
3.0.3



ZIP



TAR

2. The download will save to a file named something like `shadowbox-3.0.3.zip`. Unzip this file, and it will expand into a directory named `shadowbox-3.0.3`.
3. Copy the *contents* of this directory (not the directory itself) to your TubePress Pro installation at

```
<tubepress_home>/src/main/web/players/shadowbox/lib/.
```

4. *Optional.* You may see scrollbars in the Shadowbox.js display. To fix this, simply edit `lib/shadowbox.css`. On line 8, remove the `overflow:auto` attribute. i.e. change it from

```
#sb-player.html{height:100%;overflow:auto;}
```

to

```
#sb-player.html{height:100%;}
```

Upgrade

This page details the official upgrade procedures for TubePress. The installation steps are different depending on if you are using TubePress Pro or the free version of TubePress.

Free Version We strongly recommend using [WordPress's built-in update mechanism](#) to keep the free version of TubePress up to date.

If you would rather upgrade manually, here is the procedure:

1. Deactivate TubePress from WP Admin > Plugins.
2. Remove the existing TubePress installation directory from `wp-content/plugins/tubepress`.
3. Repeat the *free WordPress plugin installation procedure*.

TubePress Pro

Caution: Do not use WordPress's auto-update for TubePress Pro! Your installation will be downgraded to the free version of TubePress.

Here is the safe upgrade procedure for TubePress Pro when used in WordPress:

1. Deactivate TubePress Pro from WP Admin > Plugins.
2. Remove the existing TubePress Pro installation directory from `wp-content/plugins/tubepress_pro_x_y_z`.
3. Repeat the *TubePress Pro installation procedure* with the latest copy of TubePress Pro.

Uninstall

If you'd like to completely uninstall TubePress from your WordPress installation for any reason, here are the steps:

1. Deactivate TubePress from WP Admin > Plugins.
2. Remove the TubePress installation directory from `wp-content/plugins/`.

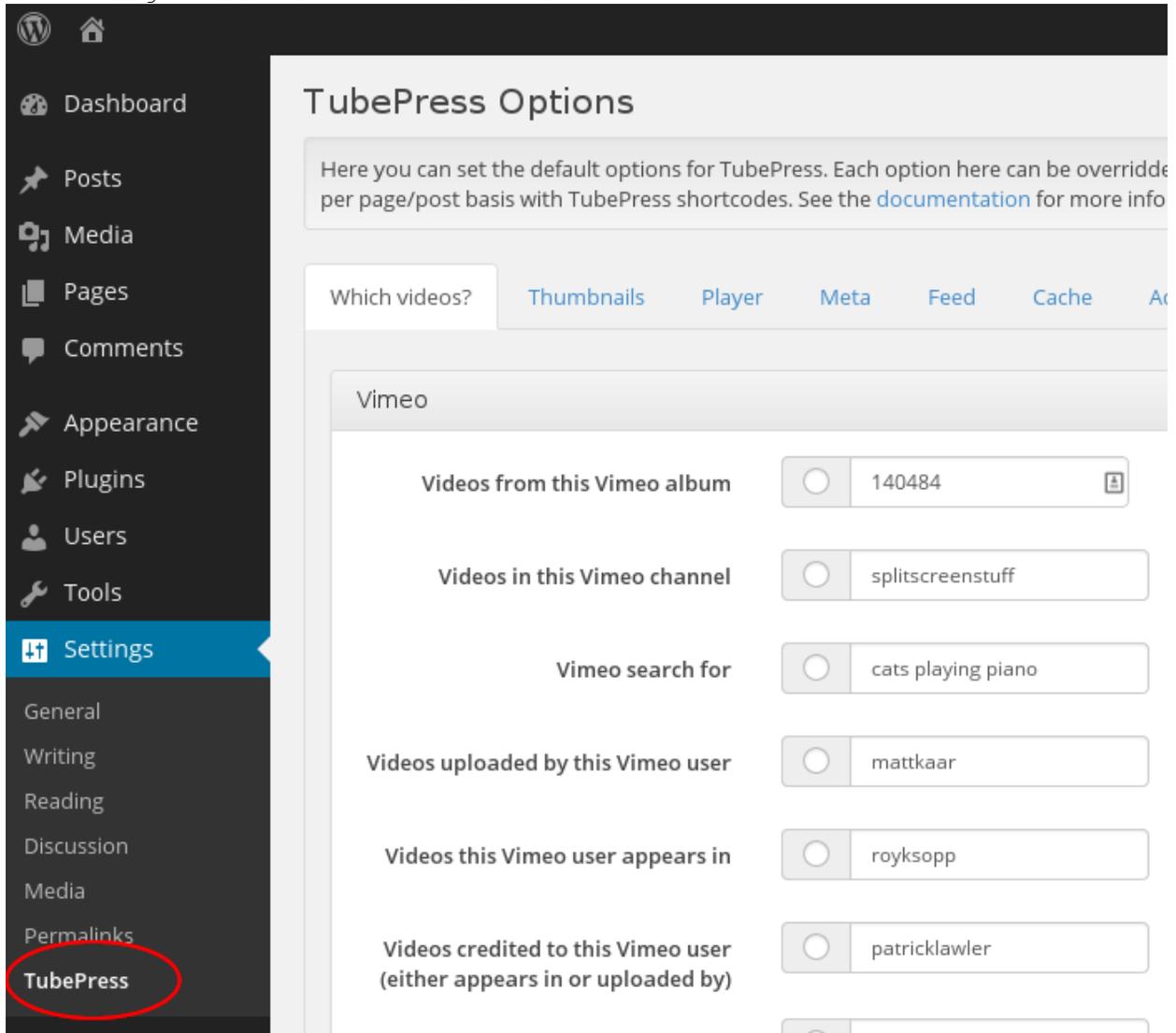
Using TubePress in WordPress

On This Page

- [Configuration](#)
- [Add TubePress to Posts and Pages](#)
- [Add TubePress to WordPress Templates](#)
- [Changing the Default Language](#)

Configuration

TubePress comes with a user-friendly options page that can be found from your WordPress admin dashboard. Simply visit `Settings > TubePress` to access the screen.



Add TubePress to Posts and Pages

Follow these instructions to use TubePress (free or Pro) in WordPress posts and/or pages. It's dead simple. Simply type

```
[tubepress]
```

anywhere in a WordPress post or page. When a user visits this post/page, TubePress will replace `[tubepress]` with its generated HTML content.

You can configure the default TubePress options from within your WordPress administration dashboard by visiting `Settings > TubePress`. If you just type `[tubepress]` on a post or a page, these options will be used. **The real power of the plugin lies in the ability to be invoked in unique ways on multiple posts and pages.** This is done by passing options to the `[tubepress]` shortcode. For example, say on one page you want to display a gallery of

YouTube videos that are tagged with “football”, want 15 videos per page, and want to play the videos in an HTML popup window. On this page, you would type

```
[tubepress mode="tag" tagValue="football" playerLocation="popup" resultsPerPage="15"]
```

on the page where you want the gallery to show up. And say on another page you want to display a different gallery with the options that you set in Settings > TubePress. On this page you would type

```
[tubepress]
```

where you want TubePress to inject its content. Follow this procedure for unlimited galleries on unlimited posts/pages. Just use *any TubePress option* to control TubePress.

Add TubePress to WordPress Templates

Note: TubePress Pro is required to use this feature, and basic PHP knowledge is recommended.

Follow these instructions to use TubePress Pro inside WordPress templates. The code snippet below shows the world’s simplest WordPress template with TubePress Pro added. As you can see by the highlighted lines below, there are only 2 steps:

```

1  <?php
2
3  include WP_PLUGIN_DIR . '/tubepress_pro_x_y_z/src/main/php/classes/TubePressPro.php';
4
5  get_header();
6
7  print TubePressPro::getHtmlForShortcode('mode="tag" tagValue="pittsburgh steelers"');
8
9  if (have_posts()) :
10     while (have_posts()) :
11         the_post();
12         the_content();
13     endwhile;
14 endif;
15 get_sidebar();
16 get_footer();
17 ?>
```

Notes on the two highlighted lines above:

1. Include the TubePress Pro library file `TubePressPro.php`. Not that in the code sample above you will need to replace `tubepress_pro_x_y_z` with the actual version of TubePress Pro that you downloaded (e.g. `tubepress_pro_3_2_6`).
2. The `getHtmlForShortcode()` static function takes a single string parameter which is any space-separated combination of *TubePress options*. It returns the HTML output of TubePress as a string. You may repeat this function call as many times as you like, any with any combination of options.

Changing the Default Language

TubePress is available in several languages besides English. The image below is an example of TubePress when configured for Italian (left) and Russian (right).



My Favorite Videos: Dwayne Johnson, aka "The Rock"
 1:38
 Autore: celebrityplaylists
 Keywords: The Rock wrestling Dwayne Johnson
 URL
 Categoria: People & Blogs
 Valutazioni: 840
 Valutazione: 4.64
 ID: acSw0yBBhio
 Visualizzazioni: 527,256
 Pubblicato: 2 days ago
 Dwayne Johnson, aka "The Rock," tells you a little something about his favorite ...



Your State of the Union - Submit Now
 1:00
 Autore: citizentube
 Keywords: obama state of the union citizentube speech politics news president of the united states president barack obama google moderator
 URL
 Categoria: News & Politics
 Valutazioni: 377
 Valutazione: 2.60
 ID: avitxo8ueEA
 Visualizzazioni: 189,419
 Pubblicato: 4 days ago
 On Tuesday, January 27, President Obama will deliver his first-ever State of the...



One Year In, a Closer Look at the Obama Presidency
 11:43
 Autore: PBSNewsHour
 Keywords: President Obama PBS NewsHour Jeffrey Brown Democrats Health Care
 URL
 Categoria: News & Politics
 Valutazioni: 598
 Valutazione: 2.64
 ID: AzMmLvejHMM
 Visualizzazioni: 122,442
 Pubblicato: 4 days ago
 As the one year anniversary of President Barack Obama's Inauguration arrives, Je...



My Favorite Videos: Dwayne Johnson, aka "The Rock"
 1:38
 Автор: celebrityplaylists
 Ключевые слова: The Rock wrestling Dwayne Johnson
 ссылка
 Категория: People & Blogs
 Рейтинги: 840
 Рейтинг: 4.64
 ID: acSw0yBBhio
 Просмотров: 527,256
 Добавлен: 2 days ago
 Dwayne Johnson, aka "The Rock," tells you a little something about his favorite ...



Your State of the Union - Submit Now
 1:00
 Автор: citizentube
 Ключевые слова: obama state of the union citizentube speech politics news president of the united states president barack obama google moderator
 ссылка
 Категория: News & Politics
 Рейтинги: 377
 Рейтинг: 2.60
 ID: avitxo8ueEA
 Просмотров: 189,419
 Добавлен: 4 days ago
 On Tuesday, January 27, President Obama will deliver his first-ever State of the...



One Year In, a Closer Look at the Obama Presidency
 11:43
 Автор: PBSNewsHour
 Ключевые слова: President Obama PBS NewsHour Jeffrey Brown Democrats Health Care
 ссылка
 Категория: News & Politics
 Рейтинги: 598
 Рейтинг: 2.64
 ID: AzMmLvejHMM
 Просмотров: 122,442
 Добавлен: 4 days ago
 As the one year anniversary of President Barack Obama's Inauguration arrives, Je...

TubePress will determine which language to use based on the `WP_LANG` setting that you can define in `wp-config.php`. Details for how to switch the language of your WordPress blog can be found [here](#).

Using Add-Ons & Themes

On This Page

- [What are Add-Ons & Themes?](#)
- [TubePress Content Directory](#)
- [Installing Add-Ons & Themes](#)

What are Add-Ons & Themes?

Much like WordPress itself, TubePress can be infinitely customized by add-ons and themes.

Note: *TubePress* add-ons and themes should not be confused with *WordPress* plugins and themes. While conceptually similar, the two are not interchangeable.

The central source for all TubePress add-ons and themes is the [TubePress Marketplace](#). There you can browse and download add-ons and themes developed by both TubePress LLC as well as community contributions.

TubePress Content Directory

Both add-ons and themes are stored in the “TubePress Content Directory”, which is a special directory inside your WordPress installation located at `wp-content/tubepress-content`. TubePress is able to automatically generate this directory for you, in most cases, and by default it’s mostly empty.

The TubePress Content Directory has a few rules. First, it must always be named `tubepress-content`. It must also have the following immediate subdirectories:

add-ons TubePress add-ons that aren't shipped with TubePress.

config Boot-time configuration for TubePress.

themes Stores any custom TubePress themes.

Installing Add-Ons & Themes

To install an add-on or theme, simply unzip into the `add-ons` or `themes` subdirectory respectively. For instance, if you have downloaded an add-on named `some_tubepress_feature.zip`, you would unzip into the `add-ons` directory so that the filesystem path of the add-on would look something like `tubepress-content/add-ons/some_tubepress_feature/`. The next time TubePress is invoked, it will detect and load the new files.

Advanced Topics

On This Page

- [Boot Cache](#)
 - [Enabling](#)
 - [Clearing the Cache](#)
 - [Detailed Explanation](#)

Boot Cache

Starting with TubePress 3.1.0, TubePress can take advantage of a “boot cache” that can dramatically speed up TubePress. In our lab testing, we typically see a **300% - 400% reduction in execution time** with the boot cache enabled.

Enabling Enabling the cache is easy; you'll just need a text editor. Here's how to do it...

1. Ensure that TubePress debugging is enabled so that you can verify that the cache is working.

In standalone PHP environments, this is enabled by default. In WordPress environments, ensure the box at WP Admin > Settings > TubePress > Advanced > Enable Debugging is ticked.

2. TubePress's boot process is controlled by a single file located within your TubePress content directory at `config/boot.json`. In TubePress 3.1.0+, this file and the `config` directory are included by default. In older versions of TubePress, simply create `tubepress-content/config/boot.json` and fill it with the following contents:

```
{
  "cache" : {
    "ioc-container" : {
      "enabled" : false
    },
    "add-ons" : {
      "enabled" : false
    },
    "classloader" : {
      "enabled" : false
    },
    "option-descriptors" : {
      "enabled" : false
    }
  }
}
```

```

    },
    "killer-key" : "tubepress_boot_cache_kill",
    "dir" : null
  },
  "add-ons" : {
    "blacklist": []
  },
  "classloader" : {
    "enabled" : true
  }
}

```

3. Replace each instance of `false` with `true`, and set the value of `killer-key` to a random string. Your copy of `tubepress-content/config/boot.json` should now look something like this:

```

{
  "cache" : {
    "ioc-container" : {
      "enabled" : true
    },
    "add-ons" : {
      "enabled" : true
    },
    "classloader" : {
      "enabled" : true
    },
    "option-descriptors" : {
      "enabled" : true
    },
    "killer-key" : "tIEKrw84k7z760811D815363425xa15370W",
    "dir" : null
  },
  "add-ons" : {
    "blacklist": []
  },
  "classloader" : {
    "enabled" : true
  }
}

```

4. Verify that the boot cache is working by examining your TubePress debug output. You should see something similar to the following:

```

Default Boot Config Service: Attempting to read boot config from /var/www/ttg.lan/wordpress/
Default Boot Config Service: Successfully read boot config from /var/www/ttg.lan/wordpress/w
Default Boot Config Service: classloader caching is enabled
Default Boot Config Service: add-ons caching is enabled
...
Default Add-on Discoverer: Successfully hydrated from cache file at ... /serialized-addons.t
Default Boot Config Service: ioc-container caching is enabled
...
Default IOC Boot Helper: Successfully hydrated from cache file at ... /cached-ioc-container.
...
Default Boot Config Service: option-descriptors caching is enabled
Default Option Descriptor Reference: Successfully hydrated from cache file at ... /serialize
...
TubePress Bootstrapper: Boot completed in 28.892893 milliseconds

```

If your debug output is missing any of the highlighted phrases, it means that something is misconfigured. Feel

free to post a question in [the forum](#) to get help.

Clearing the Cache After adding or removing TubePress add-ons, you'll need to clear the boot cache so TubePress can recognize the changes to your installation.

You can clear the cache using any web browser, and you'll simply need to know the value of `killer-key` that was set in your `config/boot.json`. As an example, suppose that the value of `killer-key` is `abc123`, then you would add `abc123=true` to the end of any URL where TubePress is used to clear the cache. For instance,

```
http://myblog.com/videos?abc123=true
```

Detailed Explanation

Note: The rest of this section details the contents of `config/boot.json` for advanced users. Most users can skip this section.

Let's examine each piece of the file...

```
"cache" : {
  "ioc-container" : {
    "enabled" : true
  },
  "add-ons" : {
    "enabled" : true
  },
  "classloader" : {
    "enabled" : true
  },
  "option-descriptors" : {
    "enabled" : true
  }
}
```

This section of the file enables or disables the caching of individual elements of TubePress's internals. Most users will set all of these elements to either `true` or `false`.

```
"killer-key" : "tubepress_boot_cache_kill",
```

The killer key" can be used to remotely and securely clear the boot cache. The value of `killer-key` can be used as a query parameter to signal to TubePress to clear the boot cache. e.g. if the value of `killer-key` is `456xyz`, then `456xyz=true` to the URL of a page using TubePress will clear the entire boot cache.

```
"dir" : null
```

The `dir` option allows you to manually configure a directory where TubePress will store its boot cache. If you leave its value as `null`, TubePress will attempt to use the system's cache directory.

```
"add-ons" : {
  "blacklist": []
},
```

This section allows you to identify, by name, a set of add-ons that will be excluded from TubePress. If you are not using a particular add-on, adding it to the blacklist will improve TubePress's performance.

```
"classloader" : {
  "enabled" : true
}
```

By default, TubePress uses its own high-performance PSR-0 compliant class loader. If you would like to use a class loader defined elsewhere, you can set this value to `false`.

FAQ & Troubleshooting

On This Page

- [Common Problems](#)
 - [JavaScript Errors](#)
 - [Cross-Domain Ajax](#)
 - [Missing `tubepress.js`](#)
 - [Multiple Instances of jQuery](#)
 - [WordPress’s Visual Editor Interference](#)
 - [API Latency](#)
 - [Locked-Down PHP Installation](#)
- [Troubleshooting Tools](#)
 - [Debugging Mode](#)
 - [Firebug](#)

Common Problems

Use the table below to find a solution to your problem.

Problem	Possible causes and solutions
Video thumbnails are “unclickable”	<ul style="list-style-type: none">• <i>JavaScript Errors</i>• <i>Missing <code>tubepress.js</code></i>• <i>Multiple Instances of jQuery</i>
After clicking thumbnail, video never loads	<i>Cross-Domain Ajax</i>
Layout is unstyled or ugly	<i>WordPress’s Visual Editor Interference</i>
My videos aren’t showing up with TubePress	<i>API Latency</i>
“No HTTP transports could execute ...” message	<i>Locked-Down PHP Installation</i>

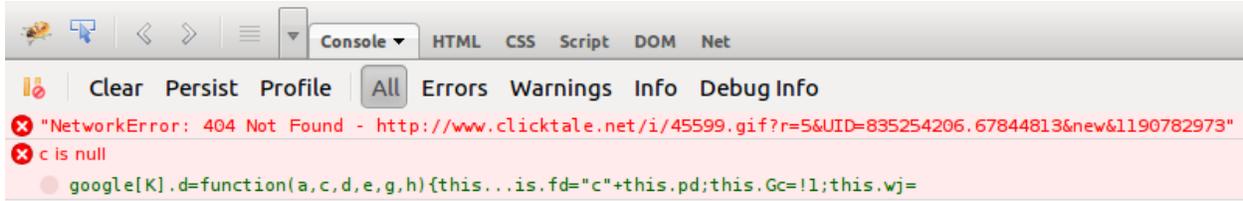
JavaScript Errors If your website has JavaScript errors, they can prevent TubePress from operating correctly.

TubePress expects your page to load cleanly; i.e. there should be no JavaScript errors caused by other scripts, plugins, themes, etc. The reason is that TubePress relies on jQuery’s `ready()` function, which can get interrupted if other scripts cause errors.

To diagnose if this is your problem, use the following procedure:

1. Using Firefox, navigate to a page with a TubePress gallery
2. Open up Firebug.
3. Open Firebug’s “Console” tab
4. Refresh the page.
5. Check Firebug’s console for JavaScript errors

The image below shows what JavaScript errors look like in Firebug.



The resolution is always to fix the script causing the error. Usually this entails identifying which component of your site (script, plugin, etc) is causing the error. Unfortunately since sites can get extremely complicated, there’s no “one size fits all” solution.

Typically by hovering over the JavaScript errors in Firebug, you can pinpoint which JavaScript file is causing the trouble.

Cross-Domain Ajax TubePress uses Ajax extensively in the browser - e.g. to asynchronously load the video player whenever a user clicks a thumbnail. By virtue of this, TubePress is forced to adhere to the same origin policy of JavaScript. The same origin policy should be embraced - at least until the Internet settles on a new policy - as it a crucial guard against malicious sites.

However, this means that a small domain misconfiguration can lead to a broken/misbehaving TubePress installation.

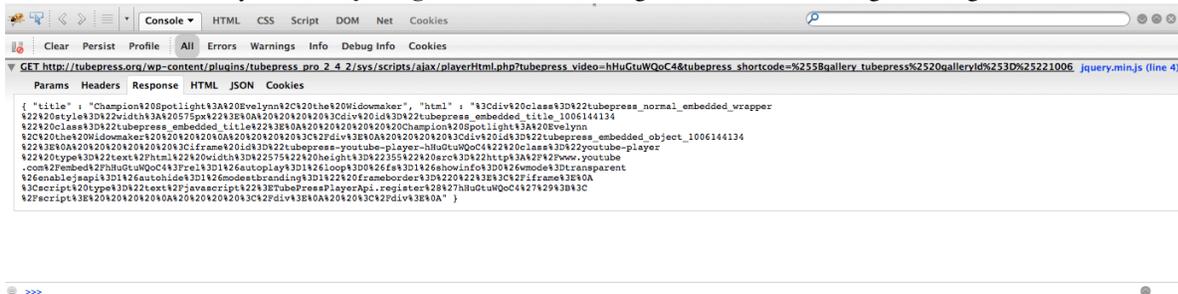
As an example, say that you have a domain `ehough.com` and are serving a web site at this address; loading up `http://ehough.com` works normally for your users, and TubePress detects that `ehough.com` is the domain that it will use. But you *also* have the site available (without HTTP redirects) via `http://www.ehough.com`. Users that access the site via the `www.ehough.com` domain will experience a broken TubePress installation, as TubePress will attempt to make a cross-domain Ajax request to `ehough.com`.

To diagnose if this if your problem, use the following procedure:

1. Using Firefox, navigate to a page with a TubePress gallery
2. Open up Firebug
3. Open Firebug’s “Console” tab
4. Click on any TubePress thumbnail. You should see a new XHR request show up as TubePress performs its Ajax request.



5. Expand this request (using the expand icon on the left) to view the results. You should see a small/medium JSON result. If you see **anything** else here, or nothing at all, then something is wrong

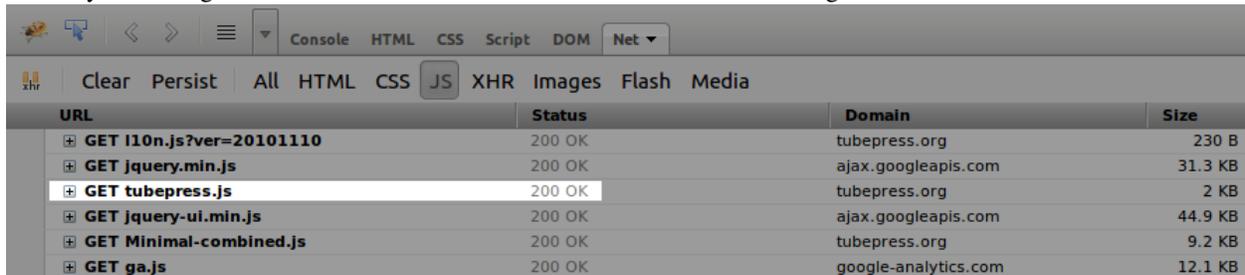


To fix this issue, [Google recommends](#) to choose a “preferred domain” and stick with it. Here are some guidelines on how to achieve this:

- Use HTTP 301 redirects (“RedirectPermanent”) in your `.htaccess` file to smartly redirect users to your preferred domain. * In Apache, you can do this with an `.htaccess` file. * In IIS, you can do this through the administrative console.
- WordPress users should make sure that both their “Site Address” and “WordPress Address” match their preferred domain (you can set these at WP Admin > Settings > General)
- If you use TubePress Pro in a PHP environment, make sure that the domain you set with `TubePressPro::setBaseUrl()` matches your preferred domain

Missing `tubepress.js` Any page that uses TubePress needs to have `tubepress.js` loaded somewhere in the document. If this file is missing, your gallery’s thumbnails won’t respond to clicks.

To diagnose if this is your problem, you can view your HTML source or use Firebug to verify if this file is being loaded correctly. The image below shows what a correct load will look like in Firebug.



The screenshot shows the Firebug 'Net' tab with a table of network requests. The 'GET tubepress.js' request is highlighted.

URL	Status	Domain	Size
GET i10n.js?ver=20101110	200 OK	tubepress.org	230 B
GET jquery.min.js	200 OK	ajax.googleapis.com	31.3 KB
GET tubepress.js	200 OK	tubepress.org	2 KB
GET jquery-ui.min.js	200 OK	ajax.googleapis.com	44.9 KB
GET Minimal-combined.js	200 OK	tubepress.org	9.2 KB
GET ga.js	200 OK	google-analytics.com	12.1 KB

In WordPress, this file will be automatically loaded via your theme’s `wp_head()` function invocation. However, some custom themes will (accidentally or intentionally) omit this call. Switching back to the “Twenty Thirteen” WordPress theme will verify if your custom theme is the culprit. [Here](#) is some further information on including the `wp_head()` function in your theme.

Multiple Instances of jQuery This problem is much more prevalent in WordPress-based sites, where other plugin and theme authors fail to include jQuery “the right way”. Multiple loads of jQuery will not only significantly hurt the load-time of your site, but it will also break TubePress.

There are two ways to see if your site is loading jQuery more than once:

1. Inspect your HTML source, hit `Ctrl-F` and search for “jquery”. You should only see one JavaScript file loaded for jQuery.
2. Use Firebug’s “Net” tab to examine the scripts loaded on your page. The image below shows what multiple jQuery loads looks like in Firebug.

URL	Status	Domain
▶ GET www-embed_core_module-vfl	200 OK	s.yimg.com
▶ GET tubeppress.js?ver=3.3.1	304 Not Modified	s.yimg.com
▶ GET swfobject.js	304 Not Modified	s.yimg.com
▶ GET shutter-reloaded.js?ver=1.3.3	304 Not Modified	s.yimg.com
▶ GET ngg.slideshow.min.js?ver=1.05	304 Not Modified	s.yimg.com
▶ GET jquery.min.js	304 Not Modified	jquery.com
▶ GET jquery.js?ver=1.7.1	304 Not Modified	s.yimg.com
▶ GET jquery.equalcolumns.js	304 Not Modified	s.yimg.com
▶ GET jquery.cycle.all.min.js?ver=2.9.1	304 Not Modified	s.yimg.com
▶ GET AC_RunActiveContent.js	304 Not Modified	s.yimg.com

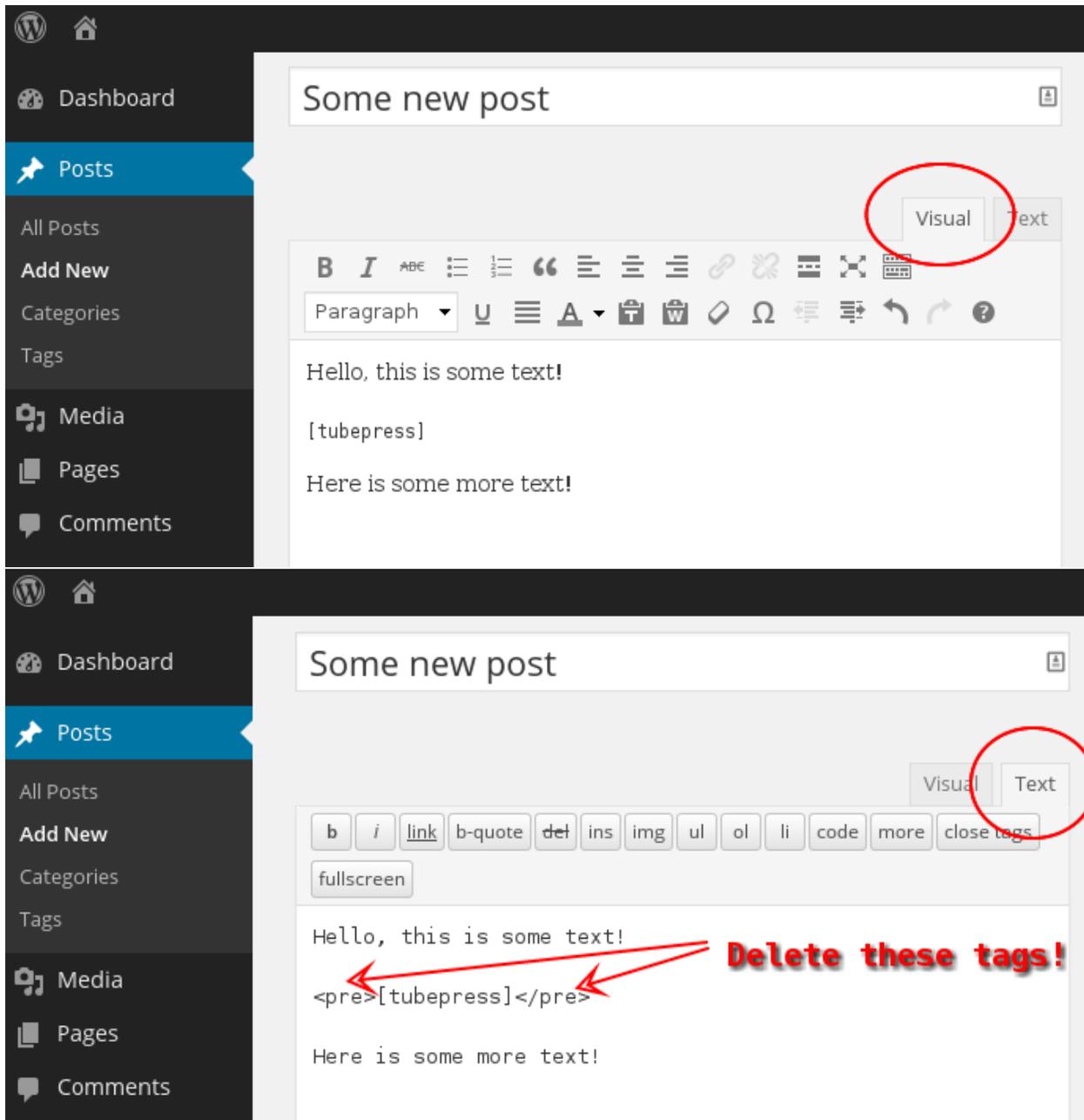
10 requests

The solution in this case is to pare down your scripts to a single jQuery load. How this solution is accomplished depends on your environment.

- In TubePress Pro for PHP, this will usually involve simply editing your site’s header to only include jQuery once.
- In TubePress for WordPress, typically the best method is to disable your plugins one-by-one until you find the offending plugin and/or theme. At that point, you’ll need to either edit the plugin/theme’s PHP or ask the original author to include jQuery “the right way”.

WordPress’s Visual Editor Interference When authoring a post or a page from your WordPress administrative dashboard, WordPress allows you to use both the [Visual Editor](#) as well as the [Text Editor](#). Unfortunately the Visual Editor sometimes “wraps” TubePress shortcodes with HTML `<pre>` or `<code>` tags. This can ruin the appearance of TubePress, often resulting in incorrect font faces or “diagonal” thumbnail layouts.

The solution is to use WordPress’s Text Editor and remove any `<pre>` or `<code>` tags near your `[tubeppress]` shortcodes.



API Latency What you see on youtube.com or vimeo.com will not be immediately and exactly identical to what's available to TubePress. There is often a caching delay before videos are available over the API (which TubePress uses). It's almost always less than an hour, but has been known to take up to an entire day. The punchline here is: **it can take up to a day for your gallery to sync up to the video provider.**

Please read [YouTube's explanation](#) for more details regarding the API delay for YouTube.

Locked-Down PHP Installation If you get an error message from TubePress that looks like

No HTTP transports could execute GET to ...

it means that your PHP installation has been configured to block nearly all network operations. This is often done as a security measure to prevent malicious PHP scripts from "phoning home."

TubePress needs to open HTTP connections to talk to video providers like YouTube or Vimeo. Under the hood, TubePress uses the [shortstop HTTP client](#), which attempts to use the following PHP mechanisms:

- [cURL extension](#)
- [HTTP extension](#)
- `fsockopen()`
- `fopen()`
- [Streams extension](#)

So the solution is to allow TubePress to use one or more of these mechanisms. You, or your hosting provider, must perform **at least one** of the following:

- Install and allow the [cURL extension](#)
- Install and allow the [HTTP extension](#)
- Remove `fsockopen()` from the list of [disabled functions](#)
- Remove `fopen()` from the list of [disabled functions](#)

Troubleshooting Tools

Debugging Mode TubePress's debug mode will print a huge amount of logging statements to the screen. It will describe in (extreme) detail what TubePress is doing behind the scenes.

```
[2014-01-09 17:57:37] [DEBUG] Default Boot Config Service: option-descriptors caching is enabled
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Attempting read contents of cache file at /var/
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Attempting to hydrate from cache file at /var/w
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Successfully hydrated from cache file at /var/w
[2014-01-09 17:57:37] [DEBUG] TubePress Bootstrapper: Boot completed in 71.796894 milliseconds
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Found a shortcode: [gallery_tubepress mode="user" userValue=
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Candidate options detected in shortcode: [gallery_tubepress mo
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: mode = "user" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: mode = "user" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: userValue = "PenWeddings" (unfiltre
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: userValue = "PenWeddings" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: youtubeHideBlackBars = "true" (unfilt
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: youtubeHideBlackBars = "1" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: resultsPerPage = "12" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: resultsPerPage = "12" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbWidth = "144" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbWidth = "144" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbHeight = "81" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbHeight = "81" (filtered)
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: mode = user
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: userValue = PenWeddings
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: youtubeHideBlackBars = 1
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: resultsPerPage = 12
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: thumbWidth = 144
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: thumbHeight = 81
[2014-01-09 17:57:39] [DEBUG] Search Output Shortcode Handler: Not configured for search results
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: galleryId = 1478746967
[2014-01-09 17:57:39] [DEBUG] Thumb Gallery Shortcode Handler: Starting to build thumbnail gallery 1478746967
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Attempting to load template instance from gallery.tpl.php with fallb
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Requested theme is 'default'
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Found gallery.tpl.php first try at /var/www/front/wordpress/wp-conte
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Successfully loaded template from /var/www/front/wordpress/wp-cor
```

To use it, you just need to add `tubepress_debug=true` to the URL string (in your browser's address bar) of any page that uses TubePress on it. For instance, if the TubePress page you're trying to debug has an address of

```
http://ehough.com/?page_id=19
```

then you should add `tubepress_debug=true` to the URL to enable debugging:

```
http://ehough.com/?page_id=19&tubepress_debug=true
```

Most of the debug output is self-explanatory. If the output is cut off unexpectedly, it's likely that TubePress encountered a fatal error. Check your PHP error logs for more information.

Firebug One particular tool which is invaluable in debugging client-side problems is [Firebug](#). Firebug is an add-on for Firefox that gives you x-ray vision for any website. It's powerful, yet easy to use. If you're new to Firebug, here are some introductory resources to help you learn.

What is Firebug? From the official Firebug site

Introductory screencast Shows the general concepts and use-cases of Firebug.

Another introductory screencast A longer, more detailed video going over installation and usage.

Firebug Guide for Web Developers A detailed, well-written overview of Firebug (with pictures).

2.2 TubePress Manual for PHP

TubePress Pro is available to use as a mature PHP library, which makes it accessible to nearly any website that runs PHP.

2.2.1 Manual Contents

Install / Upgrade

This page details how to install, upgrade, and uninstall TubePress when used as a standalone PHP library.

On this page

- [Requirements](#)
- [Install](#)
 - [Vimeo API Setup](#)
 - [Optional TubePress Pro Installations](#)
 - * [JW Player \(version 5.x\)](#)
 - * [Shadowbox.js](#)
- [Upgrade](#)

Requirements

It's a good idea to first make sure that your web server meets the minimum requirements to run TubePress. You will need:

- [PHP 5.2](#) or higher
- [PHP DOM extension](#) (enabled by default in PHP, only required for using TubePress with YouTube)
- [PHP PCRE extension](#) built with PCRE version 7.1 or higher (enabled in most PHP installations)

Additionally, if you intend to use TubePress in any language other than English, TubePress for PHP requires the [PHP Multibyte String extension](#) (enabled in most PHP installations).

Not sure if your server meets these requirements? You should either

1. Ask your hosting provider (good idea to include a link to this page!)
2. Use our PHP file which checks your server for the requirements. Here's how to use it:
 1. Using a web browser, visit [this URL](#). You'll see some PHP source code.
 2. Save the source code to disk (File > Save from your web browser's menu, or just hit Ctrl-S). It should be saved somewhere on your local machine, and the name of the file should be `check.php`.
 3. Upload `check.php` to your website's document root.
 4. Using a web browser, visit `http://<yourdomain.com>/check.php`. The output of this page should be self-explanatory and will look similar to the following:

Check TubePress requirements...

Check (1 / 4): PHP version... **OK** (detected PHP version 5.4.3)

Check (2 / 4): DOM extension... **OK** (detected dom extension version 20031129)

Check (3 / 4): PCRE extension... **WARN** (pcre detected, but unable to determine version - please manually verify the version below.)

Check (4 / 4): mbstring extension... **WARN** (mbstring extension is missing, but this extension is only required if you intend to run TubePress Pro in a standalone PHP environment)

Checks passed, but there were warnings. This server *might* meet the requirements for TubePress

PHP Version 5.4.3 	
System	Linux web200.webfaction.com 2.6.18-274.18.1.el5PAE #1 SMP Thu Feb 9 13:25:50 EST 2012 i686
Build Date	May 14 2012 12:57:55
Configure Command	./configure '--host=i686-redhat-linux-gnu' '--build=i686-redhat-linux-gnu' '--target=i386-redhat-linux' '--program-prefix=' '--program-suffix=54' '--prefix=/usr/local' '--with-pdo-mysql' '--with-pdo-pgsql' '--enable-bcmath' '--enable-calendar' '--enable-exif' '--enable-ftp' '--enable-mbstring' '--enable-soap' '--enable-

Install

Here's how to install TubePress Pro when used as a standalone PHP library.

1. If you haven't already, [purchase a TubePress Pro license](#).
2. Download TubePress Pro from [here](#).
3. Unzip the file you just downloaded (tube`press_pro_x_y_z`.zip). It will expand into a directory named tube`press_pro_x_y_z`. (Note: x, y, and z are the major, minor, and micro version numbers of TubePress).
4. Upload the entire tube`press_pro_x_y_z` directory to your web server in a place that's *web accessible*.
5. Make a note of the following two items, which you'll need when invoking TubePress:
 - the full path of this directory on your web server's filesystem (e.g. /var/www/html/myblog.com/lib/tube`press_pro_x_y_z`)
 - the web-accessible URL to reach this directory (e.g. [http://myblog.com/lib/tube`press_pro_x_y_z`](http://myblog.com/lib/tube<code>press_pro_x_y_z</code>))
6. *Vimeo users only*. Perform the *one-time Vimeo API setup*.
7. *Optional*. *Install support for Shadowbox.js and/or JW Player 5*.

Vimeo API Setup

Note: If you aren't using Vimeo, you can safely skip this section.

In order to use TubePress with Vimeo, you are required to obtain a “consumer key” and “consumer secret” from Vimeo. This is a one-time setup required by Vimeo. Thankfully it's easy to create these credentials and supply them to TubePress.

1. [Log in to Vimeo](#), if you haven't already.
2. [Click here](#) to create a new Vimeo app. You can fill in anything for the fields. Below is a sample form that's filled out. Then click the *Create App* button.

My Advanced API Apps /

Create a new App

App Name *

my amazing website

App Description *

When users authenticate via your app, this is what they'll see.

Use TubePress within my website.

App URL *

Your app's publicly accessible home page where users can get more information or download your app.

http://foo.com

App Logo URL

The URL of a logo to show with your app when authenticating. Displayed at 100x100 px.

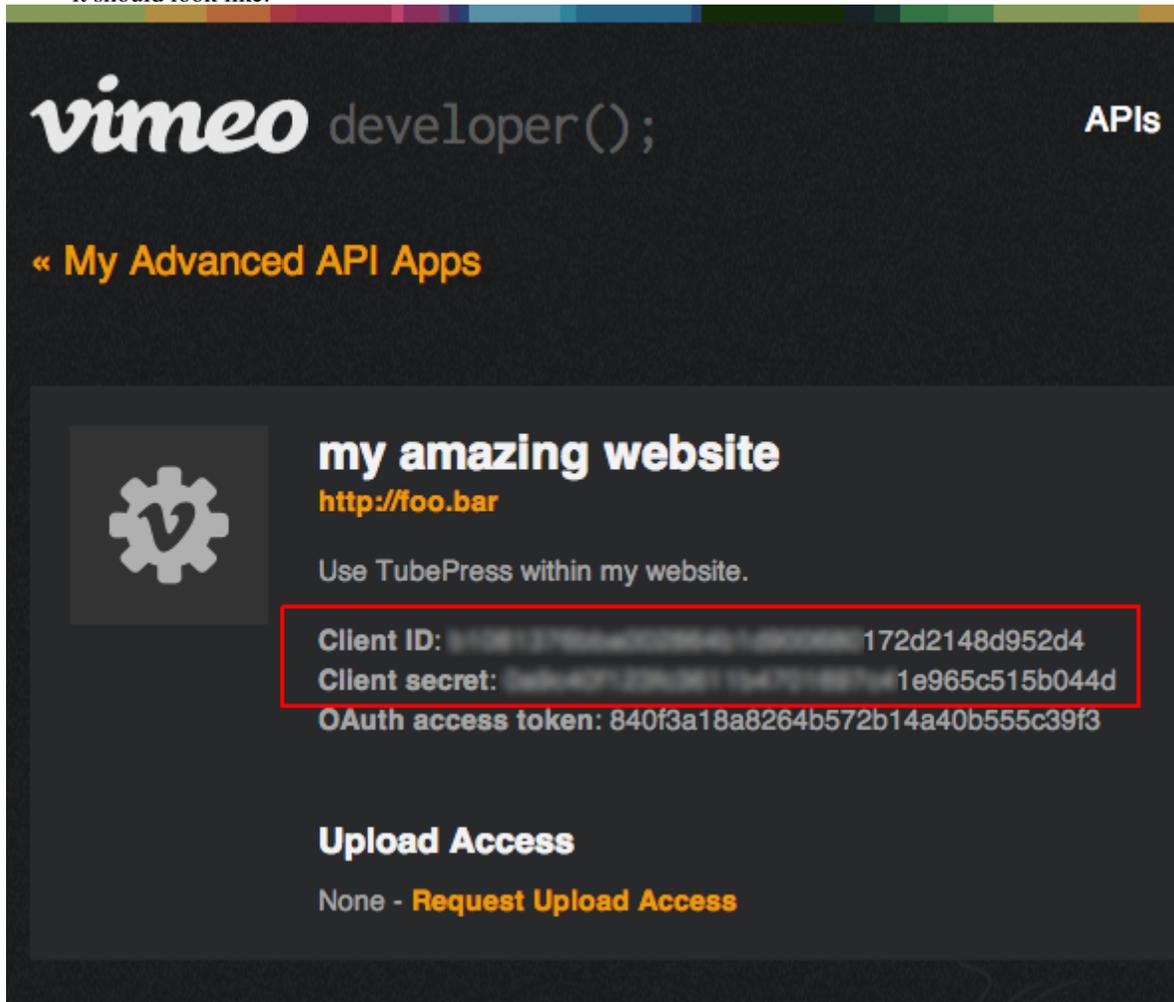
App Callback URL

This is the URL we'll send a user back to when authenticating.

I agree that my application does not violate the [Vimeo API License Agreement](#) or the [Vimeo Terms of Service](#)

* Required

3. Make a note of the “Client ID” and “Client secret” of your new app. See the red box below for an idea of what it should look like.



That’s it! You now simply need to provide these credentials to TubePress. You’ll have to provide the key and secret via the *vimeoKey* and *vimeoSecret* options in TubePress. You can pass these into the `getHtmlForShortcode()` function:

```
<?php print TubePressPro::getHtmlForShortcode('vimeoKey="M7e8oGsvU8Ld4TW7e8oLsvU8Ld4TM7e8oLs" vimeoSecret="172d2148d952d4"')
```

Make sure not to mix up the “key” and the “secret”!

Optional TubePress Pro Installations Due to licensing restrictions, TubePress Pro does not ship with every library that the free WordPress plugin includes. However it’s very easy to install these optional libraries manually. This section provides installation instructions.

JW Player (version 5.x) TubePress can use JW Player 5.10 to play YouTube videos. We are working to integrate JW Player 6.x, but this feature is not yet ready for production.

1. Download the file `player.swf` from [here](#).
2. Upload the file to your TubePress Pro installation at

```
<tubepress_home>/src/main/web/flash/longtail/player.swf.
```

Shadowbox.js

1. Download Shadowbox.js from [here](#). The image below shows the required options when configuring your download. **You may select additional checkboxes (though not recommended), but make sure that at least the settings shown are chosen.**

Shadowbox.js

[Home](#)
[Download](#)
[Support](#)
[Usage](#)
[Options](#)
[API](#)
[FAQ](#)

This page allows you to create a custom build of Shadowbox that includes only the features you need for your project. Compiling all of the necessary source files into one improves performance and

Choose the [adapter](#) you'll be using:

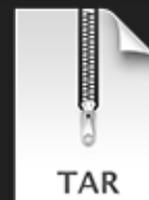
Select the player(s) for the types of content you want to display:

- Images
- External sites and pages
- Inline and/or dynamic HTML
- Flash (swf files)
- Flash video (flv files)
- QuickTime
- Windows Media Player

Choose the language support you need:

- Include support for using CSS selectors to select links

The current version of Shadowbox is
3.0.3



2. The download will save to a file named something like `shadowbox-3.0.3.zip`. Unzip this file, and it will expand into a directory named `shadowbox-3.0.3`.
3. Copy the *contents* of this directory (not the directory itself) to your TubePress Pro installation at

```
<tubepress_home>/src/main/web/players/shadowbox/lib/.
```

4. *Optional.* You may see scrollbars in the Shadowbox.js display. To fix this, simply edit `lib/shadowbox.css`. On line 8, remove the `overflow:auto` attribute. i.e. change it from

```
#sb-player.html{height:100%;overflow:auto;}
```

to

```
#sb-player.html{height:100%;}
```

Upgrade

The easiest and most reliable way to upgrade TubePress Pro is to perform a side-by-side installation with the new version alongside the old. Here's the exact process.

1. Repeat the TubePress Pro installation procedure with the latest copy of TubePress Pro.
2. *Optional.* Copy over the TubePress Content Directory from your old TubePress installation (`tubepress_pro_x_y_z>/tubepress-content`). This is only required if you are using custom TubePress add-ons or themes.
3. In the PHP files where you use TubePress Pro, edit `TubePressPro::setBaseUrl()` and the path to `TubePressPro.php` to point to your new install location.

Using TubePress in PHP

On This Page

- [Add TubePress to any PHP Page](#)
- [Changing the Default Language](#)

Add TubePress to any PHP Page

Follow these instructions to use TubePress Pro in a standalone PHP environment. You can integrate TubePress into any PHP page on your site. For demonstration purposes, we will use the very simple PHP/HTML file show below. Feel free to create this file, or extrapolate the concepts into your existing PHP. As you can see by the highlighted lines of code, there are **four** key points.

```
1 <?php
2
3     include "/var/www/html/myblog.com/lib/tubepress_pro_x_y_z/src/main/php/classes/TubePressPro.php",
4
5     TubePressPro::setBaseUrl("http://myblog.com/lib/tubepress_pro_x_y_z");
6 ?>
7
8 <html>
9     <head>
10        <title>TubePress Pro</title>
11
12        <?php print TubePressPro::getHtmlForHead(true); ?>
13
14    </head>
15    <body>
```

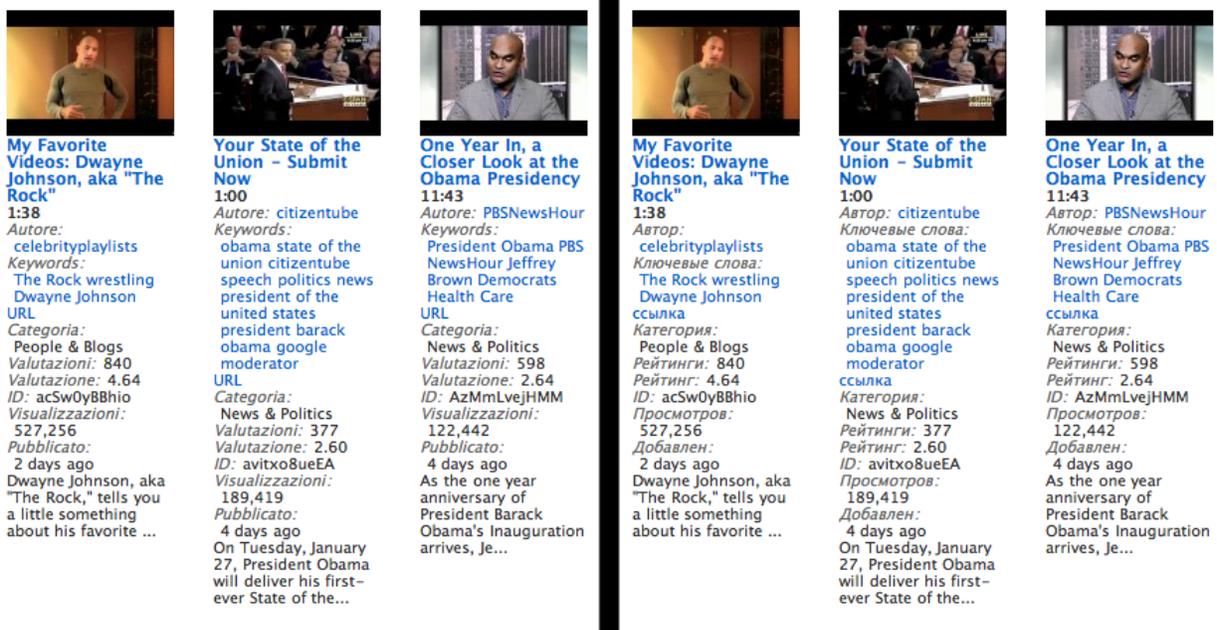
16
17
18
19
20

```
<?php print TubePressPro::getHtmlForShortcode('mode="tag" tagValue="pittsburgh steelers" resu
</body>
</html>
```

1. Include the TubePress Pro class definition (TubePressPro.php). An absolute path works best.
2. Call TubePressPro::setBaseUrl() with the *web-accessible* URL of your TubePress Pro installation.
3. Include this statement in the HTML head of your document to print out the required TubePress CSS and JavaScript libraries. It takes a single parameter which indicates whether or not to include jQuery. If you are already including jQuery in your document, use `false` for this parameter.
4. The `getHtmlForShortcode()` function takes a single string parameter which is any valid TubePress shortcode. It returns the HTML output of TubePress. You may repeat this function call as many times as you like.

Changing the Default Language

TubePress is available in several languages besides English. The image below is an example of TubePress when configured for Italian (left) and Russian (right).



You can tell TubePress which language to use via the `lang` option. Simply supply your two-letter language and country code:

```
<?php
TubePressPro::getHtmlForShortcode(' lang="it_IT"');
```

Using Add-Ons & Themes

On This Page

- [What are Add-Ons & Themes?](#)
- [TubePress Content Directory](#)
- [Installing Add-Ons & Themes](#)

What are Add-Ons & Themes?

TubePress “add-ons” are small software packages that add or change the default functionality of TubePress. TubePress “themes” are template sets that can change the layout and/or appearance of TubePress.

The central source for all TubePress add-ons and themes is the [TubePress Marketplace](#). There you can browse and download add-ons and themes developed by both TubePress LLC as well as community contributions.

TubePress Content Directory

Both add-ons and themes are stored in the “TubePress Content Directory”, which is a special directory inside your TubePress installation located at `tubepress_pro_x_y_z/tubepress-content`.

The TubePress Content Directory has a few rules. First, it must always be named `tubepress-content`. It must also have the following immediate subdirectories:

add-ons TubePress add-ons that aren’t shipped with TubePress.

config Boot-time configuration for TubePress.

themes Stores any custom TubePress themes.

Installing Add-Ons & Themes

To install an add-on or theme, simply unzip into the `add-ons` or `themes` subdirectory respectively. For instance, if you have downloaded an add-on named `some_tubepress_feature.zip`, you would unzip into the `add-ons` directory so that the filesystem path of the add-on would look something like `tubepress-content/add-ons/some_tubepress_feature/`. The next time TubePress is invoked, it will detect and load the new files.

Advanced Topics

On This Page

- [Boot Cache](#)
 - [Enabling](#)
 - [Clearing the Cache](#)
 - [Detailed Explanation](#)

Boot Cache

Starting with TubePress 3.1.0, TubePress can take advantage of a “boot cache” that can dramatically speed up TubePress. In our lab testing, we typically see a **300% - 400% reduction in execution time** with the boot cache enabled.

Enabling Enabling the cache is easy; you'll just need a text editor. Here's how to do it...

1. Ensure that TubePress debugging is enabled so that you can verify that the cache is working.

In standalone PHP environments, this is enabled by default. In WordPress environments, ensure the box at WP Admin > Settings > TubePress > Advanced > Enable Debugging is ticked.

2. TubePress's boot process is controlled by a single file located within your TubePress content directory at `config/boot.json`. In TubePress 3.1.0+, this file and the config directory are included by default. In older versions of TubePress, simply create `tubepress-content/config/boot.json` and fill it with the following contents:

```
{
  "cache" : {
    "ioc-container" : {
      "enabled" : false
    },
    "add-ons" : {
      "enabled" : false
    },
    "classloader" : {
      "enabled" : false
    },
    "option-descriptors" : {
      "enabled" : false
    },
    "killer-key" : "tubepress_boot_cache_kill",
    "dir" : null
  },
  "add-ons" : {
    "blacklist": []
  },
  "classloader" : {
    "enabled" : true
  }
}
```

3. Replace each instance of `false` with `true`, and set the value of `killer-key` to a random string. Your copy of `tubepress-content/config/boot.json` should now look something like this:

```
{
  "cache" : {
    "ioc-container" : {
      "enabled" : true
    },
    "add-ons" : {
      "enabled" : true
    },
    "classloader" : {
      "enabled" : true
    },
    "option-descriptors" : {
      "enabled" : true
    },
    "killer-key" : "tIEKrw84k7z760811D815363425xa15370W",
    "dir" : null
  },
  "add-ons" : {
    "blacklist": []
  },
}
```

```
"classloader" : {
    "enabled" : true
}
}
```

4. Verify that the boot cache is working by examining your TubePress debug output. You should see something similar to the following:

```
Default Boot Config Service: Attempting to read boot config from /var/www/ttg.lan/wordpress/
Default Boot Config Service: Successfully read boot config from /var/www/ttg.lan/wordpress/
Default Boot Config Service: classloader caching is enabled
Default Boot Config Service: add-ons caching is enabled
...
Default Add-on Discoverer: Successfully hydrated from cache file at ... /serialized-addons.t
Default Boot Config Service: ioc-container caching is enabled
...
Default IOC Boot Helper: Successfully hydrated from cache file at ... /cached-ioc-container.
...
Default Boot Config Service: option-descriptors caching is enabled
Default Option Descriptor Reference: Successfully hydrated from cache file at ... /serialize
...
TubePress Bootstrapper: Boot completed in 28.892893 milliseconds
```

If your debug output is missing any of the highlighted phrases, it means that something is misconfigured. Feel free to post a question in [the forum](#) to get help.

Clearing the Cache After adding or removing TubePress add-ons, you'll need to clear the boot cache so TubePress can recognize the changes to your installation.

You can clear the cache using any web browser, and you'll simply need to know the value of `killer-key` that was set in your `config/boot.json`. As an example, suppose that the value of `killer-key` is `abc123`, then you would add `abc123=true` to the end of any URL where TubePress is used to clear the cache. For instance,

```
http://myblog.com/videos?abc123=true
```

Detailed Explanation

Note: The rest of this section details the contents of `config/boot.json` for advanced users. Most users can skip this section.

Let's examine each piece of the file...

```
"cache" : {
    "ioc-container" : {
        "enabled" : true
    },
    "add-ons" : {
        "enabled" : true
    },
    "classloader" : {
        "enabled" : true
    },
    "option-descriptors" : {
        "enabled" : true
    }
}
```

This section of the file enables or disables the caching of individual elements of TubePress’s internals. Most users will set all of these elements to either `true` or `false`.

```
"killer-key" : "tubepress_boot_cache_kill",
```

The killer key” can be used to remotely and securely clear the boot cache. The value of `killer-key` can be used as a query parameter to signal to TubePress to clear the boot cache. e.g. if the value of `killer-key` is `456xyz`, then `456xyz=true` to the URL of a page using TubePress will clear the entire boot cache.

```
"dir" : null
```

The `dir` option allows you to manually configure a directory where TubePress will store its boot cache. If you leave its value as `null`, TubePress will attempt to use the system’s cache directory.

```
"add-ons" : {
    "blacklist": []
},
```

This section allows you to identify, by name, a set of add-ons that will be excluded from TubePress. If you are not using a particular add-on, adding it to the blacklist will improve TubePress’s performance.

```
"classloader" : {
    "enabled" : true
}
```

By default, TubePress uses its own high-performance [PSR-0](#) compliant class loader. If you would like to use a class loader defined elsewhere, you can set this value to `false`.

FAQ & Troubleshooting

On This Page

- Common Problems
 - JavaScript Errors
 - Cross-Domain Ajax
 - API Latency
 - Locked-Down PHP Installation
- Troubleshooting Tools
 - Debugging Mode
 - Firebug

Common Problems

Use the table below to find a solution to your problem.

Problem	Possible causes and solutions
Video thumbnails are “unclickable”	<i>JavaScript Errors</i>
After clicking thumbnail, video never loads	<i>Cross-Domain Ajax</i>
My videos aren’t showing up with TubePress	<i>API Latency</i>
“No HTTP transports could execute …” message	<i>Locked-Down PHP Installation</i>

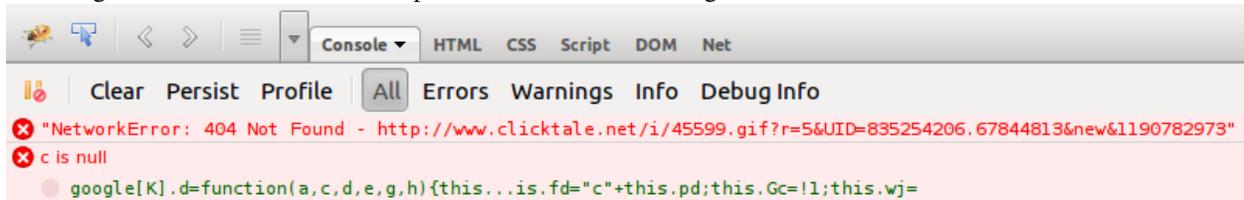
JavaScript Errors If your website has JavaScript errors, they can prevent TubePress from operating correctly.

TubePress expects your page to load cleanly; i.e. there should be no JavaScript errors caused by other scripts, plugins, themes, etc. The reason is that TubePress relies on jQuery's `ready()` function, which can get interrupted if other scripts cause errors.

To diagnose if this is your problem, use the following procedure:

1. Using Firefox, navigate to a page with a TubePress gallery
2. Open up Firebug.
3. Open Firebug's "Console" tab
4. Refresh the page.
5. Check Firebug's console for JavaScript errors

The image below shows what JavaScript errors look like in Firebug.



The resolution is always to fix the script causing the error. Usually this entails identifying which component of your site (script, plugin, etc) is causing the error. Unfortunately since sites can get extremely complicated, there's no "one size fits all" solution.

Typically by hovering over the JavaScript errors in Firebug, you can pinpoint which JavaScript file is causing the trouble.

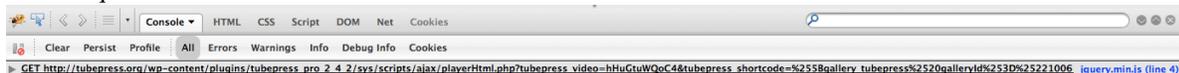
Cross-Domain Ajax TubePress uses [Ajax](#) extensively in the browser - e.g. to asynchronously load the video player whenever a user clicks a thumbnail. By virtue of this, TubePress is forced to adhere to the [same origin policy](#) of JavaScript. The same origin policy should be embraced - at least until the Internet settles on a [new policy](#) - as its a crucial guard against malicious sites.

However, this means that a small domain misconfiguration can lead to a broken/misbehaving TubePress installation.

As an example, say that you have a domain `ehough.com` and are serving a web site at this address; loading up `http://ehough.com` works normally for your users, and TubePress detects that `ehough.com` is the domain that it will use. But you *also* have the site available (without HTTP redirects) via `http://www.ehough.com`. Users that access the site via the `www.ehough.com` domain will experience a broken TubePress installation, as TubePress will attempt to make a cross-domain Ajax request to `ehough.com`.

To diagnose if this if your problem, use the following procedure:

1. Using Firefox, navigate to a page with a TubePress gallery
2. Open up Firebug
3. Open Firebug's "Console" tab
4. Click on any TubePress thumbnail. You should see a new XHR request show up as TubePress performs its Ajax request.



- Remove `fopen()` from the list of disabled functions

Troubleshooting Tools

Debugging Mode TubePress's debug mode will print a huge amount of logging statements to the screen. It will describe in (extreme) detail what TubePress is doing behind the scenes.

```
[2014-01-09 17:57:37] [DEBUG] Default Boot Config Service: option-descriptors caching is enabled
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Attempting read contents of cache file at /var/
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Attempting to hydrate from cache file at /var/w
[2014-01-09 17:57:37] [DEBUG] Default Option Descriptor Reference: Successfully hydrated from cache file at /var/w
[2014-01-09 17:57:37] [DEBUG] TubePress Bootstrapper: Boot completed in 71.796894 milliseconds
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Found a shortcode: [gallery_tubepress mode="user" userValue=
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Candidate options detected in shortcode: [gallery_tubepress mo
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: mode = "user" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: mode = "user" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: userValue = "PenWeddings" (unfiltre
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: userValue = "PenWeddings" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: youtubeHideBlackBars = "true" (unfilt
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: youtubeHideBlackBars = "1" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: resultsPerPage = "12" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: resultsPerPage = "12" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbWidth = "144" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbWidth = "144" (filtered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbHeight = "81" (unfiltered)
[2014-01-09 17:57:39] [DEBUG] Shortcode Parser: Name-value pair detected: thumbHeight = "81" (filtered)
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: mode = user
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: userValue = PenWeddings
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: youtubeHideBlackBars = 1
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: resultsPerPage = 12
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: thumbWidth = 144
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: thumbHeight = 81
[2014-01-09 17:57:39] [DEBUG] Search Output Shortcode Handler: Not configured for search results
[2014-01-09 17:57:39] [DEBUG] Memory Execution Context: Accepted valid value: galleryId = 1478746967
[2014-01-09 17:57:39] [DEBUG] Thumb Gallery Shortcode Handler: Starting to build thumbnail gallery 1478746967
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Attempting to load template instance from gallery.tpl.php with fallb
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Requested theme is 'default'
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Found gallery.tpl.php first try at /var/www/front/wordpress/wp-conte
[2014-01-09 17:57:39] [DEBUG] Theme Handler: Successfully loaded template from /var/www/front/wordpress/wp-cor
```

To use it, you just need to add `tubepress_debug=true` to the URL string (in your browser's address bar) of any page that uses TubePress on it. For instance, if the TubePress page you're trying to debug has an address of

```
http://ehough.com/?page_id=19
```

then you should add `tubepress_debug=true` to the URL to enable debugging:

```
http://ehough.com/?page_id=19&tubepress_debug=true
```

Most of the debug output is self-explanatory. If the output is cut off unexpectedly, it's likely that TubePress encountered a fatal error. Check your PHP error logs for more information.

Firebug One particular tool which is invaluable in debugging client-side problems is [Firebug](#). Firebug is an add-on for Firefox that gives you x-ray vision for any website. It's powerful, yet easy to use. If you're new to Firebug, here are some introductory resources to help you learn.

What is Firebug? From the official Firebug site

Introductory screencast Shows the general concepts and use-cases of Firebug.

Another introductory screencast A longer, more detailed video going over installation and usage.

Firebug Guide for Web Developers A detailed, well-written overview of Firebug (with pictures).

2.3 TubePress for Wix

TubePress is available to all [Wix.com](#) websites via the [Wix App Market](#). Since TubePress LLC manages the hosting of TubePress, there is no installation, upgrade, or maintenance required!

2.3.1 Manual Contents

Using TubePress in Wix.com

This page details how to use TubePress for Wix. It assumes that you already have a Wix website up and running; if not, [Wix offers excellent documentation to get you started](#).

On This Page

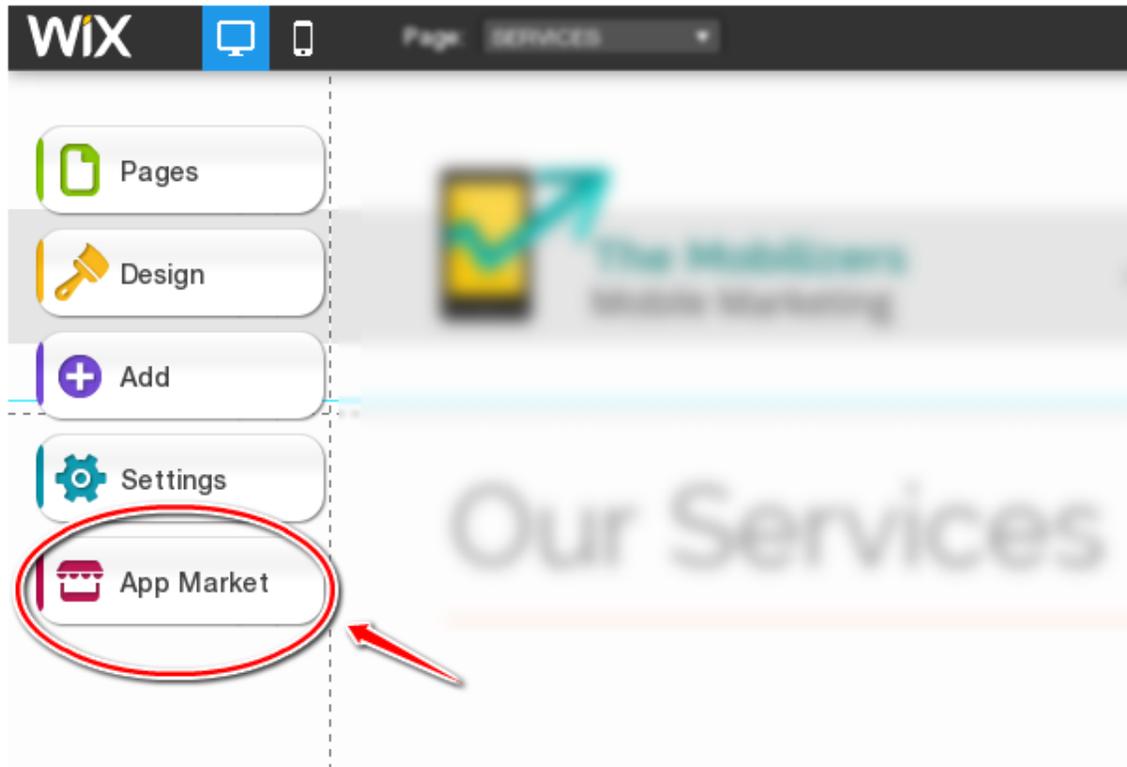
- [Adding the TubePress App](#)
- [Configuring - Simple](#)
- [Configuring - Full Settings](#)

Adding the TubePress App

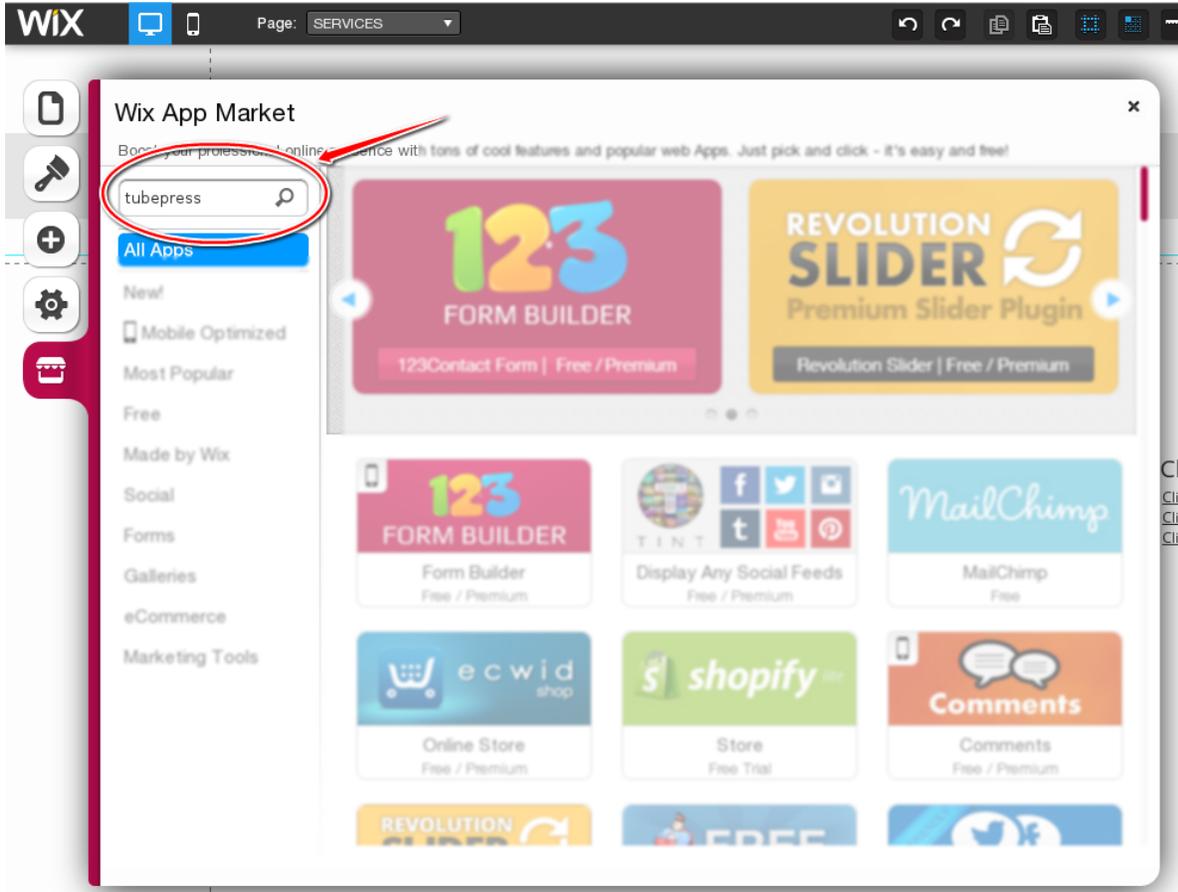
TubePress is available to [Wix.com](#) as an “app” in the [Wix App Market](#). As such, [Wix provides detailed documentation on how to use the App Market](#).

For the sake of completeness, here's how to add TubePress to your site.

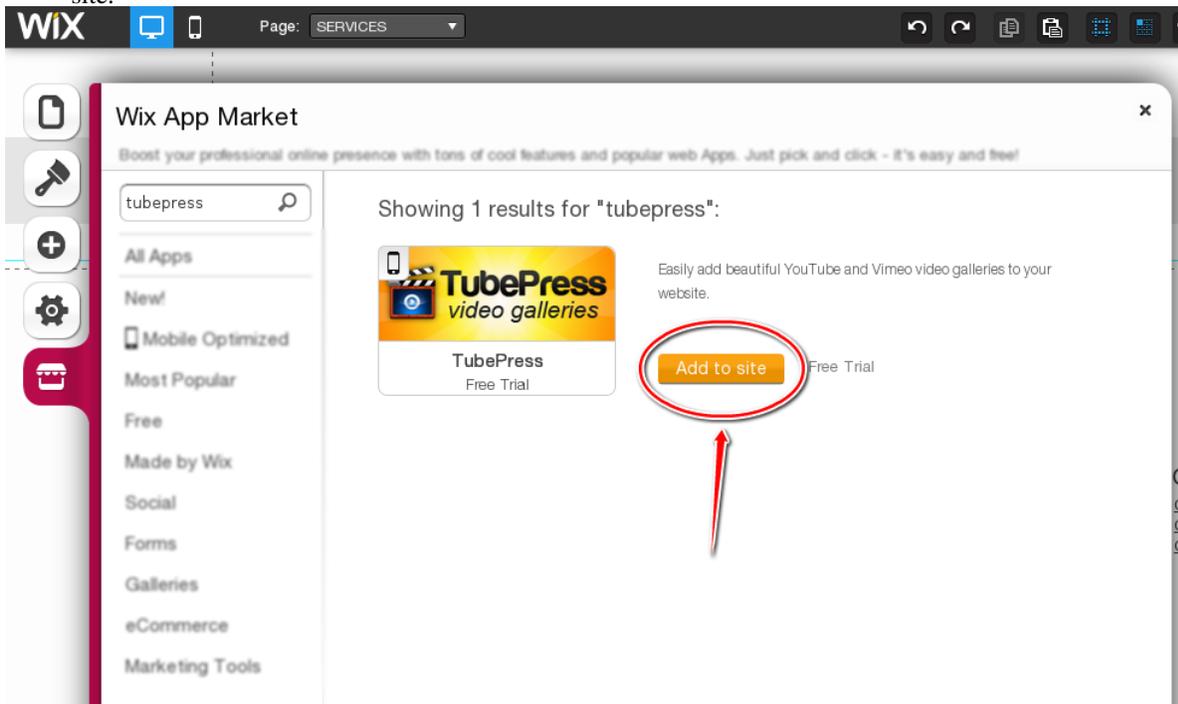
1. Inside the Wix Editor, click the App Market button.



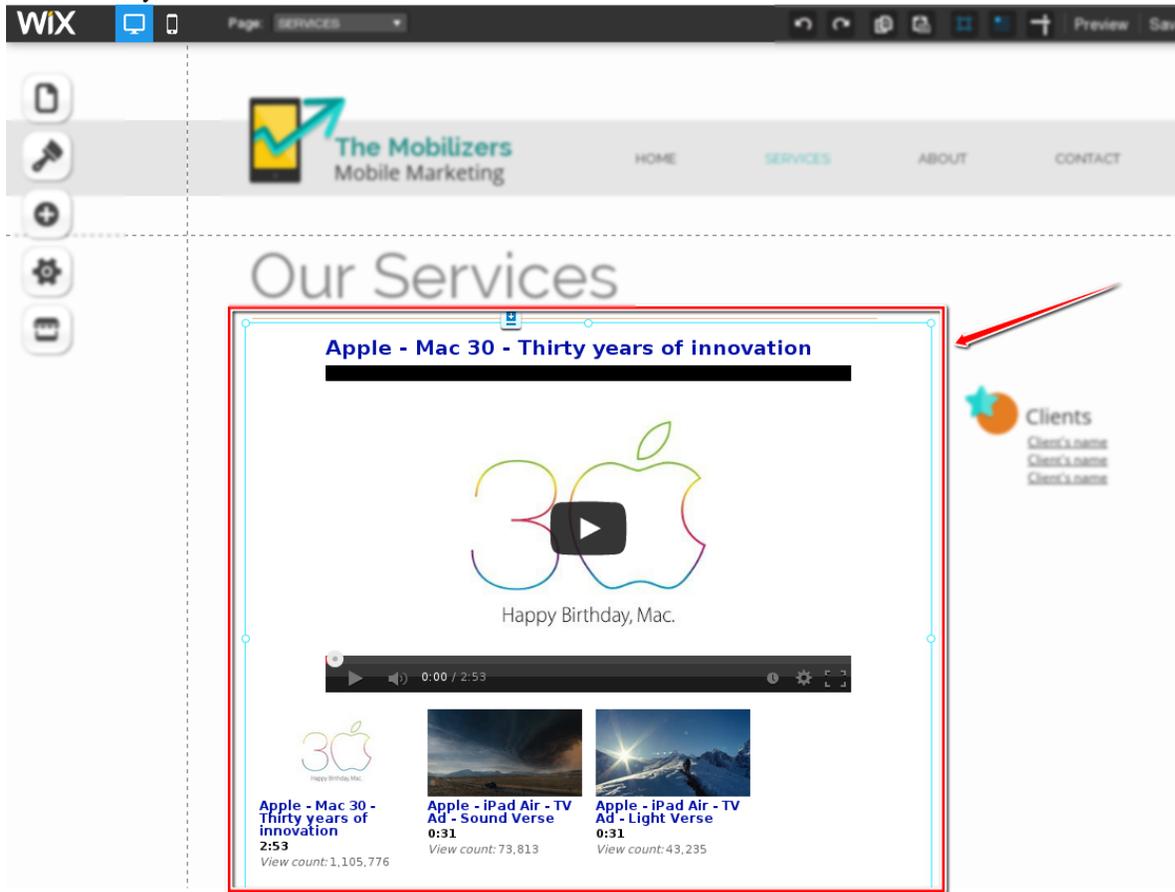
2. The Wix App Market window will open. In the search box at the top left, type in `tubepress` and hit enter.



3. The TubePress app will show in your search results. Click the Add to Site button to add TubePress to your site.



4. TubePress will be added to your site in a “box” that you may now move around and resize to your liking, just like any other Wix element.

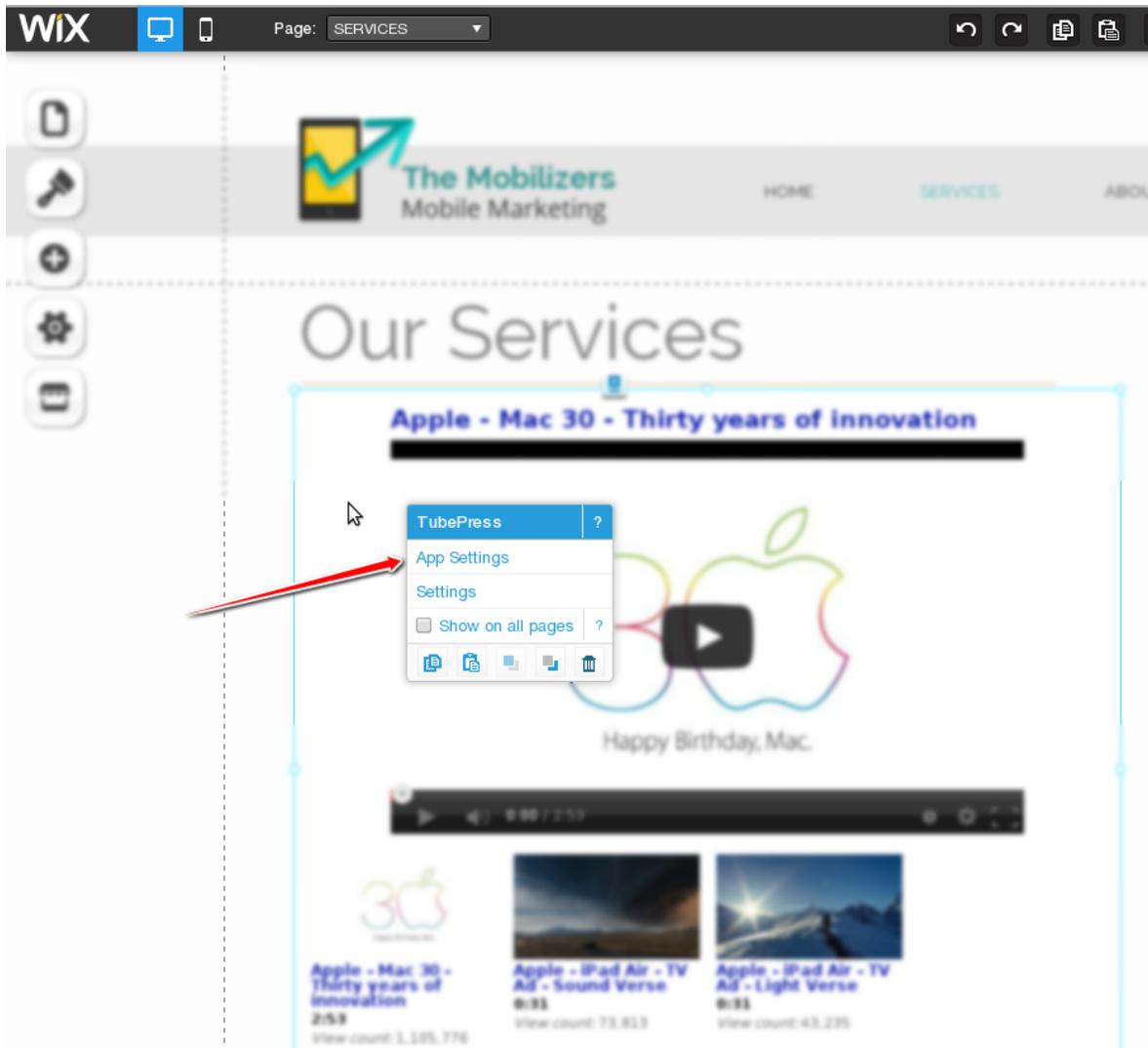


You can repeat steps 1 - 4 as many times as you like!

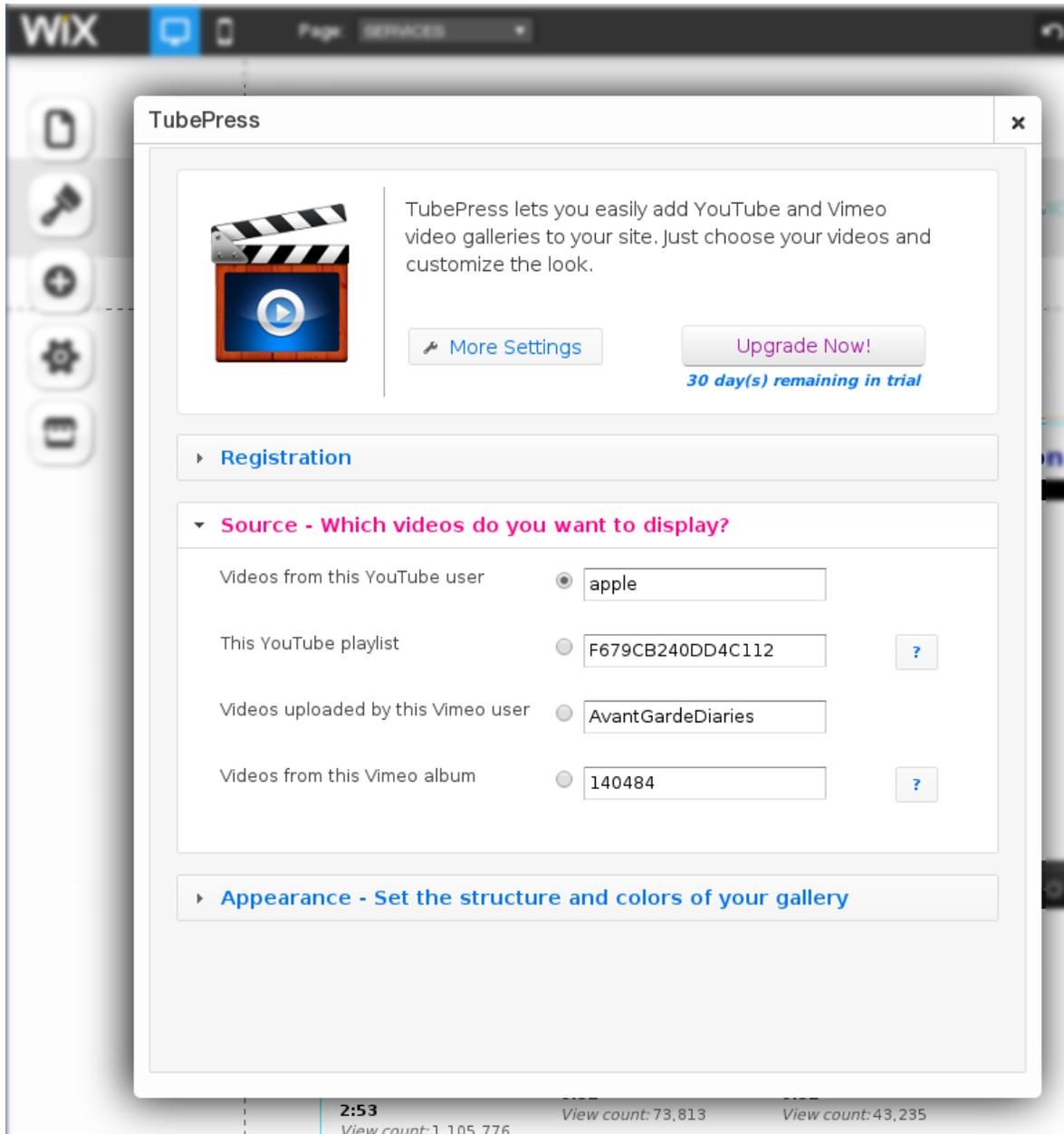
Configuring - Simple

Configuring TubePress in Wix is extremely simple.

1. Click on the TubePress instance that you'd like to configure. You'll see a popup menu. Click the link for “App Settings”.



2. A small window will popup which contains some simple settings for TubePress. Feel free to click around and change some of the options. Your changes will be reflected “live” so that you can see the result immediately.



Configuring - Full Settings

In the previous section you learned how to access a few settings for your TubePress instance. However, **the options found in the small popup window are actually a small fraction of the options available to you.** We simply “hide” them so as to not overwhelm new users.

Here’s how to access the rest of the settings:

1. In the basic TubePress settings window, click the “More Settings” button.

TubePress lets you easily add YouTube and Vimeo video galleries to your site. Just choose your videos and customize the look.

[More Settings](#)

▶ **Registration**

▼ **Source - Which videos do you want to display?**

Videos from this YouTube user

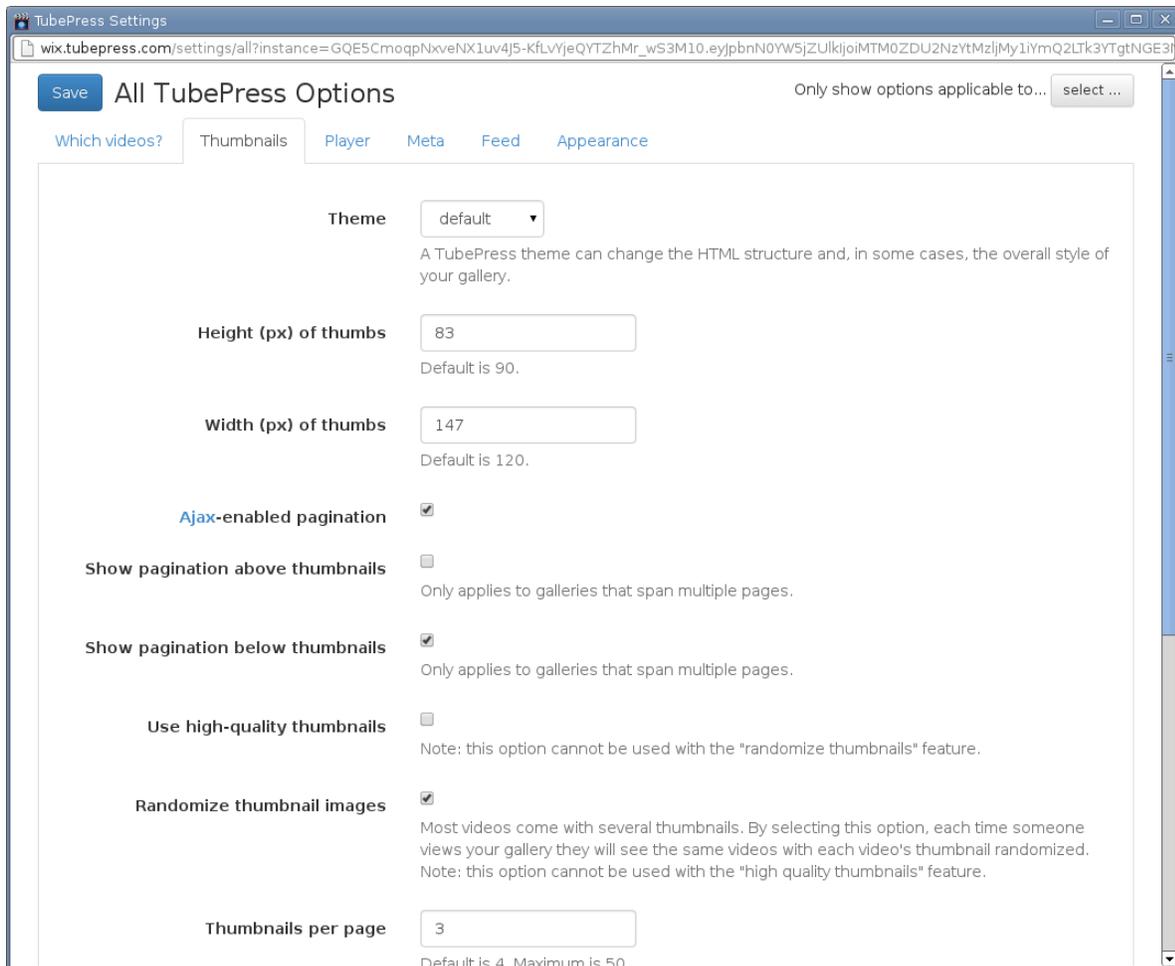
This YouTube playlist [?](#)

Videos uploaded by this Vimeo user

Videos from this Vimeo album [?](#)

▶ **Appearance - Set the structure and colors of your gallery**

2. A new window will open which contains *all* of the TubePress settings. Feel free change any setting you like, and click the “Save” button once you’re done.



Billing

TubePress is available to all Wix.com users for a free 30-day trial. To continue using the service beyond the trial period, we charge a small monthly fee. Wix provides [detailed instructions](#) on how to pay for TubePress.

Common Questions

- How many sites can I use TubePress on?
- I started a new website. How do I transfer my TubePress purchase?
- What happens if I miss a payment?

How many sites can I use TubePress on? Wix bills on a per-site basis. For instance if you have two Wix.com websites that are both using TubePress, you will be billed twice: once for each website.

However, there is no limit to the number of times that you may add TubePress to any website. Even if you use TubePress in twenty different places on your site, you will still just be billed once.

I started a new website. How do I transfer my TubePress purchase? Since Wix handles billing on our behalf, you will need to [contact Wix](#) to transfer your upgraded apps to your new site.

What happens if I miss a payment? If your trial period ends before payment, or if you miss a payment, TubePress will simply “go dark” on your website and your site visitors will see an empty box where TubePress used to be. You may continue to access the TubePress settings from within the site editor.

After Wix receives payment, TubePress will pick up right where you left off. You won’t lose anything.

FAQ & Troubleshooting

- [Common Questions](#)
 - [When I navigate to a new page, the video keeps playing in the background.](#)

Common Questions

When I navigate to a new page, the video keeps playing in the background. This is a known problem. Until Wix.com provides us with the technical capability, TubePress has no means of detecting and responding to navigation changes.

The good news is that Wix has indicated that this capability is on the very near horizon. You may read more details in [this forum thread](#).

2.4 Official TubePress Add-Ons

2.4.1 YouTube Black Bars Remover

On This Page

- [Introduction](#)
- [Manuals](#)
 - [TubePress for WordPress](#)
 - [Standalone PHP](#)

Introduction

Most YouTube video thumbnails have horizontal black bars on the top and bottom. This TubePress add-on completely removes these bars, leaving behind a clean and more professional image.

Note: Use of this add-on will change the aspect ratio of YouTube thumbnails from 4:3 to 16:9. You will likely want to adjust your settings for *thumbHeight* and *thumbWidth*.

Before

1 2 3 next »



Integrity Hills wedding by Big Cedar {Branson Wedding video}
6:19



All Souls Episcopal and Skirvin Hotel wedding {Oklahoma City wedding}
5:00



Southern Hills Country Club wedding {Tulsa wedding video}
6:17



PostOak Lodge wedding {Tulsa wedding video}
4:10



Bride learns she has brain tumor {Harwelden Mansion Wedding}
6:06



Lofty Spaces Wedding {Dallas Wedding Video}
5:24



Alden Hotel wedding {Houston wedding video}
5:14



Asbury Methodist & Mayo Hotel wedding {Tulsa wedding video}
5:04



Ramage Farms wedding {Texarkana wedding video}
2:33



Vietnamese Temple Wedding & Tea Ceremony
5:27



Dresser Mansion vintage wedding {Tulsa wedding video}
3:34



Louisiana Wedding Videography {A Southern Wedding}
5:21



Crossings Church & Skirvin Hotel wedding {Oklahoma City wedding video}
2:17



Crossings Church & OKC Farmer's Market wedding {Oklahoma City wedding video}
2:36



Katherine + Adrian | Marland Mansion Wedding Video
4:03



Gaillardia Country Club wedding {Oklahoma City wedding video}
3:30



Mildred B. Cooper Chapel Wedding Video {Arkansas Wedding}
4:32



First Presbyterian & OKC Golf & Country Club wedding {Oklahoma City wedding video}
2:11



ONEOK Field & Mayo Hotel Tulsa Wedding Video
3:10



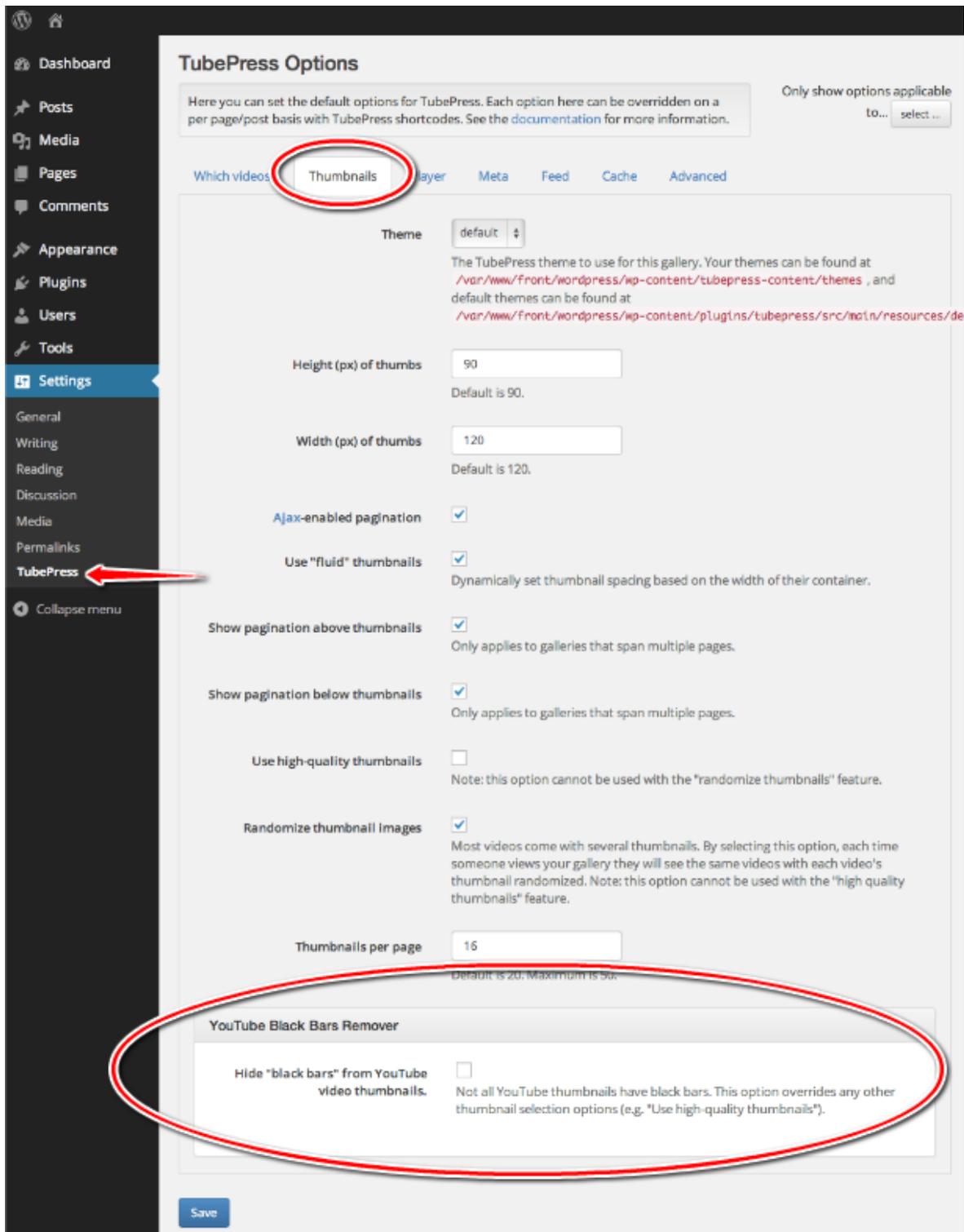
Kristin + Peter | Florida Keys Wedding Video
3:32

1 2 3 next »

Manuals

TubePress for WordPress

1. Purchase and download this add-on from [the TubePress Marketplace](#).
2. Unzip the file you downloaded (`youtube-black-bars-remover_x_y_z.zip`) into the add-ons sub-directory of your *TubePress Content Directory*.
3. Configure this add-on from WP Admin > Settings > TubePress. Navigate to the “Thumbnails” tab and scroll to the bottom of the page. There you will see a checkbox which enables/disables the black bar removal.



Standalone PHP

1. Purchase and download this add-on from the TubePress Marketplace.
2. Unzip the file you downloaded (youtube-black-bars-remover_x_y_z.zip) into the add-ons sub-

directory of your *TubePress Content Directory*.

- This add-on introduces the `youtubeHideBlackBars` option that you may supply to `TubePressPro::getHtmlForShortcode()`. e.g.

```
<?php
```

```
print TubePressPro::getHtmlForShortcode('mode="user" userValue="3hough" youtubeHideBlackBars="tr
```

2.4.2 Vimeo All Access

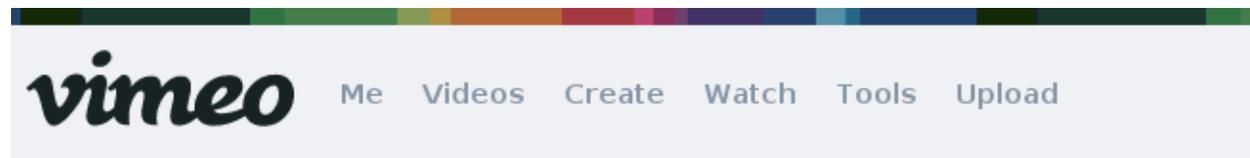
On This Page

- Introduction
- Manuals
 - TubePress for WordPress
 - TubePress for PHP
 - TubePress for Wix
- Obtaining API Credentials
 - Access Token and Secret
 - Consumer Key (Client ID) and Secret

Introduction

This TubePress add-on provides TubePress with enhanced access to your Vimeo **Plus** or **PRO** account, thus allowing TubePress to work with **private Vimeo videos**.

This is particularly useful if you'd like to hide your videos from display on `vimeo.com` and *only* show them on your TubePress-powered site. To make a video private, choose the last radio icon in the privacy settings for the video as shown below:



Video Settings / Privacy



Privacy Settings

 Video hidden on Vimeo.com

Who can watch this video?

- Anyone - Allow anyone to see this video
- Only me - Make this video visible to me and no one else
- Only people I follow - Make my videos visible only to people I follow on Vimeo
- Only people I choose - I'll select people with whom I want to share this video
- Only people with a password - Protect this video with a password
- Hide this video from Vimeo.com - I want to post this video on the web but don't want it to be available on Vimeo.com. (This will hide your video on Vimeo.com and remove it from public Albums, Channels, and Groups.)

Manuals

TubePress for WordPress

1. Purchase and download this add-on from the [TubePress Marketplace](#).
2. Unzip the file you downloaded (`vimeo-all-access_x_y_z.zip`) into the `add-ons` subdirectory of your *TubePress Content Directory*.
3. Ensure you have completed the *one-time Vimeo API setup*.
4. *Obtain your Vimeo access credentials*.
5. Supply your newly-obtained credentials (from Step 4) to TubePress at WP Admin > Settings > TubePress > Feed. On this tab you will see two new text boxes for this add-on:

The screenshot shows the WordPress dashboard with the TubePress Options settings page. The left sidebar contains navigation links for Dashboard, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, Settings (highlighted), General, Writing, Reading, Discussion, Media, Permalinks, TubePress, and Collapse menu. The main content area is titled 'TubePress Options' and includes a sub-header 'Here you can set the default options for TubePress. Each option here can be overridden on a per page/post basis with TubePress shortcodes. See the [documentation](#) for more information.' Below this are tabs for 'Which videos?', 'Thumbnails', 'Player', 'Meta', 'Feed', 'Cache', and 'Advanced'. The 'Feed' tab is active. The settings include: 'Order videos by' (relevance), 'Per-page sort order' (date published (newest first)), 'Maximum total videos to retrieve' (0), 'Video blacklist' (empty text box), and 'Restrict search results to videos from author' (empty text box). A red circle highlights the 'Vimeo All Access' option. Below this are two text input fields: 'Vimeo API "Access Token"' (524b6ee801f9cb64d3603166cc22345f) and 'Vimeo API "Access Token Secret"' (d1b268a3dc4fed973c2fbc886b85d4509a49a3df). Both fields include a link to the documentation for instructions.

TubePress for PHP

1. Purchase and download this add-on from [the TubePress Marketplace](#).
2. Unzip the file you downloaded (`vimeo-all-access_x_y_z.zip`) into the add-ons subdirectory of your *TubePress Content Directory*.
3. Ensure you have completed the *one-time Vimeo API setup*.
4. *Obtain your Vimeo access credentials.*
5. This add-on introduces two new options: `vimeoApiAccessToken` and `vimeoApiAccessTo-kenSecret`. You may supply these options, along with `vimeoKey` and `vimeoSecret`, to `TubePressPro::getHtmlForShortcode()`. e.g.

```
<?php
```

```
print TubePressPro::getHtmlForShortcode('vimeoKey="xxxx" vimeoSecret="xxxx" vimeoAccessToken
```

TubePress for Wix

Users of cloud-based TubePress installations (e.g. TubePress Express, *TubePress for Wix*) should follow these instructions.

1. Obtain your *Vimeo consumer credentials*.
2. Obtain your *Vimeo access credentials*.
3. Open up the *full settings page* for your TubePress instance.
4. In the window that opens, navigate to the “Feed” tab and scroll down to the “Vimeo All Access” box:

The screenshot shows the 'All TubePress Options' settings page. The 'Feed' tab is selected and circled in red. Below the tabs, there are several settings: 'Order videos by' (date published (newest first)), 'Per-page sort order' (none), 'Maximum total videos to retrieve' (300), 'Video blacklist', and 'Restrict search results to videos from author'. At the bottom, the 'Vimeo All Access' section is highlighted with a red box and contains four input fields: 'Vimeo API "Client ID"', 'Vimeo API "Client secret"', 'Vimeo API "Access Token"', and 'Vimeo API "Access Token Secret"'. Each field has a 'See the documentation for instructions.' link below it.

5. Enter the credentials obtained in Step 1 into the “Client ID” and “Client secret” text boxes.
6. Enter the credentials obtained in Step 2 into the “Access Token” and “Access Token Secret” boxes.
7. Click the “Save” button.

Obtaining API Credentials

Access Token and Secret

You will need to supply an additional set of Vimeo API credentials to TubePress: an “Access Token” with a corresponding “Access Token Secret”. This is a one-time step for this add-on.

Warning: It is extremely important to never publicly divulge these credentials.

1. Visit <https://developer.vimeo.com/apps>. You should see a list of any Vimeo apps that you have created. Click on the app that you are using for TubePress.
2. At the bottom of the page, you will see a box containing your Access Token and Access Token Secret:

Your access token

You can use this OAuth Access Token to access your account with this app.

Access token	dd5a625d8
Access token secret	aef801e5f77bdb
Permissions	Read, write, delete

Consumer Key (Client ID) and Secret

Note: Only cloud-based TubePress users (e.g. TubePress Express, *TubePress for Wix*, etc) need to follow these instructions, as users of downloadable distributions of TubePress should have completed this step during installation.

1. [Log in to Vimeo](#), if you haven't already.
2. [Click here](#) to create a new Vimeo app. You can fill in anything for the fields. Below is a sample form that's filled out. Then click the *Create App* button.

My Advanced API Apps /

Create a new App

App Name *

my amazing website

App Description *

When users authenticate via your app, this is what they'll see.

Use TubePress within my website.

App URL *

Your app's publicly accessible home page where users can get more information or download your app.

http://foo.com

App Logo URL

The URL of a logo to show with your app when authenticating. Displayed at 100x100 px.

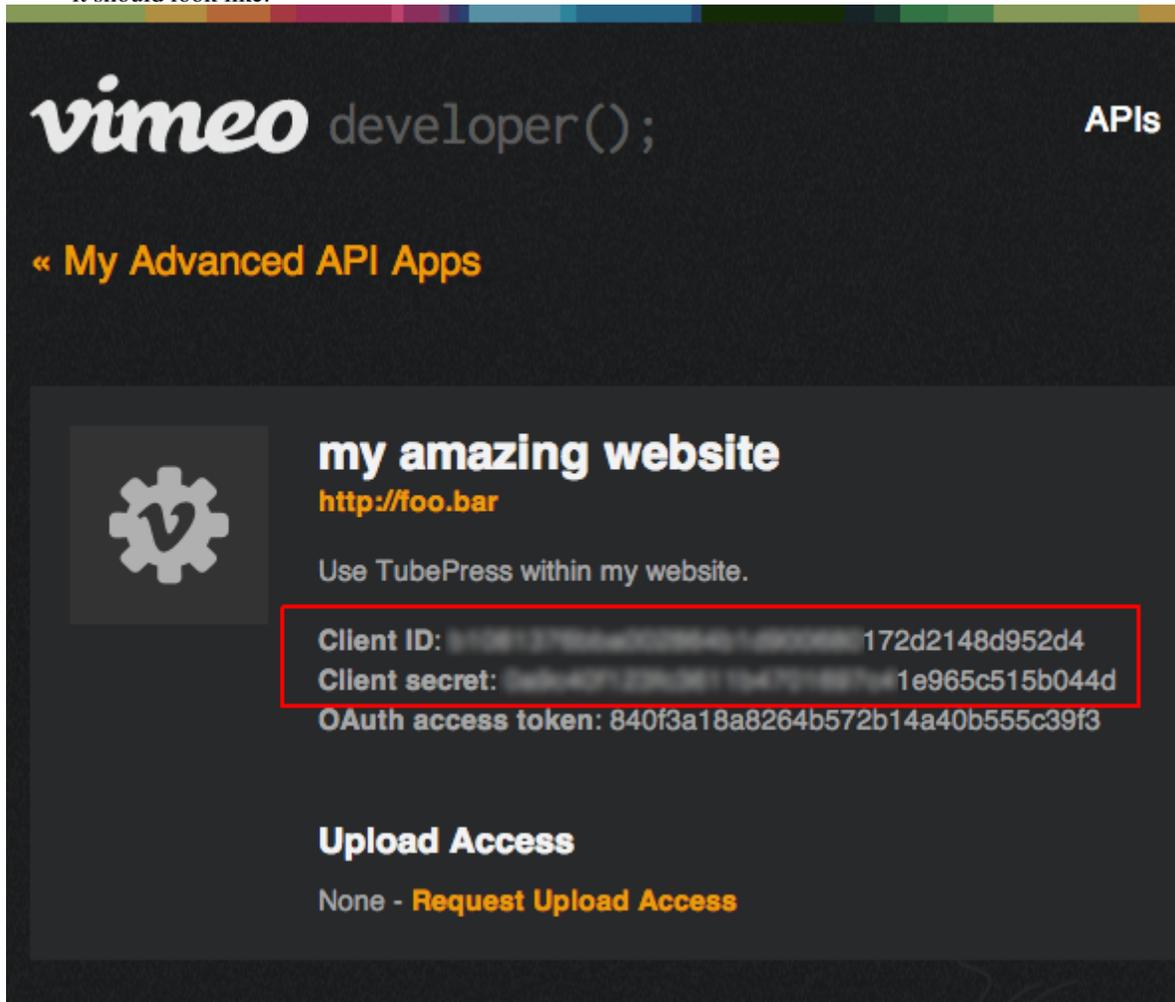
App Callback URL

This is the URL we'll send a user back to when authenticating.

I agree that my application does not violate the [Vimeo API License Agreement](#) or the [Vimeo Terms of Service](#)

* Required

3. Make a note of the “Client ID” and “Client secret” of your new app. See the red box below for an idea of what it should look like.



2.4.3 Flexible Thumbnail Rows

On This Page

- Introduction
- Manuals
 - TubePress for WordPress
 - Standalone PHP
 - TubePress for Wix

Introduction

This add-on automatically adjusts the height of thumbnail rows in TubePress galleries, ensuring that the metadata (runtime, title, description, etc) for each video is completely visible. Without this add-on, TubePress uses a fixed height for the thumbnail rows, and any metadata that is taller than the row will be cut off.

Before



Brides of Oklahoma gown shoot with Bertille Sefolsha

3:42
ID: 96HdK-RzobE
View count: 150
Date posted: Jan 9, 2014



Pear Tree Estate wedding

4:48
ID: TglhaK9HWSY
View count: 1,183
Date posted: Jan 1, 2014
Here's a list of some of the



Hotel Zaza, St. Anne's, The Corinthian wedding

6:24
ID: mV4xV8Lx7r8
View count: 1,910
Date posted: Nov 21, 2013



Perini Ranch wedding film

4:37
ID: UtwYmr-zE5s
View count: 2,234
Date posted: Nov 6, 2013
Rebecca and Travis are so



Mayo Hotel wedding

4:38
ID: qsTfITxudY
View count: 1,733
Date posted: Oct 30, 2013
I met with Bronwyn and John at the Coffee House on



Coles Garden 8mm wedding film

4:55
ID: 1XYxEHoGuDc
View count: 813
Date posted: Oct 21, 2013
Dana and Zach didn't



Norman wedding video {Oklahoma wedding videographers}

5:38
ID: YhH55iB50kM
View count: 882



Skelly Lodge wedding

4:38
ID: QhidsDdV55o
View count: 1,482
Date posted: Oct 9, 2013
We instantly loved Emily and Michael's plans for a



Mayo Hotel, First United Methodist wedding

6:06
ID: CYXkw0jQAPk
View count: 1,398
Date posted: Oct 4, 2013



Coles Garden wedding

4:43
ID: _PLN4oWsdnM
View count: 1,789
Date posted: Sep 27, 2013
Kylee and Jay (or JJ) as we quickly learned he is called



Skirvin Hotel wedding {Oklahoma wedding videographers}

4:39
ID: B2JrN2doB34
View count: 1,552
Date posted: Sep 20, 2013



Dominion House wedding

4:59
ID: gE-R6dRUWdk
View count: 966
Date posted: Sep 13, 2013
The Dominion House in



Engagement shoot, Love Story film at Boone Pickens Stadium

7:59
ID: wQczcmCfLn8
View count: 1 042



Five Oaks Lodge wedding

4:36
ID: C83E-w9g-hs
View count: 2,500
Date posted: Aug 4, 2013
There was an overall



Philbrook Museum of Art wedding

4:20
ID: 8FneuFnFljw
View count: 2,308
Date posted: Jul 29, 2013
Jeff and Iulia had a simple



Integrity Hills wedding by Big Cedar {Branson Wedding video}

6:19
ID: oF0fqZBYNk8
View count: 2 093

After



Manuals

TubePress for WordPress

1. Purchase and download this add-on from [the TubePress Marketplace](#).
2. Unzip the file you downloaded (`flexible-thumbnail-rows_x_y_z.zip`) into the `add-ons` subdirectory of your *TubePress Content Directory*.
3. Configure this add-on from WP Admin > Settings > TubePress. Navigate to the “Thumbnails” tab and scroll to the bottom of the page. There you will see a checkbox which enables/disables the add-on.

The screenshot shows the WordPress dashboard with the TubePress Options page open. The left sidebar contains a menu with 'Settings' expanded and 'TubePress' selected, indicated by a red arrow. The main content area is titled 'TubePress Options' and has several tabs: 'Which videos?', 'Thumbnails', 'Player', 'Meta', 'Feed', 'Cache', and 'Advanced'. The 'Thumbnails' tab is circled in red. Below the tabs, there are various settings for the gallery, including 'Theme' (set to 'default'), 'Height (px) of thumbs' (90), 'Width (px) of thumbs' (120), 'Ajax-enabled pagination' (unchecked), 'Use "fluid" thumbnails' (checked), 'Show pagination above thumbnails' (checked), 'Show pagination below thumbnails' (checked), 'Use high-quality thumbnails' (checked), 'Randomize thumbnail images' (checked), and 'Thumbnails per page' (20). The 'Flexible Thumbnail Rows' section is circled in red and contains the option 'Flexible thumbnail rows' which is checked. A 'Save' button is located at the bottom left of the options area. The text 'szoter.com' is visible in the bottom right corner of the screenshot.

Standalone PHP

1. Purchase and download this add-on from the TubePress Marketplace.

2. Unzip the file you downloaded (`flexible-thumbnail-rows_x_y_z.zip`) into the `add-ons` subdirectory of your *TubePress Content Directory*.
3. This add-on introduces the `flexibleThumbnailRows` option that you may supply to `TubePressPro::getHtmlForShortcode()`. e.g.

```
<?php
```

```
print TubePressPro::getHtmlForShortcode('mode="user" userValue="3hough" flexibleThumbnailRows="t
```

TubePress for Wix

This add-on is pre-installed for users of cloud-based TubePress installations (e.g. TubePress Express, *TubePress for Wix*). Here's how to use it.

1. Open up the *full settings page* for your TubePress instance.
2. In the window that opens, navigate to the “Thumbnails” tab and scroll down to the “Flexible Thumbnail Rows” box:

Save All TubePress Options Only show options applicable to... select ...

Which videos: Thumbnails Player Meta Feed Appearance

Theme default
A TubePress theme can change the HTML structure and, in some cases, the overall style of your gallery.

Height (px) of thumbs 83
Default is 90.

Width (px) of thumbs 147
Default is 120.

Ajax-enabled pagination

Show pagination above thumbnails
Only applies to galleries that span multiple pages.

Show pagination below thumbnails
Only applies to galleries that span multiple pages.

Use high-quality thumbnails
Note: this option cannot be used with the "randomize thumbnails" feature.

Randomize thumbnail images
Most videos come with several thumbnails. By selecting this option, each time someone views your gallery they will see the same videos with each video's thumbnail randomized. Note: this option cannot be used with the "high quality thumbnails" feature.

Thumbnails per page 3
Default is 4. Maximum is 50.

YouTube Black Bars Remover

Hide "black bars" from YouTube video thumbnails.
Not all YouTube thumbnails have black bars. This option overrides any other thumbnail selection options (e.g. "Use high-quality thumbnails").

Flexible Thumbnail Rows

Flexible thumbnail rows
Equalize the height of thumbnail rows so that no video metadata is cut off.

Save

szoter.com

3. Enable or disable the add-on with the checkbox.
4. Click the "Save" button.

2.4.4 QuickPlay

On This Page

- Introduction
- Manuals
 - TubePress for WordPress
 - TubePress for PHP
- Additional Notes
 - Controlling Height and Width
 - Show the Latest Video from a Collection

Introduction

This TubePress add-on replaces video thumbnail images with fully-functional embedded video players. *Among other things*, this allows your users to watch your videos right within the context of your gallery.

Before



Brides of Oklahoma gown shoot with Bertille Sefolosha

3:42

For their 2014 Spring/Summer issue, the Brides of Oklahoma chose a dramatic blac...



Pear Tree Estate wedding

4:48

Here's a list of some of the awesome things about Kelly & Aaron's wedding day. 1...



Hotel Zaza, St. Anne's, The Corinthian wedding

6:24

This group is all kinds of crazy, in a totally lovable way. And let me say how e...



Perini Ranch wedding film

4:37

Rebecca and Travis are so sweet, so calm, and so easy going. They planned a beau...



Mayo Hotel wedding

4:38

I met with Bronwyn and John at the Coffee House on Cherry Street before their we...



Coles Garden 8mm wedding film

4:55

Dana and Zach didn't appear to have a care in the world on their wedding day. Al...



Norman wedding video (Oklahoma wedding videographers)

5:38

There were so many fun, special, and unique things about Whitney and Tom's Norma...



Skelly Lodge wedding

4:38

We instantly loved Emily and Michael's plans for a wedding in the woods and dinn...

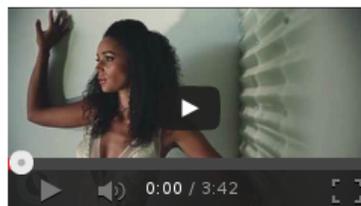


Mayo Hotel, First United Methodist wedding

6:06

Despite some failed attempts to meet before their wedding, thanks to Facebook an...

After



Brides of Oklahoma gown shoot with Bertille Sefolosha

3:42

For their 2014 Spring/Summer issue, the Brides of Oklahoma chose a dramatic blac...



Pear Tree Estate wedding

4:48

Here's a list of some of the awesome things about Kelly & Aaron's wedding day. 1...



Hotel Zaza, St. Anne's, The Corinthian wedding

6:24

This group is all kinds of crazy, in a totally lovable way. And let me say how e...



Note: This add-on is currently only available for the downloadable TubePress distributions

Manuals

TubePress for WordPress

1. Purchase and download this add-on from the [TubePress Marketplace](#).
2. Unzip the file you downloaded (`quickplay_x_y_z.zip`) into the add-ons subdirectory of your *TubePress Content Directory*.
3. Copy (or symlink) the directory at `wp-content/tubepress-content/add-ons/quickplay_x_y_z/tubepress-t` to `wp-content/tubepress-content/themes/quickplay`
4. Anytime you *add TubePress to a post or page*, set the options `enableQuickplay='true'` and `theme='quickplay'`. e.g.

```
[tubepress enableQuickplay="true" theme="quickplay" ... ]
```

TubePress for PHP

1. Purchase and download this add-on from the [TubePress Marketplace](#).
2. Unzip the file you downloaded (`quickplay_x_y_z.zip`) into the add-ons subdirectory of your *TubePress Content Directory*.
3. Copy (or symlink) the directory at `tubepress-content/add-ons/quickplay_x_y_z/tubepress-theme` to `tubepress-content/themes/quickplay`
4. Anytime you *add TubePress to a page*, set the options `enableQuickplay='true'` and `theme='quickplay'`. e.g.

```
<?php
```

```
print TubePressPro::getHtmlForShortcode('enableQuickplay="true" theme="quickplay" ... ');
```

Additional Notes

Controlling Height and Width

To control the height and width of the embedded video players, you can simply use the *embeddedHeight* and *embeddedWidth* options. *thumbHeight* and *thumbWidth* are **ignored** when this add-on is in use.

Show the Latest Video from a Collection

A common question we get from customers is

How can I have an embedded video player that automatically shows the latest video from my YouTube or Vimeo channel?

With QuickPlay, this is easy. The trick is to set *resultCountCap* to 1 and *orderBy* to newest.

For instance, say that you'd like to show the latest video from YouTube user *engineerguyvideo*. WordPress users would use the following shortcode:

```
[tubepress mode='user' userValue='engineerguyvideo' resultCountCap='1' orderBy='newest' enableQuickp
```

Similarly, PHP users could use the following invocation of `TubePressPro::getHtmlForShortcode()`:

```
<?php
```

```
TubePressPro::getHtmlForShortcode("mode='user' userValue='engineerguyvideo' resultCountCap='1' or
```

The result is an always-updating, fully customizable single video embed that you can embed anywhere on your site.



How a lead-acid battery works

4:56

View count: 153,850

Reference

3.1 Options

3.1.1 Core Options

On This Page

- **General**
 - mode
 - output
 - video
- **Thumbnails**
 - ajaxPagination
 - fluidThumbs
 - hqThumbs
 - paginationAbove
 - paginationBelow
 - randomize_thumbnails
 - resultsPerPage
 - thumbHeight
 - thumbWidth
- **Embedded Video Player**
 - autoNext
 - autoplay
 - embeddedHeight
 - embeddedWidth
 - enableJsApi
 - lazyPlay
 - loop
 - playerImplementation
 - playerLocation
 - showInfo
- **Video Feed**
 - orderBy
 - perPageSort
 - resultCountCap
 - searchResultsRestrictedToUser
 - videoBlacklist
- **Video Meta Display**
 - author
 - category
 - dateFormat
 - description
 - descriptionLimit
 - id
 - length
 - relativeDates
 - tags
 - title
 - uploaded
 - url
 - views
- **Interactive Search**
 - searchProvider
 - searchResultsDomId
 - searchResultsOnly
 - searchResultsUrl
- **API Cache**
 - cacheCleaningFactor
 - cacheDirectory
 - cacheEnabled
 - cacheLifetimeSeconds

- **Advanced**
 - debugging_enabled
 - galleryId
 - httpMethod

General

General Options

- `mode`
- `output`
- `video`

mode

Option name	mode
Description	Defines the source for a video gallery.
Provided with	Any TubePress distribution
Default value	<i>youtubeMostPopular</i>
Valid values	<ul style="list-style-type: none"> • <i>Any YouTube source</i> • <i>Any Vimeo source</i>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress mode="youtubeRelated"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="youtubeRe</code>

output

Option name	output
Description	Defines what HTML TubePress will generate. Unless explicitly set, TubePress will generate a video gallery.
Provided with	Any TubePress distribution
Default value	<i>empty</i>
Valid values	<p>searchInput HTML input for interactive searching.</p> <p>searchOutput Search results from interactive searching.</p> <p>player HTML for a single embedded video (player only).</p> <p>ajaxSearchInput HTML input for asynchronous interactive searching. TubePress Pro only.</p> <p><i>empty</i> HTML for a video gallery.</p>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress output="searchInput"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('output="searchI</code>

video

Option name	video
Description	Causes TubePress to generate HTML for a single video embed.
Provided with	Any TubePress distribution
Default value	<i>empty</i>
Valid values	Any video ID (YouTube, Vimeo, etc) that TubePress knows how to handle.
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress video="lJtHNEDnrnY"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' video="lJtHNEDnrnY"');</code>

Thumbnails

<p>Thumbnail Options</p> <ul style="list-style-type: none"> • <code>ajaxPagination</code> • <code>fluidThumbs</code> • <code>hqThumbs</code> • <code>paginationAbove</code> • <code>paginationBelow</code> • <code>randomize_thumbnails</code> • <code>resultsPerPage</code> • <code>thumbHeight</code> • <code>thumbWidth</code>

ajaxPagination

Option name	ajaxPagination
Description	Use Ajax to paginate (clicking on “Next”, etc) through galleries.
Provided with	<ul style="list-style-type: none"> • TubePress Pro • Any cloud-hosted TubePress
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress ajaxPagination="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' ajaxPagination=</code>

fluidThumbs

Option name	fluidThumbs
Description	Dynamically adjust the width of thumbnail columns to best fit their container. This generally improves the appearance of the thumbnail gallery, but can cause problems in complex site layouts.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress fluidThumbs="false"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' fluidThumbs="false"');

hqThumbs

Option name	hqThumbs
Description	Use higher-quality thumbnails. This allows thumbnail sizes greater than 120px x 90x without causing distortion.
Provided with	<ul style="list-style-type: none"> • TubePress Pro • Any cloud-hosted TubePress
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress hqThumbs="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' hqThumbs="true"');

paginationAbove

Option name	paginationAbove
Description	Show pagination links (“Next”, “Prev”, etc) above the thumbnail gallery.
Provided with	Any TubePress distribution
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress paginationAbove="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' paginationAbove="true"');

`paginationBelow`

Option name	<code>paginationBelow</code>
Description	Show pagination links (“Next”, “Prev”, etc) below the thumbnail gallery.
Provided with	Any TubePress distribution
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress paginationBelow="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('paginationBelow="true"');</code>

`randomize_thumbnails`

Option name	<code>randomize_thumbnails</code>
Description	Most videos come with several thumbnails. By setting this option to true, each time a user visits a gallery they will see a randomly-selected thumbnail for each video. This option conflicts with <code>hqThumbs</code> .
Provided with	Any TubePress distribution
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress randomize_thumbnails="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('randomize_thumbnails="true"');</code>

`resultsPerPage`

Option name	<code>resultsPerPage</code>
Description	How many thumbnails to display on each page of a gallery.
Provided with	Any TubePress distribution
Default value	<code>20</code>
Valid values	Any integer from 1 to 50
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress resultsPerPage="30"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('resultsPerPage="30"');</code>

`thumbHeight`

Option name	<code>thumbHeight</code>
Description	The desired height (in pixels) of video thumbnails.
Provided with	Any TubePress distribution
Default value	<code>90</code>
Valid values	Any positive integer.
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress thumbHeight="60"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('thumbHeight="60"');</code>

`thumbWidth`

Option name	<code>thumbWidth</code>
Description	The desired width (in pixels) of video thumbnails.
Provided with	Any TubePress distribution
Default value	120
Valid values	Any positive integer.
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress thumbWidth="150"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('thumbWidth="150"');</code>

Embedded Video Player

<p>Embedded Video Player Options</p> <ul style="list-style-type: none"> • <code>autoNext</code> • <code>autoplay</code> • <code>embeddedHeight</code> • <code>embeddedWidth</code> • <code>enableJsApi</code> • <code>lazyPlay</code> • <code>loop</code> • <code>playerImplementation</code> • <code>playerLocation</code> • <code>showInfo</code>
--

`autoNext`

Option name	<code>autoNext</code>
Description	Automatically start the next video in a gallery when playback of a video finishes.
Provided with	<ul style="list-style-type: none"> • TubePress Pro • Any cloud-hosted TubePress
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress autoNext="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('autoNext="true"');</code>

`autoplay`

Option name	<code>autoplay</code>
Description	Automatically start video playback of <i>any</i> embedded video when the page is loaded.
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress autoplay="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('autoplay="true"');</code>

`embeddedHeight`

Option name	<code>embeddedHeight</code>
Description	The height, in pixels, of the embedded video player that TubePress builds.
Provided with	All TubePress distributions
Default value	<code>350</code>
Valid values	Any positive integer
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress embeddedHeight="350"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('embeddedHeight="350"');</code>

`embeddedWidth`

Option name	<code>embeddedWidth</code>
Description	The width, in pixels, of the embedded video player that TubePress builds.
Provided with	All TubePress distributions
Default value	<code>425</code>
Valid values	Any positive integer
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress embeddedWidth="350"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('embeddedWidth="350"');</code>

`enableJsApi`

Option name	<code>enableJsApi</code>
Description	Enable or disable the TubePress JavaScript API for this gallery. Enabling this API incurs a tiny performance overhead, but is required for some features (such as <i>autoNext</i>).
Provided with	TubePress Pro
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress enableJsApi="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('enableJsApi="true"');</code>

lazyPlay

Option name	lazyPlay
Description	If enabled, video playback will auto-start after users clicks a video's thumbnail.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress lazyPlay="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' lazyPlay="true"');

loop

Option name	loop
Description	If enabled, immediately restart playback of each video after it finishes.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress loop="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' loop="true"');

playerImplementation

Option name	playerImplementation
Description	Defines the "brand" of the embedded video player.
Provided with	All TubePress distributions except TubePress for Wix
Default value	provider_based
Valid values	<p>provider_based Uses the provider's player (i.e. the standard YouTube player)</p> <p>embedplus Plays videos with EmbedPlus</p> <p>longtail Plays videos with JW Player</p>
Supported provider(s)	YouTube
Shortcode example	[tubepress playerImplementation="longtail"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('playerImplement

`playerLocation`

<p>Option name Description</p> <p>Provided with Default value Valid values</p>	<p><code>playerLocation</code> Defines the “location” of the embedded video player. This allows you to choose the location and effect of how the embedded videos play. All TubePress distributions, though availability varies <code>normal</code></p> <p>normal Embedded player is placed above thumbnail gallery popup Videos play in an HTML popup window youtube User is taken to the video’s home on youtube.com for viewing. vimeo User is taken to the video’s home on vimeo.com for viewing. shadowbox Video plays in a modal window with Shadowbox.js jqmodal Video plays in a modal window with jq-Modal solo Page refreshes, and video player replaces the thumbnail gallery static Like <code>normal</code>, but each thumbnail click triggers a page refresh tinybox Video plays in a modal window with Tiny-Box. Not available in free WordPress plugin. fancybox Video plays in a modal window with Fancybox. Not available in free WordPress plugin.</p>
<p>Supported provider(s) Shortcode example Standalone PHP example</p>	<p>Vimeo and YouTube <code>[tubepress playerLocation="popup"]</code> <code>TubePressPro::getHtmlForShortcode('playerLocation=</code></p>

`showInfo`

<p>Option name Description</p> <p>Provided with Default value Valid values Supported provider(s) Shortcode example Standalone PHP example</p>	<p><code>showInfo</code> Show or hide the video’s title, description, and other meta information on the embedded video itself before playback begins. All TubePress distributions <code>false</code> <code>true</code> or <code>false</code> Vimeo and YouTube <code>[tubepress showInfo="true"]</code> <code>TubePressPro::getHtmlForShortcode('showInfo="true');</code></p>
--	---

Video Feed

perPageSort

Option name	perPageSort
Description	Defines an additional sorting to apply to each individual page of a video gallery.
Provided with	All TubePress distributions
Default value	none
Valid values	<p>commentCount Videos with more comments will be shown before others.</p> <p>duration Longest-running videos shown first.</p> <p>newest Newer videos first.</p> <p>none Do nothing.</p> <p>oldest Older videos first.</p> <p>random Shuffles the videos.</p> <p>rating Highest-rated videos first.</p> <p>title Videos will be shown in alphabetical order of their titles.</p> <p>viewCount Most-viewed videos first.</p>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress perPageSort="title"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode ('perPageSort="ti</code>

resultCountCap

Option name	resultCountCap
Description	Set a maximum limit on the total number of videos in a gallery. This can both limit the number of videos that show up on a page (if <code>resultsPerPage > resultCountCap</code>), or reduce the number of pagination links for a gallery. Set to 0 to disable any limit.
Provided with	All TubePress distributions
Default value	0
Valid values	Any non-negative integer
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress resultCountCap="100"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode ('resultCountCap="100"');</code>

searchResultsRestrictedToUser

Option name	searchResultsRestrictedToUser
Description	For keyword-based galleries, or during interactive searching, this option can filter the results to videos uploaded by the given user.
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	Any YouTube or Vimeo username
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress searchResultsRestrictedToUser="3hough"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('searchResultsRestrictedToUser="3hough"')</code>

videoBlacklist

Option name	videoBlacklist
Description	A list of video IDs that should never appear in TubePress's output.
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	A comma-separated list of YouTube or Vimeo IDs
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress videoBlacklist="HSrtIrVCm64, BnS-rTbFw2g, 3045633"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('videoBlacklist="HSrtIrVCm64, BnS-rTbFw2g, 3045633"');</code>

Video Meta Display

<p>Video Meta Display Options</p> <ul style="list-style-type: none"> • <code>author</code> • <code>category</code> • <code>dateFormat</code> • <code>description</code> • <code>descriptionLimit</code> • <code>id</code> • <code>length</code> • <code>relativeDates</code> • <code>tags</code> • <code>title</code> • <code>uploaded</code> • <code>url</code> • <code>views</code>

author

Option name	author
Description	Toggle display of the video uploader's username.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress author="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('author="true"');</code>

category

Option name	category
Description	Toggle display of the video category.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	YouTube
Shortcode example	<code>[tubepress category="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('category="true"');</code>

dateFormat

Option name	dateFormat
Description	Set the textual formatting of date information for videos. See <code>date()</code> for examples.
Provided with	All TubePress distributions
Default value	<code>M j, Y</code>
Valid values	Any valid format for PHP's <code>date()</code> function
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress dateFormat="l jS \of F Y h:i:s A"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('dateFormat="l jS \of F Y h:i:s A"');</code>

description

Option name	description
Description	Toggle display of the video description.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress description="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('description="true"');</code>

`descriptionLimit`

Option name	<code>descriptionLimit</code>
Description	The maximum number of characters of a video’s description that should be displayed. Descriptions over this limit will be truncated with . . . Set to 0 for no limit.
Provided with	All TubePress distributions
Default value	0
Valid values	Any non-negative integer
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress descriptionLimit="150"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('descriptionLimit="150"');</code>

`id`

Option name	<code>id</code>
Description	Toggle display of the video ID.
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress id="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('id="true"');</code>

`length`

Option name	<code>length</code>
Description	Toggle display of the video runtime.
Provided with	All TubePress distributions
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo
Shortcode example	<code>[tubepress length="false"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('length="false"');</code>

`relativeDates`

Option name	<code>relativeDates</code>
Description	Toggle display of relative dates, such as “last year” instead of “Nov 3, 1980”
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo or YouTube
Shortcode example	<code>[tubepress relativeDates="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('relativeDates="true"');</code>

tags

Option name	tags
Description	Toggle display of the video keywords.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo
Shortcode example	[tubepress tags="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('tags="true"');

title

Option name	title
Description	Toggle display of the video title.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress title="false"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('title="false"');

uploaded

Option name	uploaded
Description	Toggle display of the video upload date.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress uploaded="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('uploaded="true"');

url

Option name	url
Description	Toggle display of the video URL.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress url="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('url="true"');

views

Option name	views
Description	Toggle display of the video view count.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress views="false"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' views="false"');

Interactive Search

Interactive Search Options
<ul style="list-style-type: none"> • searchProvider • searchResultsDomId • searchResultsOnly • searchResultsUrl

searchProvider

Option name	searchProvider
Description	The name of a video provider (e.g. YouTube or Vimeo) which should be searched for matching videos.
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	The name of a search provider. Current either youtube or vimeo.
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress searchProvider="vimeo"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' searchProvider="vimeo"');

searchResultsDomId

Option name	searchResultsDomId
Description	Used with the detached <code>playerLocation</code> option, this option defines a jQuery selector for which TubePress should place the search results.
Provided with	TubePress Pro
Default value	<i>empty</i>
Valid values	Any valid jQuery selector
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress searchResultsDomId="#tubepress-search-results-div"]
Standalone PHP example	TubePressPro::getHtmlForShortcode(' searchResultsDomId="#tubepress-search-res

`searchResultsOnly`

Option name	<code>searchResultsOnly</code>
Description	If set to true, this shortcode will produce search results only after the user has submitted search terms. It will be “invisible” if the user is not searching.
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress searchResultsOnly="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('searchResultsOnly="true"');</code>

`searchResultsUrl`

Option name	<code>searchResultsUrl</code>
Description	A URL defining where TubePress should send the user’s search terms. By default, this is <code>\$_SERVER[PHP_SELF]</code> .
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	Any absolute URL
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress searchResultsUrl="http://mysite.com/search.php"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('searchResultsUrl="http://mysite.com/search.php"');</code>

API Cache

API Cache Options
<ul style="list-style-type: none"> • <code>cacheCleaningFactor</code> • <code>cacheDirectory</code> • <code>cacheEnabled</code> • <code>cacheLifetimeSeconds</code>

cacheCleaningFactor

Option name	cacheCleaningFactor
Description	Defines how often TubePress will perform a full clean of its API cache. If you enter <i>x</i> , the API cache will be cleaned approximately every 1/ <i>x</i> cache writes. Enter 0 to disable all cache cleaning.
Provided with	All downloadable TubePress distributions
Default value	20
Valid values	Any non-negative integer
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress cacheCleaningFactor="0"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' cacheCleaningFactor="0"');</code>

cacheDirectory

Option name	cacheDirectory
Description	The absolute path of a directory in which the TubePress API cache can store its contents. This directory must be writable by the web server's PHP process owner. If this option is left empty, TubePress will attempt to find and use the system temporary directory.
Provided with	All downloadable TubePress distributions
Default value	<i>empty</i>
Valid values	An absolute path of a writeable (by the PHP process owner) directory on the web server, or <i>empty</i> .
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress cacheDirectory="/tmp/tubepress-cache"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' cacheDirectory="/tmp/tubepress-cache"');</code>

cacheEnabled

Option name	cacheEnabled
Description	Enables or disables the TubePress API cache. This can significantly improve the performance of TubePress at the slight expense of freshness.
Provided with	All downloadable TubePress distributions
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress cacheEnabled="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' cacheEnabled="true"');</code>

`cacheLifetimeSeconds`

Option name	<code>cacheLifetimeSeconds</code>
Description	How long, in seconds, before an item in the TubePress API cache is considered to be stale.
Provided with	All downloadable TubePress distributions
Default value	3600
Valid values	Any non-negative integer.
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress cacheLifetimeSeconds="1800"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' cacheLifetimeSeconds="1800"');</code>

Advanced

Advanced Options
<ul style="list-style-type: none"> • <code>debugging_enabled</code> • <code>galleryId</code> • <code>httpMethod</code> • <code>https</code>

`debugging_enabled`

Option name	<code>debugging_enabled</code>
Description	Enables or disables TubePress debugging. Keeping this enabled is a slight privacy risk, so if you are not experiencing difficulty with TubePress then feel free to disable it.
Provided with	All downloadable TubePress distributions
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress debugging_enabled="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' debugging_enabled="true"');</code>

galleryId

Option name	galleryId
Description	Explicitly set the unique identifier of this HTML element. By default, TubePress will assign a large random number to each element that it produces. This allows TubePress to differentiate and coordinate between multiple elements on the same page. By setting this option, you can choose a constant ID so that you can refer to the element via JavaScript.
Provided with	All downloadable TubePress distributions
Default value	<i>empty</i>
Valid values	Any string, though a positive integer is recommended.
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress galleryId="12345"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' galleryId="12345"');</code>

httpMethod

Option name	httpMethod
Description	Defines the HTTP method to use for most Ajax operations that TubePress performs. This can be useful to change when TubePress is used in web servers with tight security requirements.
Provided with	All downloadable TubePress distributions
Default value	GET
Valid values	GET or POST (case sensitive)
Supported provider(s)	Vimeo and YouTube
Shortcode example	<code>[tubepress httpMethod="POST"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' httpMethod="POST"');</code>

https

Option name	https
Description	Serve thumbnails and embedded video player over a secure connection. This is useful if you are running TubePress inside an HTTPS-only site.
Provided with	TubePress Pro only
Default value	false
Valid values	true or false
Supported provider(s)	YouTube
Shortcode example	<code>[tubepress https="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode(' https="true"');</code>

3.1.2 YouTube Options

On This Page

- Gallery Sources
 - `playlist`
 - `favorites`
 - `tag`
 - `user`
 - `youtubeMostPopular`
 - `youtubeRelated`
- Embedded Video Player
 - `autoHide`
 - `fullscreen`
 - `modestBranding`
 - `showRelated`
 - `youtubeClosedCaptions`
 - `youtubeDisableKeyboardControls`
 - `youtubePlayerTheme`
 - `youtubeShowAnnotations`
 - `youtubeShowPlayerControls`
- Video Feed
 - `developerKey`
 - `embeddableOnly`
 - `filter_racy`
- Video Meta Display
 - `rating`
 - `ratings`

Gallery Sources

General Options

- `playlist`
- `favorites`
- `tag`
- `user`
- `youtubeMostPopular`
- `youtubeRelated`

playlist

Gallery source	playlist
Description	Videos from a YouTube playlist. The playlist is defined by the value you supply to the <code>playlistValue</code> attribute.
Notes	Limited to 200 videos. Currently, only public playlists can be shown with TubePress.
Shortcode example	<code>[tubepress mode="playlist" playlistValue="6813408AE8D50E6F"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="playlist" playlistValue="6813408AE8D50E6F"');</code>
YouTube documentation	Click here

favorites

Gallery source	favorites
Description	“Favorite” videos of a YouTube user. The YouTube user is defined by the value you supply to the <code>favoritesValue</code> attribute.
Notes	Limited to 50 videos.
Shortcode example	<code>[tubepress mode="favorites" favoritesValue="3hough"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="favorites" favoritesValue="3hough"');</code>
YouTube documentation	Click here

tag

Gallery source	tag
Description	YouTube videos matching a search term. Supply space-separated search terms to the <code>tagValue</code> attribute. To search for an exact phrase, enclose the phrase in double quotes. Use the pipe character (<code> </code>) to indicate a boolean OR, and the minus character (<code>-</code>) to exclude a term.
Notes	Limited to approximately 1000 videos.
Shortcode example	<code>[tubepress mode="tag" tagValue=' "pittsburgh steelers" highlights -playoffs']</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="tag" tagValue=' "pittsburgh steelers" -highlights playoffs');</code>
YouTube documentation	Click here

user

Gallery source	user
Description	YouTube videos uploaded by a specific YouTube user. The YouTube user is defined by the value you supply to the <code>userValue</code> attribute.
Notes	Limited to 1000 videos.
Shortcode example	<code>[tubepress mode="user" userValue="3hough"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="user" userValue="3hough"');</code>
YouTube documentation	Click here

youtubeMostPopular

Gallery source	youtubeMostPopular
Description	The most popular YouTube videos, selected using a YouTube algorithm that combines many different signals to determine overall popularity. You can supply <code>all_time</code> or <code>today</code> to the <code>youtubeMostPopularValue</code> option to control the time frame of the results.
Shortcode example	<code>[tubepress mode="youtubeMostPopular" youtubeMostPopularValue="all_time"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="youtubeMostPopular" youtubeMostPopularValue="all_time"');</code>
YouTube documentation	Click here

youtubeRelated

Gallery source	youtubeRelated
Description	YouTube videos related to a specific YouTube video. The target YouTube video is identified by the value supplied to the <code>youtubeRelatedValue</code> option.
Shortcode example	<code>[tubepress mode="youtubeRelated" youtubeRelatedValue="ZTUVgYoeN_b"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="youtubeRelated" youtubeRelatedValue="ZTUVgYoeN_b"');</code>
YouTube documentation	Click here

Embedded Video Player

Embedded Video Player

- `autoHide`
- `fullscreen`
- `modestBranding`
- `showRelated`
- `youtubeClosedCaptions`
- `youtubeDisableKeyboardControls`
- `youtubePlayerTheme`
- `youtubeShowAnnotations`
- `youtubeShowPlayerControls`

`autoHide`

Option name	<code>autoHide</code>
Description	Automatically fade out elements of the video player after a few moments of playback.
Provided with	All TubePress distributions
Default value	<code>fadeOnlyProgressBar</code>
Valid values	<p>fadeBoth Fade both the progress bar and the video controls</p> <p>fadeNone Always show both the progress bar and the video controls</p> <p>fadeOnlyProgressBar Fade only the progress bar, but continue to show the video controls.</p>
Shortcode example	<code>[tubepress autoHide="fadeNone"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('autoHide="fadeNone")</code>
YouTube documentation	Click here

`fullscreen`

Option name	<code>fullscreen</code>
Description	Enable or disable the fullscreen playback option.
Provided with	All TubePress distributions
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Shortcode example	<code>[tubepress fullscreen="false"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('fullscreen="false")</code> ;
YouTube documentation	Click here

modestBranding

Option name	modestBranding
Description	If enabled, hides the YouTube logo from the video controls area.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Shortcode example	<code>[tubepress modestBranding="false"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('modestBranding="false"');</code>
YouTube documentation	Click here

showRelated

Option name	showRelated
Description	Toggles display of related videos when playback finishes.
Provided with	All TubePress distributions
Default value	true
Valid values	true or false
Shortcode example	<code>[tubepress showRelated="false"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('showRelated="false"');</code>
YouTube documentation	Click here

youtubeClosedCaptions

Option name	youtubeClosedCaptions
Description	Show closed captions by default.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Shortcode example	<code>[tubepress youtubeClosedCaptions="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('youtubeClosedCaptions="true"');</code>
YouTube documentation	Click here

youtubeDisableKeyboardControls

Option name	youtubeDisableKeyboardControls
Description	Disable all keyboard controls for the video.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Shortcode example	<code>[tubepress youtubeDisableKeyboardControls="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('youtubeDisableKeyboardControls="true"');</code>
YouTube documentation	Click here

youtubePlayerTheme

Option name	youtubePlayerTheme
Description	The color theme of the video player .
Provided with	All TubePress distributions
Default value	dark
Valid values	dark or light
Shortcode example	[tubepress youtubePlayerTheme="light"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('youtubePlayerTheme="light"');
YouTube documentation	Click here

youtubeShowAnnotations

Option name	youtubeShowAnnotations
Description	Show video annotations by default.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Shortcode example	[tubepress youtubeShowAnnotations="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('youtubeShowAnnotations="true"');
YouTube documentation	Click here

youtubeShowPlayerControls

Option name	youtubeShowPlayerControls
Description	Toggles display of the video controls.
Provided with	All TubePress distributions
Default value	showDelayed
Valid values	<p>hide Hide all video controls</p> <p>showImmediate Show the video controls. When the Flash player is used, Flash will load immediately.</p> <p>showDelayed Show the video controls. When the Flash player is used, Flash will load when video playback begins</p>
Shortcode example	[tubepress youtubeShowPlayerControls="fadeNone"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('youtubeShowPlay
YouTube documentation	Click here

Video Feed

Video Feed Options

- `developerKey`
- `embeddableOnly`
- `filter_racy`

`developerKey`

Option name	<code>developerKey</code>
Description	YouTube API developer key. YouTube will use this developer key for logging and debugging purposes if you experience a service problem on their end.
Provided with	All downloadable TubePress distributions
Default value	<i>hidden</i>
Valid values	Any valid YouTube API key
Shortcode example	<code>[tubepress developerKey="..."]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('developerKey="..."');</code>
YouTube documentation	Click here

`embeddableOnly`

Option name	<code>embeddableOnly</code>
Description	Some videos have embedding disabled. Enabling this option will exclude these videos from your galleries. Most users will never need to modify this option.
Provided with	All downloadable TubePress distributions
Default value	<code>true</code>
Valid values	<code>true</code> or <code>false</code>
Shortcode example	<code>[tubepress embeddableOnly="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('embeddableOnly="true"');</code>
YouTube documentation	Click here

`filter_racy`

Option name	<code>filter_racy</code>
Description	Filter out restricted content. YouTube performs the filtering much in the same manner as SafeSearch Filtering for Google WebSearch .
Provided with	All downloadable TubePress distributions
Default value	<code>moderate</code>
Valid values	none YouTube will not perform any filtering. moderate YouTube will filter some videos. strict YouTube will try to exclude all restricted videos.
Shortcode example	<code>[tubepress filter_racy="strict"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('filter_racy="st</code>
YouTube documentation	Click here

Video Meta Display

`rating`

Option name	<code>rating</code>
Description	Toggle display of the video's average rating.
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Shortcode example	<code>[tubepress rating="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('rating="true");</code>

`ratings`

Option name	<code>ratings</code>
Description	Toggle display of the number of times the video has been rated.
Provided with	All TubePress distributions
Default value	<code>false</code>
Valid values	<code>true</code> or <code>false</code>
Shortcode example	<code>[tubepress ratings="true"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('ratings="true");</code>

3.1.3 Vimeo Options

On This Page

- Gallery Sources
 - vimeoAlbum
 - vimeoAppearsIn
 - vimeoChannel
 - vimeoCreditedTo
 - vimeoGroup
 - vimeoLikes
 - vimeoSearch
 - vimeoLikes
- Embedded Video Player
 - playerColor
- Video Feed
 - vimeoKey
 - vimeoSecret
- Video Meta Display
 - likes

Gallery Sources

Gallery Sources

- vimeoAlbum
- vimeoAppearsIn
- vimeoChannel
- vimeoCreditedTo
- vimeoGroup
- vimeoLikes
- vimeoSearch
- vimeoLikes

vimeoAlbum

Gallery source	vimeoAlbum
Description	Videos from a Vimeo album. The album is identified by the value supplied to the vimeoAlbumValue option.
Shortcode example	<code>[tubePress mode="vimeoAlbum" vimeoAlbumValue="852694"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoAlbum" vimeoAlbumValue="852694"');</code>
Vimeo documentation Click here	

vimeoAppearsIn

Gallery source	vimeoAppearsIn
Description	Vimeo videos in which a given user appears. The Vimeo user is identified you supply to the vimeoAppearsInValue attribute.
Notes	Limited to 50 videos.
Shortcode example	<code>[tubepress mode="vimeoAppearsIn" vimeoAppearsInValue="dabrainkilla"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoAppearsIn" vimeoAppearsInValue="dabrainkilla"');</code>
Vimeo documentation	Click here

vimeoChannel

Gallery source	vimeoChannel
Description	Videos from a Vimeo channel. The channel is identified by the value supplied to the vimeoChannelValue option.
Shortcode example	<code>[tubepress mode="vimeoChannel" vimeoChannelValue="besthd"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoChannel" vimeoChannelValue="besthd"');</code>
Vimeo documentation	Click here

vimeoCreditedTo

Gallery source	vimeoCreditedTo
Description	Vimeo videos for which a particular user has either uploaded or appeared in. The Vimeo user is identified by the value supplied to the vimeoCreditedToValue option.
Shortcode example	<code>[tubepress mode="vimeoCreditedTo" vimeoCreditedToValue="thunderingherd"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoCreditedTo" vimeoCreditedToValue="thunderingherd"');</code>
Vimeo documentation	Click here

vimeoGroup

Gallery source	vimeoGroup
Description	Videos from a Vimeo group. The group is identified by the value supplied to the vimeoGroupValue option.
Shortcode example	<code>[tubepress mode="vimeoGroup" vimeoGroupValue="underachievers"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoGroup" vimeoGroupValue="underachievers"');</code>
Vimeo documentation	Click here

`vimeoLikes`

Gallery source	<code>vimeoLikes</code>
Description	Vimeo videos that a particular user has “liked.” The Vimeo user is identified by the value supplied to the <code>vimeoLikesValue</code> option.
Shortcode example	<code>[tubepress mode="vimeoLikes" vimeoLikesValue="besthd"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoLikes" vimeoLikesValue="besthd"');</code>
Vimeo documentation	Click here

`vimeoSearch`

Gallery source	<code>vimeoSearch</code>
Description	Vimeo videos that match a supplied search term. The search term is identified by the value supplied to the <code>vimeoSearchValue</code> option.
Shortcode example	<code>[tubepress mode="vimeoSearch" vimeoSearchValue="rainbow alligator"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoSearch" vimeoSearchValue="rainbow alligator"');</code>
Vimeo documentation	Click here

`vimeoLikes`

Gallery source	<code>vimeoUploadedBy</code>
Description	Vimeo videos uploaded by a particular user. The Vimeo user is identified by the value supplied to the <code>vimeoUploadedByValue</code> option.
Shortcode example	<code>[tubepress mode="vimeoUploadedBy" vimeoUploadedByValue="ehough"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('mode="vimeoUploadedBy" vimeoUploadedByValue="ehough"');</code>
Vimeo documentation	Click here

Embedded Video Player

Embedded Video Player

- `playerColor`

`playerColor`

Option name	<code>playerColor</code>
Description	The highlight color of the Vimeo video player.
Provided with	All TubePress distributions
Default value	000000
Valid values	Any six hexadecimal characters representing an HTML color .
Shortcode example	<code>[tubepress playerColor="A32638"]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('playerColor="A32638"');</code>
YouTube documentation	Click here

Video Feed

Video Feed Options

- `vimeoKey`
- `vimeoSecret`

`vimeoKey`

Option name	<code>vimeoKey</code>
Description	The API key that TubePress will use when communicating with Vimeo.
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	Any valid Vimeo API key
Shortcode example	<code>[tubepress vimeoKey="..."]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('vimeoKey="..."');</code>
Vimeo documentation	Click here

`vimeoSecret`

Option name	<code>vimeoSecret</code>
Description	The API secret that TubePress will use when communicating with Vimeo.
Provided with	All TubePress distributions
Default value	<i>empty</i>
Valid values	Any valid Vimeo API secret
Shortcode example	<code>[tubepress vimeoSecret="..."]</code>
Standalone PHP example	<code>TubePressPro::getHtmlForShortcode('vimeoSecret="..."');</code>
Vimeo documentation	Click here

Video Meta Display

likes

Option name	likes
Description	Toggle display of the number of times the video has been “liked”.
Provided with	All TubePress distributions
Default value	false
Valid values	true or false
Shortcode example	[tubepress likes="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('likes="true"');

3.1.4 Vimeo All Access Options

This page documents the options available with the *Vimeo All Access* add-on for TubePress.

vimeoApiAccessToken

Option name	vimeoApiAccessToken
Description	The API secret that TubePress will use when communicating with Vimeo.
Provided with	<ul style="list-style-type: none"> All cloud-based TubePress distributions <i>Vimeo All Access add-on</i> add-on
Default value	<i>empty</i>
Valid values	Any valid Vimeo API access token
Shortcode example	[tubepress vimeoApiAccessToken="..."]
Standalone PHP example	TubePressPro::getHtmlForShortcode('vimeoApiAccessT
Vimeo documentation	Click here

vimeoApiAccessTokenSecret

Option name	vimeoApiAccessTokenSecret
Description	The API secret that TubePress will use when communicating with Vimeo.
Provided with	<ul style="list-style-type: none"> All cloud-based TubePress distributions <i>Vimeo All Access add-on</i>
Default value	<i>empty</i>
Valid values	Any valid Vimeo API access token secret
Shortcode example	[tubepress vimeoApiAccessTokenSecret="..."]
Standalone PHP example	TubePressPro::getHtmlForShortcode('vimeoApiAccessT
Vimeo documentation	Click here

3.1.5 Flexible Thumbnail Row Options

This page documents the options available with the *Flexible Thumbnail Rows* add-on for TubePress.

flexibleThumbnailRows

Option name	flexibleThumbnailRows
Description	Automatically adjust the height of thumbnail rows in TubePress galleries, ensuring that the metadata (runtime, title, description, etc) for each video is completely visible
Provided with	<i>Flexible Thumbnail Rows</i> add-on
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress flexibleThumbnailRows="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('flexibleThumbnailRows="true"');

3.1.6 YouTube Black Bars Remover Options

This page documents the options available with the *YouTube Black Bars Remover* add-on for TubePress.

youtubeHideBlackBars

Option name	youtubeHideBlackBars
Description	Hides the “black bars” on most YouTube thumbnails.
Provided with	<i>YouTube Black Bars Remover</i> add-on
Default value	false
Valid values	true or false
Shortcode example	[tubepress youtubeHideBlackBars="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('youtubeHideBlackBars="true"');

3.1.7 QuickPlay Options

This page documents the options available with the *QuickPlay* add-on for TubePress.

enableQuickplay

Option name	enableQuickplay
Description	Replaces video thumbnail images with fully-functional embedded video players.
Provided with	<i>QuickPlay</i> add-on
Default value	false
Valid values	true or false
Supported provider(s)	Vimeo and YouTube
Shortcode example	[tubepress enableQuickplay="true"]
Standalone PHP example	TubePressPro::getHtmlForShortcode('enableQuickplay="true"');

3.2 Index

Documentation Index