Adding multiple VirtualHosts hosting on fresh installed Debian GNU / Linux

Author : admin

Nowdays most of my new (fresh) Linux server configurations are installed with Debian Linux.

Through the years I worked with most major *GNU / Linux distributions*. Though intalling Apache for multiple domain VirtualHost-ing is almost equally easier to set up on all distros I tried, (Slackware, Redhat, Fedora) etc., I found Debian to be most convenient in terms of freqeuent easy updates and general security.

Every time I configure a new host which is supposed to host a dozen of websites with Apache webserver and a DB backend, it is of course necessery to enable the server to have **support multiple domain VirtualHosts**.

I thought there are people out who look to configure Multiple domains on fresh installed Apache webserver and this how this short post get born.

I will explain hereby in short *how I configure VirtualHosts on new Debian Linux servers with fresh installed Apache*.

All required to have a working many domains hosted VirtualHosts on Debian is:

1. Have installed Apache serve package

apt-get --yes install apache2

This would install all packages necessery for VirtualHost-ing.
After *apache2* installed the system should have at least this packages present.

dpkg -l |grep -i apache2
ii apache2-mpm-prefork 2.2.16-6+squeeze7 Apache HTTPServer - traditional non-threaded model
ii apache2-utils 2.2.16-6+squeeze7 utility programs for webservers
ii apache2.2-bin 2.2.16-6+squeeze7 Apache HTTPServer common binary files
ii apache2.2-common 2.2.16-6+squeeze7 Apache HTTPServer common files
ii libapache2-mod-php5 5.3.3-7+squeeze14 server-side, HTML-embedded scripting language (Apache 2 module)

Nowadays, having enabled **mod_rewrite** is necessery in almost any website, so the next thing I usually do is **enable mod_rewrite webserver module**.

ln -sf /etc/apache2/mods-available/rewrite.load /etc/apache2/mods-enabled/rewrite.load

apache2ctl -k restart

By default there is an *index.html* page containing the annoying phrase **It Works!**

I really dislike this default page and many times I start configuring a server, I wonder how to remove it; if you're like me before doing anything other I advice you edit **/var/www/index.html** to change it to **Coming Soon** or just substitute the file with some nice looking *Coming Soon* page (custom page) ...

Once this is done, I proceed adding as many Virtualhosts as I need with the respective *Virtualhost* names. To Do so on Debian, just create new Vhost config files in files **/etc/apache2/sites-available/yoursite.com**, **/etc/apache2/sites-available/yoursite1.com** etc.br />
Before creating any other *VHosts*, I usually edit the main webserver VirtualHost which is located in */etc/apache2/sites-available/000-default*, there in the VirtualHost section normally need to add proper:

ServerName and **ServerAlias** variables as well as change *DocumentRoot* to whatever the default server host Virtualhost directory will be.

An example of *000-default* Vhost config I do looks like so:

ServerName server-main-host-name.com
ServerAlias www.server-main-host-name.com server-main-host-name.com
DocumentRoot /var/www
....

Onwards add the same *ServerName server-main-host-name.com* as a new line in **/etc/apache2/apache2.conf** config

Now for those not too unfamiliar with VirtualHost types, it is useful to say there are two ways of VirtualHosts:

- *IP Based VirtualHost*

  and
- *Host Based VirtualHosts*

IP Based VirtualHosts are added by using Apache directive syntax:

ServerName ....
ServerAlias ....

whether *Host Based VirutalHosts* are added by using in config file, the IP address on which the respective Vhost will reside:

ServerName ....
ServerAlias ....

Host Based *VirtualHosts* directive syntax can be either in form:

a)*Virtualhost* *
or
b) *Virtualhost port_number (Virtualhost 80, VirtualHost 90)* etc.

If a host is configured with directive , this means it will listen for incoming connections on any port Apache is configured to listen on, whether if used with a concrete port number it will only enable VirtualHosts for whole Apache server on the concrete port.

Based on the configuration, *VirtualHost 80* or *Virtualhost **, the variable which will enable globally on the Apache server multiple VirtualHosts has to be modified e.g.:
Whether VirtualHost with port number is configured , **NameVirtualHost 80** should be used or otherwise **NameVirtualHost ***

Once you choose the type of Virtualhost-ing, just continue on adding the VirtualHosts ...
In the first created VirtualHost config file, let's say **/etc/apache2/sites-available/first-virtualhost.com**

NameVirtualHost * has to be added as first line in file; in other words the file content should look similar to:

NameVirtualHost *

ServerAdmin hipo_aT_pc-freak.net AddDefaultCharset UTF-8 DocumentRoot /var/www/ ServerName pc-freak.net ServerAlias www.pc-freak.net....

The same steps has to go for all domain names in separate files except the variable **NameVirtualHost *** should not be added in the rest of new created Vhosts.

Many of the *new configured Debian + Apache servers* does not require support for SSL, therefore where SSL support is not necessery I prefer disabling it.
To do so it is necessery to comment out everything dealing with Secure Socket Layer in **/etc/apache2/ports.conf**, as of time of writting lines to comment are:

If you add NameVirtualHost *:443 here, you will also have to change
the VirtualHost statement in /etc/apache2/sites-available/default-ssl
to
Server Name Indication for SSL named virtual hosts is currently not
supported by MSIE on Windows XP.

Listen 443

Listen 443