DebianPHP

⚠ **THIS IS A SCRATCHPAD DOCUMENT, PLEASE CONSIDER THIS WHEN READING ON** 💡 (See ScratchPad for a definition of this term.)

Installing Apache 2 and PHP 5 on Debian

Lately a lot of folks on #Apache have requested help with installation of Apache 2[.2] and PHP 5 on Debian-based systems.

Even though this is a rather straightforward process, we have so far neglected to document it. This page provides that documentation.

Preparing your system

Before getting started, you should prepare your system, and before preparing your system, you should prepare yourself. You should know: What do you want to achieve? What technology is best to achieve that goal? What constraints are there?

Now that you know that you want to create a website, using *Apache 2*, *PHP 5* and MySQL, that your knowledge is your only constraint, and that this wiki will help you learn more, you can start preparing your system by typing [either as *root*, or prefixed with *sudo*]:

```
apt-get update
```

This will download the latest lists of packages, to make sure you install the current versions of the software.

Installing the Software

Installing the required packages is as straightforward as installing any other package on Debian; the trick is knowing what packages you need:

```
apt-get install apache2-mpm-prefork libapache2-mod-php5
```

This should actually suffice. Debian's package mangement system should now automatically chose the missing packages which need to be installed. If you're running the latest version of Debian [as of this writing: Etch], you might want to also install the latest [and best] version of apache, by explicitly specifying *apache2.2-common* in the above install command.

If you also need MySQL, adding *php5-mysql* and *mysql-server-5.0*, should do the rest of the magic.

After that install is complete, visiting _http://localhost/_ in your browser should prove that you now have a running web server.

Configuring PHP

As Debian is trying to be extra-friendly, it splits a single package up into all possible and impossible modules. In Apache's case those modules are called *lib apache2-mod-something* and their configuration files are saved in */etc/apache2/mods-available*. To enable a module, you'd type something like:

```
a2enmod php5
```

Which will simply symlink */etc/apache2/mods-available/php5.conf* and */etc/apache2/mods-available/php5.load* to */etc/apache2/mods-enabled/*.

If you had php4 installed you might get the following error when restarting apache:

```
Forcing reload of web server (apache2)...apache2: Syntax error on line 185 of /etc/apache2/apache2.conf: Syntax
error on line 1 of /etc/apache2/mods-enabled/php4.load: Cannot load /usr/lib/apache2/modules/libphp4.so into
server: /usr/lib/apache2/modules/libphp4.so: cannot open shared object file: No such file or directory
 failed!
```

To solve this disable the php4 module:

```
a2dismod php4
```

For debian Lenny (5.0) and lower

You're half done now. The last two things that need to be taken care of are to **correct** Debian's default PHP configuration for Apache. You can do this by editing /etc/apache2/mods-available/php5.conf:

```
Change:

AddType application/x-httpd-php .php

to:

<FilesMatch \.php$>
  SetHandler application/x-httpd-php
</FilesMatch>

and save the changes.
```

And now you'll have to tell Apache to reload its configuration and thus enable PHP:

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

That's it. You should now be able to run PHP scripts on your newly installed web server.

Stale Cached MimeType

A very frequent question on #apache is, "I've just installed PHP but my browser is offering to download my scripts". This could be because Debian's config puts "application/x-httpd-php php" in its mime.types file, but fortunately the fix is easy. Clear your browser's cache and restart the browser, and then try again.

Resources

DistrosDefaultLayout In order to become familiar with your new Debian-based Apache system you should read this. Also you may be surprised that the Debian folks not only make odd changes to the installation, but fortunately document them well. In */usr/share/doc/apache2/* You should find most of the documentation you need.