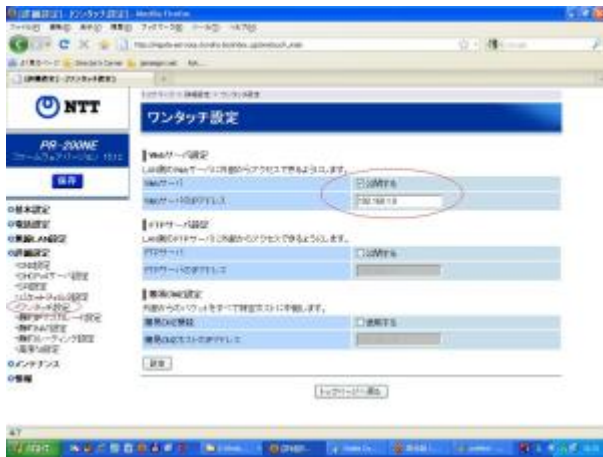


Adventure Setting Up a Home Web Server



A local company gave me a Pentium 4 class PC with Microsoft Server 2003 on it. The hard drive is only 40 gigabytes, but big enough for what I wanted to do with it. I thought to use it to setup my own webserver to host a Japanese language site to promote my employer's local English teaching and translation services.

I've been working with PCs since 1987 but I had no knowledge about setting up web servers. Any time I learn something new, it is always during a hands-on situation, actually doing it, and not just from books or web articles. This post documents what I learned and how I did it.

Using the PC as is would have violated Microsoft's licensing agreement, so I promptly got rid of MS Server 2003 and installed [Ubuntu Server Edition](#) in its place. Ubuntu Server has Apache, PHP, and MySQL – everything I need to set up a WordPress site.

Following the tutorial on <http://net.tutsplus.com/tutorials/php/how-to-setup-a-dedicated-web-server-for-free/> was pretty easy. However using Shorewall for the server firewall didn't work for me. I added firewall rules using UFW instead. The tutorial was for an earlier version of Ubuntu Linux and the Shorewall configuration blocked SSH access. Specifically, I added the firewall rules to allow port 80 for web access, and port 22 for ssh access, by entering the following commands:

```
sudo ufw enable
sudo ufw allow 80
sudo ufw allow 22
```

To assign the home webserver to a domain name, I got a free domain from [dyndns.com](#) and pointed it to the external IP address of the router. This was easy to do because it shows you the router's external IP address when you register.

Installing the Ubuntu Linux server was pretty simple thanks to clear

instructions on the tutorial page and Ubuntu's typical user-friendly installation. In addition to what the tutorial had, I also installed PHPMyAdmin which is the configuration program to set up the MySQL database for WordPress, and I installed ddclient which keeps my IP address in sync with the domain name I got from dyndns.com.

```
sudo apt-get install phpmyadmin
sudo apt-get install ddclient
```

To view PHP info on the new server, I created info.php in /var/www with the contents:

```
< ?php phpinfo(); ?>
```

and opened the file in a browser on the network: http://192.168.1.9/info.php
Cool! All sorts of information was displayed!

The part that took the longest was figuring out how to set up the NTT PR-200 NE router for port forwarding. The interface of the router's control panel is only in Japanese. But I finally figured it out after much trial and error.



NTT PR-200NE router control panel. The circled sections show the configuration I enabled.

The next problem was accessing the webserver. Every time I tried, I would get the NTT router log in screen instead of the web site! After much googling for the answer, I found this from <http://www.boutell.com/newfaq/creating/hostmyown.html>

"I followed all the steps and I get my router's login page instead of my home page!"

You are trying to access your website by name from behind your router (from one of your own PCs). With some consumer-grade routers, this does not work because the router automatically assumes any web connection to itself from inside your network is an attempt to log into the router's configuration interface. It's a pain, but this fail-safe mechanism does prevent you from locking yourself out of your router's web interface. So test from outside your own home network or have a friend do that for you. If you can access your home-hosted website from someone else's computer, then you don't have a problem. If you want to access your site from a computer behind your router, you'll have to access it at its static local IP address instead of by name.

This was the key to the problem! In order to view the website on the home web

server, I had to access the site using a laptop with WIFI and connect to a wireless network in the neighborhood which is of course outside my own network. This worked! I also learned that I could access the site using Tor which is a proxy network to enable me to access the site from a remote computer. Later I saw the site from a PC outside the property. I learned that the only way I could maintain a WordPress site on the home web server is to access the Dashboard control panel from outside the home area network. Accessing it from within the LAN messes up the WordPress configuration for it will set the WordPress installation to the IP address of the Network rather than the domain name I got from dyndns.com.

In order to improve the server performance, I installed Alternative PHP Cache following a tutorial on <http://www.ivankristianto.com/web-development/server/howto-install-alternative-php-cache-apc/1686/>

After installation, I restarted the server with the following command:

```
sudo /etc/init.d/apache2 restart
```

and then viewed <http://192.168.1.9/info.php> and saw that APC was indeed enabled.

The entire experience was a learning one for me. I now understand a bit better how a firewall works. I can now do things that I had only heard of doing before, things such as logging on remotely with ssh from both a command line and using an FTP program and the SFTP protocol using port 22 in order to copy files from one PC to another.

An external link about this subject: [How To Host Your Website From Home \(A Step By Step Guide\)](#)