

Hentzenwerke Whitepaper Series

Setting Up Virtual Hosts in Apache (A Tutorial for Windows Web Admins)

By Whil Hentzen

One common need of Web site administrators is to host multiple Web sites on the same physical machine. This functionality is called 'virtual hosting'. This document explains how to set up virtual hosting with the Apache Web server running on Fedora Core 2, using a similar setup on Microsoft's Internet Information Server as a reference point.

1. Preface

1.1 Copyright

Copyright 2004 Whil Hentzen. Some rights reserved. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs License, which basically means that you can copy, distribute, and display only unaltered copies of this work, but in return, you must give the original author credit, you may not distribute the work for commercial gain, nor create derivative works based on it without first licensing those rights from the author. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/2.0/>.

1.2 Revisions

1.2.1 History

Version	Date	Synopsis	Author
1.0.0	2004/7/21	Original	WH

1.2.2 New version

The newest version of this document will be found at www.hentzenwerke.com.

1.2.3 Feedback and corrections

If you have questions, comments, or corrections about this document, please feel free to email me at 'books@hentzenwerke.com'. I also welcome suggestions for passages you find unclear.

1.3 References and acknowledgements

Thanks to the great information in the documentation section of the Apache Web site, at www.apache.org.

1.4 Disclaimer

No warranty! This material is provided as is, with no warranty of fitness for any particular purpose. Use the concepts, examples and other content at your own risk. There may be errors and inaccuracies that in some configurations may be damaging to your system. The author(s) disavows all liability for the contents of this document.

Before making any changes to your system, ensure that you have backups and other resources to restore the system to its state before making those changes.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark. Naming of particular products or brands should not be seen as endorsements.

1.5 Prerequisites

This document was written using Fedora Core 2.0 and assumes a beginner's familiarity with use of Linux via the GUI and the Command Window, an beginner's knowledge of virtual hosts set up on Microsoft's IIS, and a beginner's familiarity with the Apache Web server.

2. What you want

You've got two domains, www.a.com and www.b.com. You want to host the websites for both [a.com](http://www.a.com) and [b.com](http://www.b.com) on a single physical box, using a single copy of Apache. This is called virtual hosting. This document describes how to do so using Fedora Core 2.

3. What you need

In order to follow along with the examples presented in this document, you will need a single box running the Apache Web server on Fedora Core 2, although earlier versions of Fedora, including Red Hat Linux are substantially similar. This box will need to be connected to the Internet either directly or through a router that performs NAT. In the first case, the box has an external IP address attached to it, such as 11.22.33.44, through the network card configuration.

In the second case, the box has an internal IP address, such as 192.168.1.1, and is connected to a router that has the 11.22.33.44 external IP address pointed to it. In this second case, the router receives requests on 11.22.33.44 and sends them on to the 192.168.1.1 IP.

In both cases, the DNS entries for `www.a.com` and `www.b.com` will both look (roughly) like this:

```
www.a.com    11.22.33.44
www.b.com    11.22.33.44
```

Both domains are pointed to the same IP address since they're both going to the same box. It'll be up to the Web server software running on the box to receive requests for each domain, and then route requests for `www.a.com` to the `a.com` virtual host, and send requests for `www.b.com` to the `b.com` virtual host.

4. How IIS works with multiple hosts

In order to provide a frame of reference with the Windows set up for those of you who have set up multiple hosts in IIS, I'll first review how multiple hosts are configured on IIS under Windows. This example shows IIS 4.0 running on NT 4.0; newer versions of IIS running on newer versions of Windows work similarly although the user interface differs.

Specifically, I'll use the domain, `www.linuxtransfer.com`, set up as a virtual host on a machine named `jeeves` that has an IP address of 192.168.1.2 (behind a router running NAT.)

First, the Web server machine, `jeeves`, is displayed in the tree view in the left pane of the Microsoft Management Console, and all servers (Web as well as other servers, such as email and FTP) are displayed as items in the right pane. In **Figure 1**, the Linux Transfer Web site and the default SMTP server are two of a number of servers running on the machine named 'jeeves'.

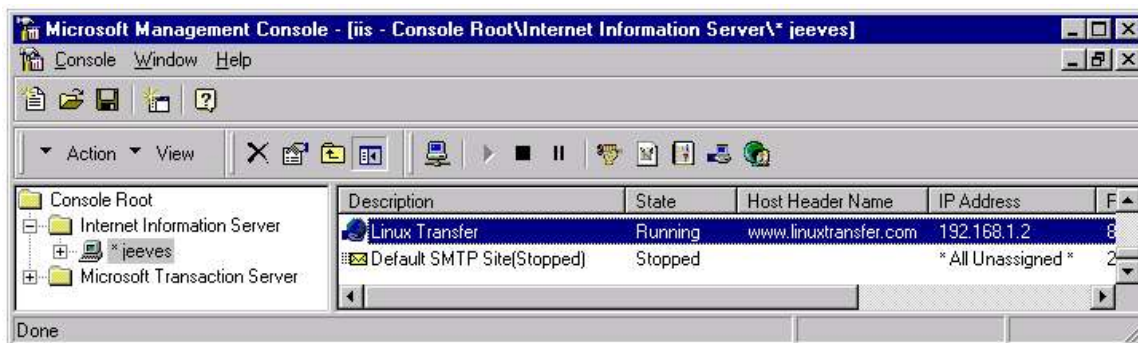


Figure 1. The Microsoft Management Console uses a tree view to display the servers on a machine.

Right-clicking on the Linux Transfer item in the right pane displays a context menu. Selecting the "Properties" menu item from that context menu opens the Properties dialog as shown in **Figure 2**.

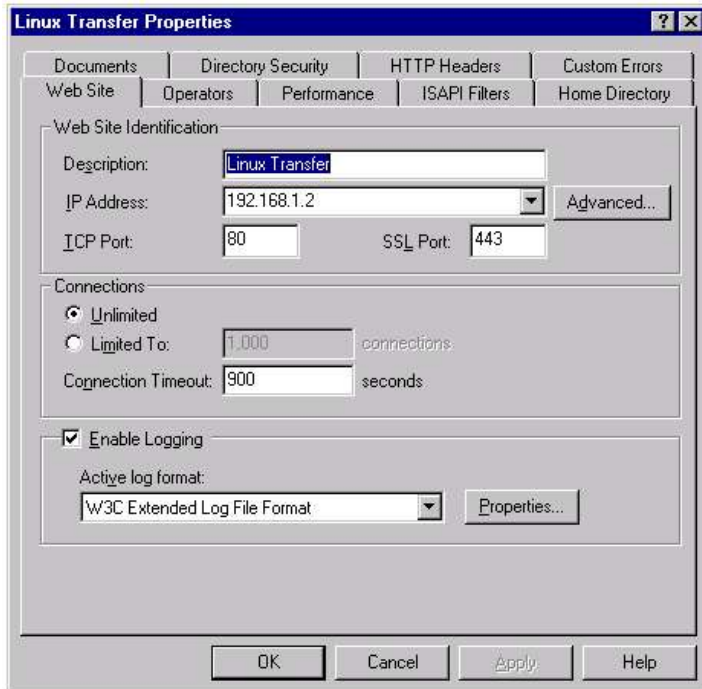


Figure 2. The Properties dialog for a particular server displays information about the server.

You'll see that the description, "Linux Transfer" from Figure 1, is also displayed in the Description text box under the Web Site Identification group.

The entry in the IP Address identifies the IP address of the physical box. Here, jeeves is behind a router that is running NAT, so the IP address of the physical machine is an internal address, namely, 192.168.1.2.

As an aside, if this box had multiple network interface cards, there would have been multiple choices in the IP Address combo box.

Attributes of this IP address are found in the dialog displayed by clicking the Advanced button to the right of the IP Address combo box, as shown in **Figure 3**.

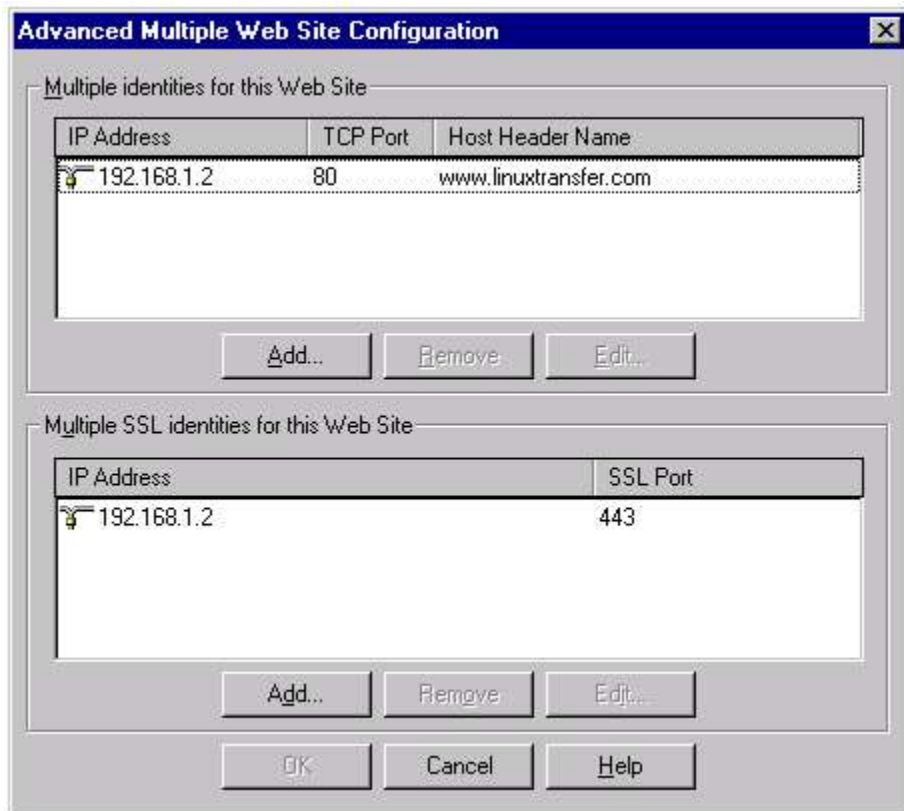


Figure 3. The Advanced Multiple Web Site Configuration dialog displays each IP address for a site.

Content for a domain can come from one of three places. The choice of "A directory located on this computer" in the top option group of the Home Directory tab, as shown in **Figure 4**, indicates that the content is located in a directory structure on this computer, as opposed to on another computer.

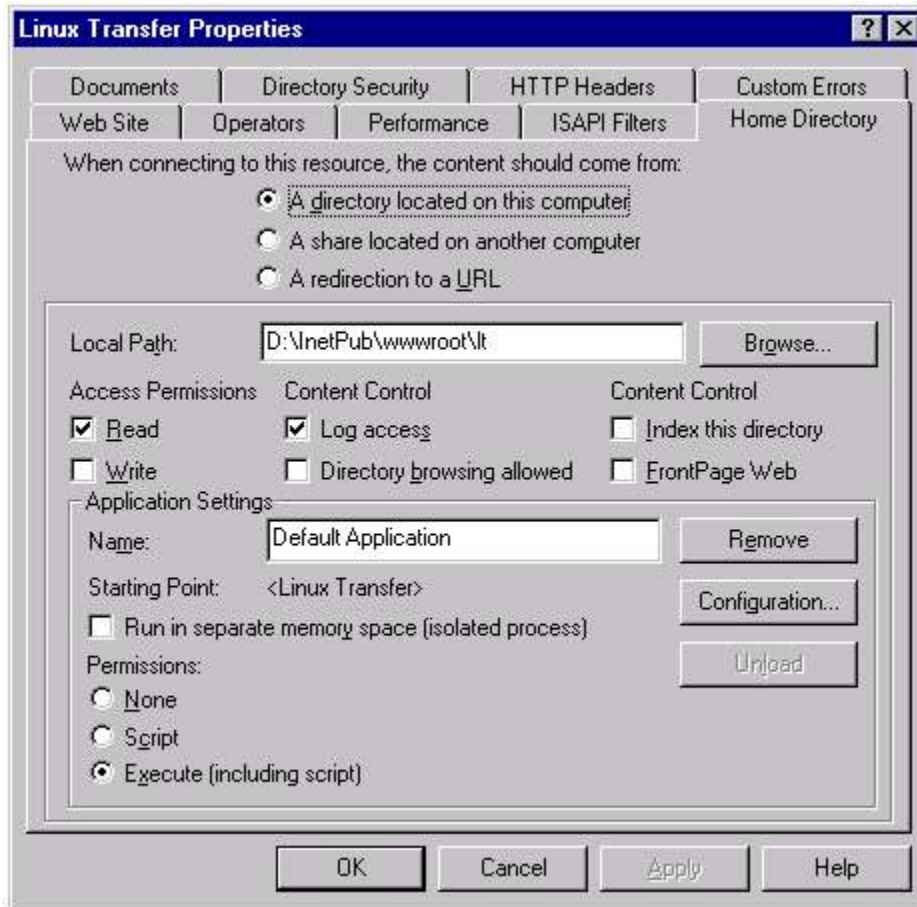


Figure 4. The Home Directory tab allows you to define where content for this Web site is going to come from.

The root of the directory structure is displayed in the Local Path text box. By default, IIS is configured to serve content in the c:\inetpub\wwwroot directory. The virtual host for www.linuxtransfer.com points to a different location, namely d:\inetpub\wwwroot\lt. Other virtual hosts served by this computer could be found in their own directory structures and similarly identified.

For example, if you were hosting www.a.com and www.b.com in IIS, you might have had directories named

```
\inetpub\wwwroot\a
```

and

```
\inetpub\wwwroot\b
```

and, in the properties for each Web site, pointed to those directories with the Local Path text box.

The Documents tab, as shown in **Figure 5**, simply identifies which HTML file or files are automatically served to users who enter the domain without identifying a specific file in the URL. For example, if you pointed your browser to 'www.linuxtransfer.com', the index.htm page, if it existed (and hopefully it does!), would be automatically be displayed. If it didn't exist, the Web server would then look for "Default.htm" next.

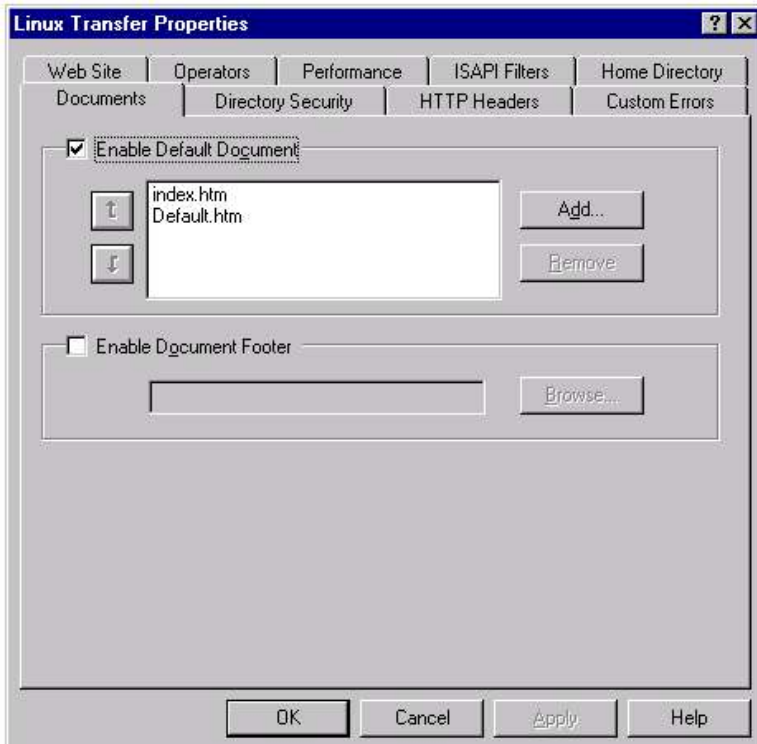


Figure 5. The Documents tab allows you to define which documents are served to Web site visitors by default.

The magic of virtual hosts is found in the HTTP Headers tab of the Properties dialog as shown in **Figure 6**. In the center of the tab, the entry in the Custom Content Headers list,

Content Location `http://www.linuxtransfer.com`

maps these virtual host properties, such as the Local Path, to the domain name.



Figure 6. The HTTP Headers tab allows you to define custom headers for virtual Web sites.

The values in the Content Location list box are edited via the Edit button to the right of the list box. The editing dialog is shown in **Figure 7**.



Figure 7. Use the Add/Edit Custom HTTP Header dialog to define headers for a virtual host.

So that's a quick look at how a virtual host is configured in IIS. Before we take a look at where the corresponding values go in Apache, however, it'd be best to review how Apache handles just a single host.

5. Preparing content directories for multiple virtual hosts

Apache's default configuration points requests to the default directory of `/var/www/html`, which is similar to the `c:\inetpub\wwwroot` value in IIS. Just like with Windows, you'll probably want to set up multiple directories, one for each virtual host. One way of doing this is by creating subdirectories under `/var/www/html`, such as

```
/var/www/html/a
```

and

`/var/www/html/b`

for `www.a.com` and `www.b.com`, respectively.

But that's not all that's involved in configuring multiple virtual hosts.

Now would be a good time to set up directories under `/var/www/html`, or wherever you're going to put the directories that will host your content for each site. You should also create default HTML files for each virtual host. They don't have to be fancy – a simple sentence saying “You are seeing the 'index.htm' page for virtual host `www.a.com`.” is plenty sufficient.

6. How to configure Apache for multiple hosts

You can set up multiple hosts in Apache in one of two different ways. The first way is to set up your first site as if it was going to be the only one on the box, and then add a second one as a 'virtual host', and configure the virtual host all over again.

The second path, and, if you think there's a possibility that you're going to set up more than one site on this box, the better choice, is to define the common attributes of all of these sites first, using what Apache calls a 'default virtual host'. Then, set up your first site as a virtual host that takes its default settings from the default virtual host definition. You can then customize the first site's settings as need be.

When it comes time to add a second site, you simply set it up as another virtual host, just like you did with the first site. The second site, too, will inherit its default settings from the default virtual host definition, and you can again customize the second site's settings as desired.

In the following discussion, I'll be using a sample Web site for the domain, `www.hidbigo.com`, that sits behind a NAT-enabled router. The IP address for the network interface card is an internal address, `10.0.0.2`. The router is configured to map the external IP address for `www.hidbigo.com` to the NIC's internal address.

6.1 Configuration via the GUI

Setting up the default virtual host is done in the Virtual Hosts tab of the HTTP configuration GUI widget, available through the Start | System Settings | Server Settings | HTTP menu, as shown in **Figure 8**.

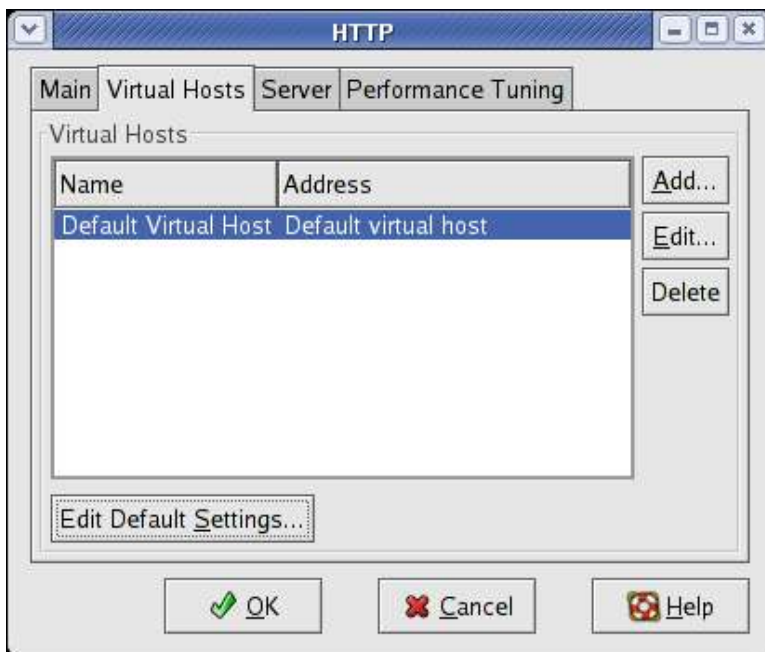


Figure 8. The Virtual Hosts tab of the HTTP configuration widget allows you to define the default settings for all virtual hosts as well as individual sites.

Click on the Edit Default Settings button at the bottom of the screen (not the 'Edit' button on the right), and configure the settings as desired. A detailed discussion of each of these settings is beyond the scope of this document, so we'll assume that the settings are either good enough, or that you've gone ahead and made the changes you want. (For all practical purposes, the default settings 'out of the box' are pretty good and if this is your first Web server under Apache, you probably don't have to change anything.)

Now it's time to set up your first virtual host. Click "Add..." on the left side of the Virtual Hosts tab to bring forward the Virtual Host Properties screen, as shown in **Figure 9**.

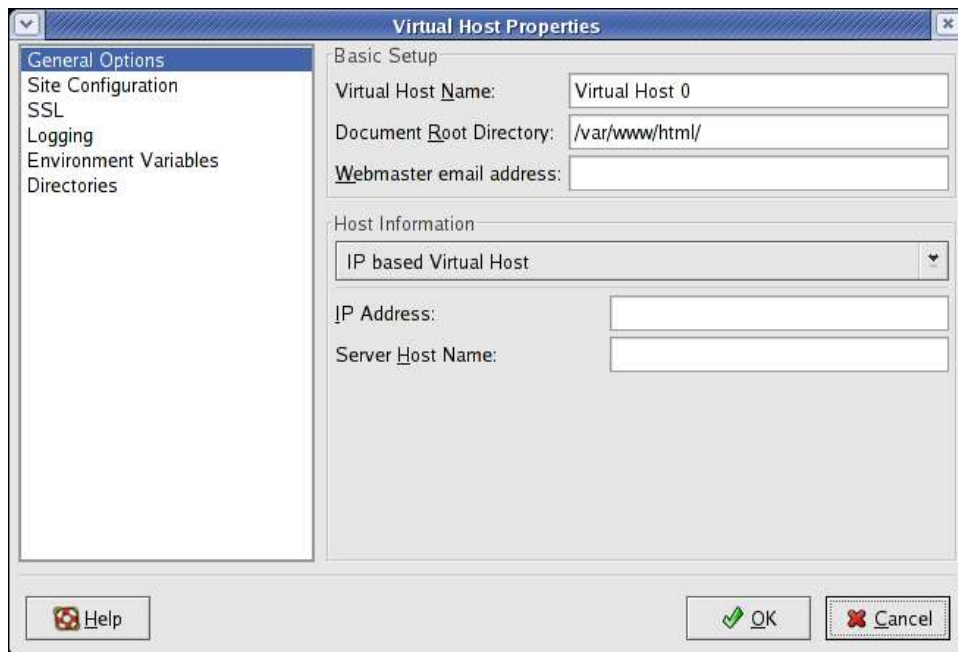


Figure 9. The Virtual Host Properties dialog allows you to set custom properties for a single virtual host.

The "General Options" item should be selected in the list box on the left side of the screen. Change the value in the "Virtual Host Name" text box to something memorable that will identify this host to you. You could just use the name of the domain, such as 'www.hidbigo.com', or you could use an abbreviation like 'hidbigo'.

Next, change the value in the "Document Root Directory" text box to reflect the directory that will contain the content for this host.

Finally, choose the type of host - IP based Virtual Host or Name-based Virtual Host from the Host Information combo box. (Since you're setting up an actual host, the choice of Default Virtual Host isn't applicable.) Which to choose?

The Apache documentation says that "Virtual hosts can be "IP-based", meaning that you have a different IP address for every web site, or "name-based", meaning that you have multiple names running on each IP address. The fact that they are running on the same physical server is not apparent to the end user."

If each of my virtual hosts was using a different IP address, I'd selected "IP based", and make the entries as shown in **Figure 10**.

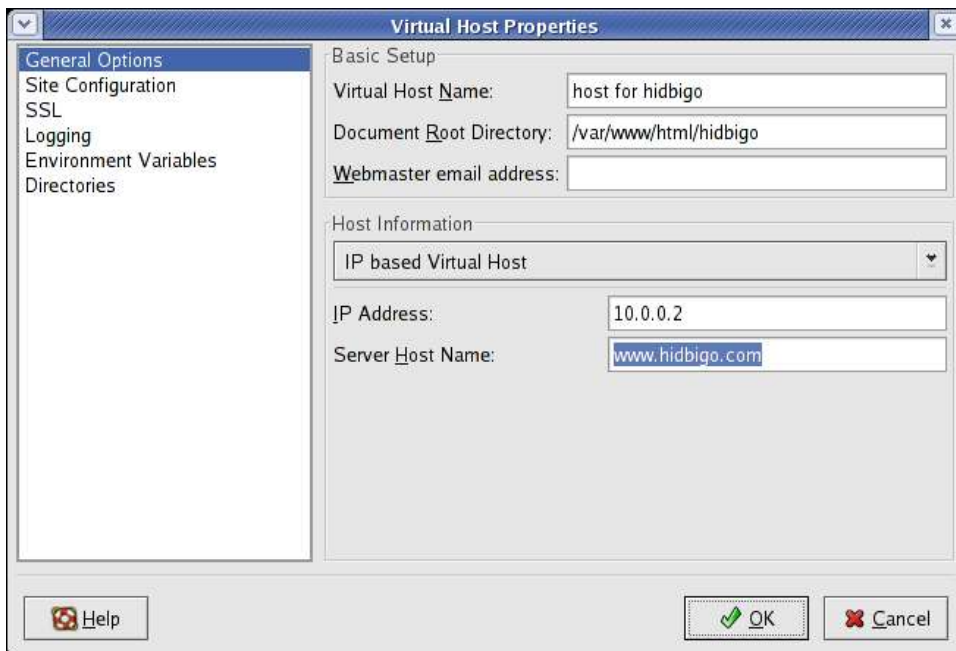


Figure 10. Setting up entries for an IP based Virtual Host.

Upon clicking OK, the HTTP properties dialog would look like Figure 11.

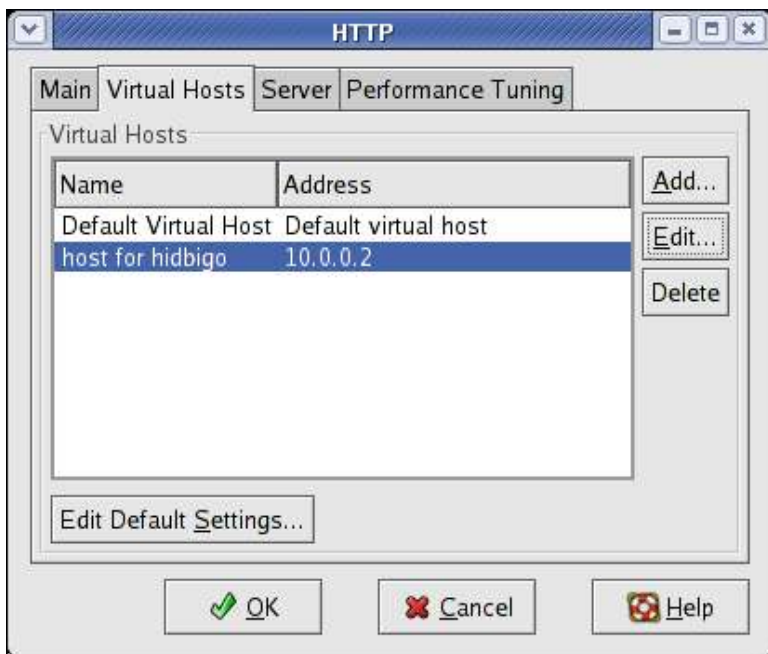


Figure 11. The resulting entry for an IP based Virtual Host.

But I'm not. I just mentioned those to show you what you might see in that scenario. In this example, however, www.hidbigo.com (and, eventually, other domains) will all reside on the same box, using the same external IP address, so the choice is Name-based Virtual Host.

Selecting "Name-based Virtual Host" changes the controls below the combo box as shown in Figure 12.

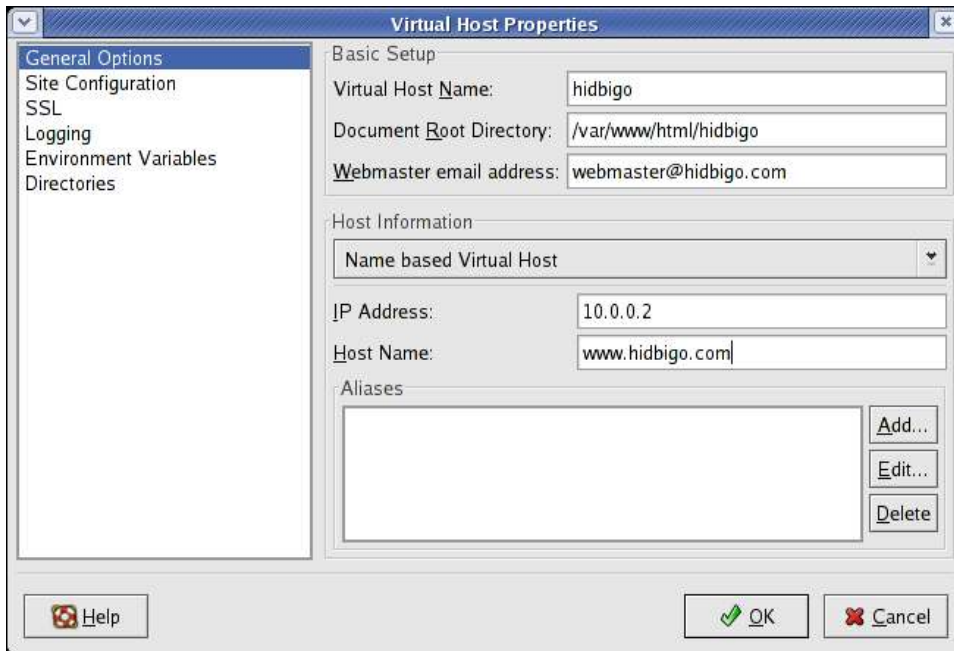


Figure 12. Setting up entries for an Name-based Virtual Host.

I made the entries as shown. Doing so, and then clicking OK, the HTTP properties dialog would look like Figure 13.

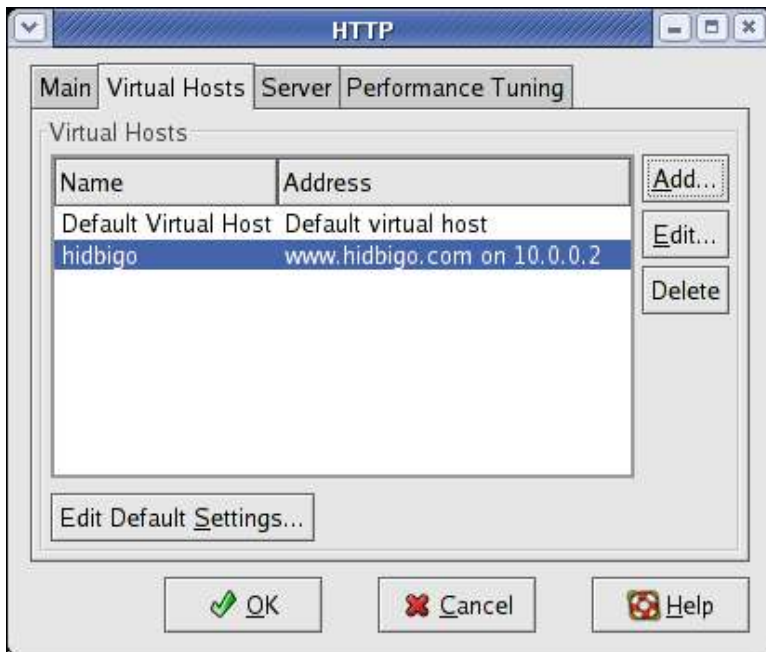


Figure 13. The resulting entry for an Name based Virtual Host.

6.2 Configuring via the httpd configuration file

There have been some reported problems with the HTTP GUI widget not saving settings properly, so it would be useful to show how to do the same thing by simply using the Apache configuration file.

Fedora Core's Apache config file is `/etc/httpd/conf/httpd.conf`, and needs to be edited as root.

At the end of the file is a section on Virtual Hosts, all commented out. The following directives will set up two name-based virtual hosts on this machine.

```
# Listen for virtual host requests on all IP addresses
NameVirtualHost *:80

<VirtualHost *:80>
DocumentRoot /var/www/html/hidbig0
ServerName www.hidbig0.com
DirectoryIndex default.htm
# Other directives here
</VirtualHost>

<VirtualHost *:80>
DocumentRoot /var/www/html/domainnumber2
ServerName www.domainnumber2.com
DirectoryIndex default.htm
# Other directives here
</VirtualHost>
```

Note that I added one directive to each virtual domain definition, that of "DirectoryIndex". This directive identifies the default document that will be served up if a visitor navigates to the domain but doesn't explicitly include a filename in the URL they typed. Thus, the directive

```
DirectoryIndex default.htm
```

will cause entering 'www.hidbig0.com' in a browser to display the 'default.htm' file.

7. Restarting Apache so it will recognize your changes

Finally, restart the Apache server. You can do this through the Services widget in the GUI (available through Start | System Settings | Server Settings | Services) as shown in **Figure 14**.

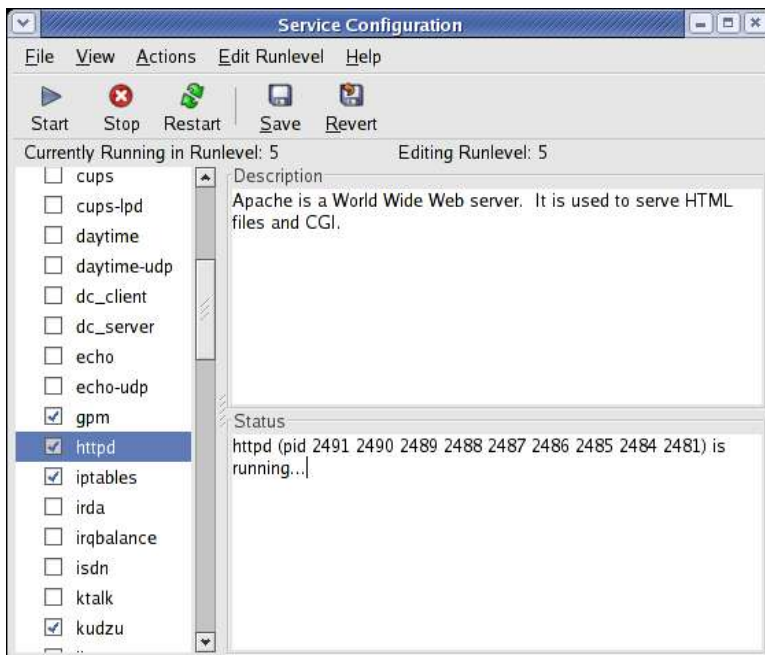


Figure 14. The Services widget in the GUI allows you to start and stop services visually.

Or you can issue the command

```
services httpd restart
```

as root, and now your virtual host will be ready to serve up content from its own directory.

8. Where to go for more information

This free whitepaper is published and distributed by Hentzenwerke Publishing, Inc. We have the largest lists of "Moving to Linux", OpenOffice.org, and Visual FoxPro books on the planet.

We also have oodles of free whitepapers on our website and more are being added regularly. Our Preferred Customer mailing list gets bi-monthly announcements of new whitepapers (and gets discounts on our books, first crack at special deals, and other stuff as we think of it.)

Click on “Your Account” at www.hentzenwerke.com to get on our Preferred Customer list.

If you found this whitepaper helpful, check out these Hentzenwerke Publishing books as well:

**Linux Transfer for Windows® Network Admins:
A roadmap for building a Linux file and print server
Michael Jang**

**Linux Transfer for Windows® Power Users:
Getting started with Linux for the desktop
Whil Hentzen**