PHP4AndPHP5

⚠️ _ THIS IS VERY MUCH A SCRATCHPAD DOCUMENT AT THE MOMENT !!_ 💡

💡 _ NOT FOR PUBLIC CONSUMPTION_ 💡

Guide to enabling both PHP4 and PHP5 in different vhosts

We often see people asking how to enable both PHP4 and PHP5 for different vhosts. This can be done if you follow the Apache recommended method of enabling PHP in the first place. There is a very popular misconception that people should use `AddType` to enable PHP. This will not work in this scenario and is not recommended.

We won't tell you how to install PHP, but you need to make sure that you select different install locations for each version. You then need to maintain a separate copy of `php.ini` for each install.

The PHP modules should be loaded by having something similar to this in your `httpd.conf`

```
...
LoadModule php_module         modules/libphp.so
LoadModule php5_module        modules/libphp5.so
...
```

N.B. *You need to check the names of the modules you are using; the names above may not match what you have on your system.*

Once you have added these, you now need to instruct Apache on how to use these modules. The Apache recommended way to do this is to use the `AddHandler` directive. Below is an example of how:

```
AddHandler application/x-httpd-php .php .phtml
Action application/x-httpd-php modules/libphp5.so
```

If you just add this to your main server config, it will use the PHP5 module. This will then apply to all vhosts. All you need to do now is specify which vhost will use which version of PHP:

```
<VirtualHost *:80>
    ServerName www.example.com
    DocumentRoot /usr/local/apache2/htdocs/example1
</VirtualHost>

<VirtualHost *:80>
    ServerName www.example2.com
    DocumentRoot /usr/local/apache2/htdocs/example2

    RemoveHandler .php
    AddHandler application/x-httpd-php .php .phtml
    Action application/x-httpd-php modules/libphp.so
</VirtualHost>

<VirtualHost *:80>
    ServerName www.example3.com
    DocumentRoot /usr/local/apache2/htdocs/example3

    Include /usr/local/apache2/conf/php4.conf
</VirtualHost>
```

You will notice that in vhost 1, we did not specify a version of PHP to use, this it used the system global version. We did this using `RemoveHandler`, which will remove the previous PHP handler directive.

There are several ways to use these options:

1. Set a global default (i.e. PHP5), then use `RemoveHandler` to specify each vhost that needs to use PHP4. 2. Set each each vhost to use a specific version (i.e., not use a global default). You could use an `Include` statement for this, as in vhost example 3; the include would include all the required PHP4 directives.