**How To Create a Sandbox Environment For Wordpress**

When you play with different layouts, content delivery methods or backend logics with Wordpress without affecting your public facing site, it is handy to have a sandbox environment where you can go nuts without affecting your production environment.

There are many ways to create a sandbox environment. You can created it anywhere (e.g. local computer or entirely separate servers). In this post, I will focus on creating a sandbox environment within the same server by creating a subdomain with your hosting provider. Subdomains let you host multiple websites with one domain name.

This activity is enormously enjoyable as you can play with many different technologies and many configuration steps. You will get use tools such as cPanel, Linux server, MySQL, phpAdmin, and ETL tool. It is cool.

**Prerequisite**

Account with a hosting provider with Wordpress installed.

SSH access to your server. You need to check the provider’s documentation (for bluehost [here](https://my.bluehost.com/hosting/help/ssh-keygen)).

Set up database access (for bluehost here).

OK, let’s begin!

Steps

1. **Create a subdomain**

Create a subdomain. Most hosting providers let you create subdomains. If not, you probably should change the provider because it is I believe one of the critical factors for hosting provider selection. Each provider has a different way of creating it. Most of them offer the cPanel hosting account manager and you can use it to create it. When it comes to managing your account, cPanel is really your best friend. You can configure pretty much everything there.

This site is hosted on [bluehost](https://www.bluehost.com/). In bluehost, you can create a subdomain through cPanel as [this](https://my.bluehost.com/hosting/help/274). You can google how to for your provider.

1. **Install Wordpress in the subdomain**

You can do this through cPanel by clicking Wordpress installation under the subdomain. It will create a separate folder structure and database for the sandbox.

1. **Configure SSH Access to the server**

For SSH access, you need to check the provider’s documentation (for bluehost [here](https://my.bluehost.com/hosting/help/ssh-keygen)). It usually requires host server’s IP address, user name, password and private key. I use [putty](https://www.putty.org/) to connect from my Windows machine.

1. **Move entire wp-content folder to sandbox folder in the server**

The sandbox folder is created within the main folder (e.g. public\_html). You need to copy the folder called wp-content. First of all, make sure to rename the original wp-content folder in the sandbox folder so that you can roll it back if something goes wrong. It is a good development practice.

[cc lang="bash" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

# (1) go into sandbox folder and rename the original wp-content

cd public\_html/sandbox

mv wp-content wp-content-backup

# (2) copy the entire wp-content from production to sandbox

cp -a ~public\_html/wp-content ./sandbox/

[/cc]

1. **Create remote database access**

You should be able to do it from cPanel or myPhpAdmin. You need to check your provider’s documentation (bluehost [here](https://my.bluehost.com/hosting/help/89)).

1. **Migrate data from production to sandbox database**

There are many ways of doing this. My solution is to use an ETL tool since I am an ETL developer. I used the open source ETL tool, called [Talend Open Studio](https://www.talend.com/products/talend-open-studio/). In my opinion, is the best free ETL tool out there. I tried Clover and Pentaho in the past, but Talend absolutely kicks arse in terms of ease of use and functionality.

I wrote an extensive how to. The detailed steps are here: [**How To Migrate Wordpress MySQL Data From Production To Sandbox With Talend**](https://www.mydatahack.com/talend-how-to-migrate-wordpress-mysql-data-from-production-to-sandbox)

1. **Update two values in the option table**

After copying all the necessary data to the sandbox database, you still need to do change two values in the options table. This can be done from phpMyDamin (you can usually go there from cPanel). These steps can be incorporated into the ETL job, too. Changing database values is fun. So, I did it manually.

1. You need to change option\_value = 0 where option\_name = blog\_public. This will tick the option to discourage search engines from indexing this site. It is not a good SEO practice to let search engines index your sandbox.

Ok, now you login to wd-admin. Nah, it won’t work. You will get the error message below.

**WordPress Admin – Sorry, you are not allowed to access this page.**

A simple database update can resolve this error. Here is how.

1. Change the prefix of option\_name column value where it has the value <prefix>\_user\_roles. Wordpress identifies environments by prefix. Since we brought the exact copy of the production environment, the prefix for user\_roles is set to the production one. You need to change it to the sandbox one. If you do not do this, you don’t have admin right when you login to wp-admin.

It’s all done. Now, you can go to your sandbox url or log into wp-admin to see if you have the same production website.

Epic!