How To Set Up Apache Virtual Hosts on Ubuntu 12.04 LTS

*Authored by: **ASPHostServer Administrator** [asphostserver@gmail.com]*
Saved From: http://faq.asphosthelpdesk.com/article.php?id=164

About Virtual Hosts

Virtual Hosts are used to run more than one domain off of a single IP address. This is especially useful to people who need to run several sites off of one virtual private server. The sites display different information to the visitors, depending on with which the users accessed the site.There is no limit to the number of virtual hosts that can be added to a server.

Set Up

The steps in this tutorial require the user to have root privileges. You can see how to set that up in the here Initial Server Setup. You can implement whatever username suits you.

 Additionally, you need to have apache already installed and running on your virtual server
 If this is not the case, you can download it with this command:

sudo apt-get install apache2

Step One— Create a New Directory

The first step in creating a virtual host is to a create a directory where we will keep the new website"s information.

 This location will be your Document Root in the Apache virtual configuration file later on. By adding a -p to the line of code, the command automatically generates all the parents for the new directory.

sudo mkdir -p /var/www/example.com/public_html

You will need to designate an actual DNS approved domain, or an IP address, to test that a virtual host is working. In this tutorial we will use example.com as a placeholder for a correct domain name.

However, should you want to use an unapproved domain name to test the process you will find information on how to make it work on your local computer in Step Seven.

Step Two—Grant Permissions

We need to grant ownership of the directory to the user, instead of just keeping it on the root system.

sudo chown -R $USER:$USER /var/www/example.com/public_html

Additionally, it is important to make sure that everyone will be able to read our new files.

sudo chmod -R 755 /var/www

Now you are all done with permissions.

Step Three— Create the Page

Within our configurations directory, we need to create a new file called index.html

sudo nano /var/www/example.com/public_html/index.html

We can add some text to the file so we will have something to look at when the IP redirects to the virtual host.

<html>

<head>

<title>www.example.com</title>

</head>

<body>

<h1>Success: You Have Set Up a Virtual Host</h1>

</body>

</html>

Save and Exit

Step Four—Create the New Virtual Host File

The next step is to set up the apache configuration. We"re going to work off a duplicate—go ahead and make a copy of the file (naming it after your domain name) in the same directory:

sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/example.com

Step Five—Turn on Virtual Hosts

Open up the new config file:

sudo nano /etc/apache2/sites-available/example.com

We are going to set up a virtual host in this file.

The first step is to insert a line for the ServerName under the ServerAdmin line.

ServerName example.com

The ServerName specifies the domain name that the virtual host uses.

If you want to make your site accessible from more than one name (for example, with www in the URL), you can include the alternate names in your virtual host file by adding a ServerAlias Line. The beginning of your virtual host file would then look like this:

<VirtualHost *:80>

ServerAdmin webmaster@example.com

ServerName example.com

ServerAlias www.example.com

[...]

The next step is to fill in the correct Document Root. For this section, write in the extension of the new directory created in Step One. If the document root is incorrect or absent you will not be able to set up the virtual host.

The section should look like this:

DocumentRoot /var/www/example.com/public_html

You do not need to make any other changes to this file. Save and Exit.The last step is to activate the host, with the built in apache shortcut:

sudo a2ensite example.com

Step Six—Restart Apache

We"ve made a lot of the changes to the configuration, and the virtual host is set up. However none of the changes that we made will take effect until Apache is restarted. Use this command to restart apache:

sudo service apache2 restart

You may see an error along the lines of

Could not reliably determine the server's fully qualified domain name, using 127.0.0.1 for ServerName

The message is just a warning, and you will be able to access your virtual host without any further issues.

Optional Step Seven—Setting Up the Local Hosts

If you have pointed your domain name to your virtual private server"s IP address you can skip this step—you do not need to set up local hosts. Your virtual hosts should work. However, if want to try out your new virtual hosts without having to connect to an actual domain name, you can set up local hosts on your computer alone.

 For this step, make sure you are on the computer itself, not your server instance.

 To proceed with this step you need to know your computer"s administrative password, otherwise you will be required to use an actual domain name to test the virtual hosts.

 If you are on a Mac or Linux, access the root user (su) on the computer and open up your hosts file:

nano /etc/hosts

If you are on a Windows Computer, you can find the directions to alter the host file on the Microsoft site.

You can add the local hosts details to this file, as seen in the example below. As long as that line is there, directing your browser toward, say, example.com will give you all the virtual host details for the corresponding IP address.

Host Database

#

localhost is used to configure the loopback interface

when the system is booting. Do not change this entry.

##

127.0.0.1 localhost

#Virtual Hosts

12.34.56.789 example.com

However, it may be a good idea to delete these made up addresses out of the local hosts folder when you are done to avoid any future confusion.

Step Eight—RESULTS: See Your Virtual Host in Action

Once you have finished setting up your virtual host, you can see how it looks online. Type your ip address into the browser (ie. http://12.34.56.789)

 Good Job!

Creating More Virtual Hosts

To add more virtual hosts, you can just repeat the process above, being careful to set up a new document root with the appropriate domain name, and then creating and activating the new virtual host file.