AfNOG 2011

Dar es Salaam, Tanzania – May 2011

Apache 2.2 with SSL, PHP, Mysql and Wordpress Exercises

1. Intro:

The packages to be installed are:

- MySQL 4.1 ------> *(This may already be installed from the RADIUS exercises)*
- Apache 2.2
- PHP 5

2. Mysql41 Server Installation

```
$ cd /usr/ports/databases/mysql41-server/
$ make install clean
```

Edit /etc/rc.conf for MySQL to start add the line

```
mysql_enable="YES"
```

Start MySQL

```
$ /usr/local/etc/rc.d/mysql-server start
```

Create root password

```
$ mysqladmin –u root password newpassword
```

The "newpassword" is the password of your choice e.g afnog11

3. Apache22 Installation:

Install apache

```
$ cd /usr/ports/www/apache22/
$ make install clean
```

In /etc/rc.conf, add the following line

```
apache22_enable="YES"
```

To start apache run

```
$ /usr/local/etc/rc.d/apache22 start
```

Check if the apache web server you have just installed works by pointing a browser to the server i.e. *http://<your-ip-address>*.

4. Configuring Virtual Hosts

Edit the httpd.conf file:

```
$ vi /usr/local/etc/apache22/httpd.conf
```

Uncomment the line below in the apache configuration file:

```
#include etc/apache22/extra/httpd-vhosts.conf
```

Ensure that Apache listens on port 80

```
# Listen for virtual host requests on all IP addresses
# (both IPv4 and IPv6)
Listen 80
```

Ensure that Apache server name is set not to bind to any DNS name listens on port 80

```
# To avoid binding to DNS Names or IP
ServerName *:80
```

Edit **/usr/local/etc/apache22/extra/httpd-vhosts.conf** to define the virtual hosts. In the exercise "name" should be replaced with your name as in the DNS exercise.

```
NameVirtualHost *:80

<VirtualHost *:80>
   ServerAdmin webmaster@<name>.afnogws.gh
   DocumentRoot /home/afnog/<name>
   ServerName www.<name>.afnogws.gh
   <Directory /home/afnog/name>
       Order deny,allow
       Allow from all
   </Directory>
</VirtualHost>
```

Create the directory "name" in the **/home/afnog** directory and give it the right permissions for apache to access it – in this case

```
$ cd /home/afnog/
$ mkdir <name>
$ chown –R www:www <name>
$ chmod u+x <name>
```

NOTE: By default apaches Directory access permissions are restrictive to deny all. This requires that any directory access for apache outside the Document root should be explicitly set.

To allow from all from httpd.conf may resolve the problem, but is not recommended. The best option is as follows;

Create a file called name.conf (where name is same as "name" above for ease of management) in **/usr/local/etc/apache22/Includes/**

5. Configuring Secure Virtual Hosts (Apache+SSL)

To create a secure virtual host accessed via https rather than http, you will need to configure your Apache server to use OpenSSL for encrypting the data servered from the web server.

The following steps should do the trick.

5.1 Create the SSL Certificates for your Apache Web Server:

```
$ cd /usr/local/etc/apache22/
$ openssl genrsa -des3 -out server.key 1024
```

NOTE: Password-Phrase is needed to encrypt the key**. However,** this pass-phrase will be needed at every apache restart. To get rid of the pass-phrase prompts at every apache restart and maintain the original key.

```
$ cp server.key server.key.org
$ openssl rsa -in server.key.org -out server.key
```

5.1.1 Create Certificate Request

```
$ openssl req -new -key server.key -out server.csr
```

NOTE: The CommonName is the name of the Website you will use in this case the localhost name i.e www.<name>.afnogws.gh

5.1.2 Self Sign your Own Certificate

```
$ openssl x509 -req -days 3650 -in server.csr -signkey \
server.key -out server.crt
```

5.2 Enable SSL in Apache

Edit the **`httpd.conf`** file and **uncomment** the line below;

```
#include etc/apache22/extra/httpd-ssl.conf
```

Edit the `httpd-ssl.conf` file and make the following changes:

```
$ vi /usr/local/etc/apache22/extra/httpd-ssl.conf
```

NOTE:

- Each virtual host must have its own certificate file see comments on “CommonName”.
- The path is where the certificate File and Keys are located in this case /usr/loca/etc/apache22/ (see virtualhost example below)

```
SSLCertificateFile    /usr/local/etc/apache22/server.crt
SSLCertificateKeyFile /usr/local/etc/apache22/server.key
```

5.3 Create the Secure Virtual Host

At the end of the file **(httpd-ssl.conf)** add the virtual hosts that will be handled with SSL

```
<VirtualHost www.<name>.afnogws.gh:443>
 ServerAdmin webmaster@<name>.afnogws.gh
 DocumentRoot /home/afnog/<name>
 ServerName www.<name>.afnogws.gh
 ErrorLog "/var/log/<name>.afnogws.gh-error.log"
 TransferLog "/var/log/<name>.afnogws.gh-access.log"
 SSLEngine on
 SSLCertificateFile /usr/local/etc/apache22/server.crt
 SSLCertificateKeyFile /usr/local/etc/apache22/server.key
 <FilesMatch "\.(cgi|shtml|phtml|php)$">
     SSLOptions +StdEnvVars
 </FilesMatch>
 <Directory "/home/afnog/<name>">
     Options -Indexes FollowSymLinks
     AllowOverride AuthConfig FileInfo
     Order allow,deny
     Allow from all
 </Directory>
 <Directory "/usr/local/www/apache22/cgi-bin">
     SSLOptions +StdEnvVars
 </Directory>
      BrowserMatch ".*MSIE.*" \
               nokeepalive ssl-unclean-shutdown \
               downgrade-1.0 force-response-1.0
      CustomLog "/var/log/httpd-ssl_request.log" \
                "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>
```

Create the directory from which you are going to server your web pages (if it does not already exist)

```
$ mkdir /home/afnog/<name>
$ chown –R www:www /home/afnog
```

Copy the default **index.html** file to the directory from which you are going to serve your web pages.

```
$ cp /usr/local/www/apache22/data/index.html /home/afnog/<name>/
```

Restart the Apache

```
$ apachectl restart
```

Check if the apache-ssl virtualhost you have just setup works by pointing a browser to the desired URL i.e. *https://www.<name>.afnogws.gh*.

6. Installing PHP & PHP Extensions

```
$ cd /usr/ports/lang/php5
$ make install clean
```

** During the installation, select Apache Option # “Build Apache Module” option

Once its completes, proceed to add the following into the apache httpd.conf file to enable PHP in apache

```
$ vi /usr/local/etc/apache22/httpd.conf
```

Find directory index as below and add index.php to the end of the line

```
DirectoryIndex index.html index.htm index.php
```

Also find the `Addtype` section and add the 2 lines below

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

Copy the php initialization file that has the php features.

```
$ cp /usr/local/etc/php.ini-development /usr/local/etc/php.ini
```

Install php5-extensions, which provide support for various modules like MySQL and others.

```
$ cd /usr/ports/lang/php5-extenstions
$ make install clean
```

** Select MySQL and IMAP support, once it is complete, restart Apache.

```
$ apachectl restart
```

6.1 Test the PHP installation

Create PHP test page

```
$ vi /home/afnog/<name>/test.php

<?php
  $hostname = gethostbyaddr($_SERVER['REMOTE_ADDR']);
  echo "Your IP Address is $hostname";
  phpinfo();
?>
```

Point your browser to the following URL:
http://www.<name>.afnogws.gh/test.php

7. Install and Configure Wordpress

Install wordpress:

```
$ cd /usr/ports/www/wordpress
$ make install clean
```

Create the database for wordpress:

```
$ mysql –u root –p
Password:
mysql> CREATE DATABASE wordpress;
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO "wpuser"@"localhost"
    -> IDENTIFIED BY "afnog11";
mysql> FLUSH PRIVILEGES;
mysql> EXIT
```

Modify your DNS zone file `<name>.afnogws.gh` to make an entry for your wordpress virtual host

```
wordpress    IN    CNAME        www
```

NOTE: Please do not forget to increase the serial number in your zone file and restart named

Create a virtual host for wordpress in apache

```
<VirtualHost *:80>
    ServerAdmin webmaster@<name>.afnogws.gh
    DocumentRoot "/usr/local/www/wordpress"
    ServerName wordpress.<name>.afnogws.gh
    ErrorLog "/var/log/wordpress.<name>.afnogws.gh-error_log"
    CustomLog "/var/log/wordpress.<name>.afnogws.gh-access_log" common
    <Directory "/usr/local/www/wordpress">
        Options -Indexes FollowSymLinks
        AllowOverride AuthConfig FileInfo
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

Modify the wordpress config in `/usr/local/www/wordpress/`

```
$ cd /usr/local/www/wordpress
$ cp wp-config-sample.php wp-config.php
```

Edit the `wp-config.php` with the database name, username and passwords created above.

Save the wordpress configuration file and restart apache.

Point your browser to your WordPress virtual host **`http://wordpress.<name>.afnogws.gh`**

Happy blogging!!!!!!!!!!!!!!!!!!!!!!!

8. Enable IPv6 in FreeBSD

As user root, edit /etc/rc.conf and add:-

ipv6_enable="YES"

Add (manual) interface configuration:-

ipv6_network_interfaces="em0"

Use the above option to limit the interface to enable IPv6. By default it's "auto", meaning to enable IPv6 on all the interfaces.

ipv6_ifconfig_em0="2001:43f8:0220:219::1XX"

(where x is your pc number, eg.1, 2, 3 …)

Add the default router:-

ipv6_defaultrouter="2001:43f8:0220:219:196:200:219:254"

Reboot!

Test with the configuration changes with "`traceroute6`" and "`ping6`"

- To your neighbor's ipv6 address
- An external host e.g. tracroute6 www.afrinic.net

Open your browser and go to:

- http://www.afrinic.net
- http://[2001:610:240:a50::2]/

or go to http://ipv6.google.com and www.kame.net

Without having to reboot do this after editing the /etc/rc.confg;

```
$ ifconfig em0 inet6 2001:4348:0220:219::1XX prefixlen 64
$ route add –inet6 default 2001:4348:0220:219:196:200:219:254

$ /etc/rc.d/network_ipv6 start
```

(If you have IPv6 AAAA record for **www.<name>.afnogws.gh** the result will be the Ipv6 address or otherwise the IPv4 address.) At your own time you can add the AAAA resource record, change serial number and see the results.