

Packaging PHP on Andrew Linux

Stephen Hathorne
Unix Systems Administrator
Infrastructure, Systems, Applications and Middleware
Carnegie Mellon
slh+@cmu.edu

This document is a tutorial on how to install PHP4 on an Andrew Linux System.

Assumptions

This document assumes that you have base familiarity with UNIX Systems Administration, that you have administrative privileges, (a root instance), on the system that need to be modified, and that you have passing familiarity with the Andrew Package System.

Conventions

\$> Represents a system prompt do not type this.
Items listed after the \$> prompt are to be typed:

Example:

\$> ls

This example indicates for the reader to type the **ls** command after the \$> prompt.

Items listed with borders and gray shaded represent file examples:

```
%define doesreallycoolstuff
```

Items listed in plain Arial Text 12 point font are explanatory in nature, and will comprise most of the document.

Italicized items represent filenames or special instructions

Bold elements generally represent section headings or important topics

Step by Step Installation

1. Modify */etc/package.proto* to install necessary files
2. Run */etc/mpp-package* to update the system
3. Modify *httpd.conf* so that apache understands what to do with a *.php* file.
4. Create a sample php file and browse to it using a web browsers

I. Modify */etc/package.proto*

/etc/package.proto is a file that is used by the Andrew Package System to update, modify and place files on an Andrew System. In order to have package load the Apache Web Server on an Andrew System we need to add the package define¹

```
%define doespachephp4  
%define hashttpd.conf
```

to */etc/package.proto*

The *%doespache* line that appears in the example should also be in */etc/package.proto*. This will load apache, and the php modules on the local disk.

Sample *package.proto* file:

```
%define doesapache  
%define hashttpd.conf  
%define doesapachephp4  
%define usesdepot  
%define localdepotdir  
%define contribdepotdir  
%define autolinuxspec  
%include /afs/andrew.cmu.edu/wsadmin/public/src/public.proto
```

Step I is now complete.

II. Run */etc/mpp-package*

This will invoke the Andrew Package system to copy, move, and reorient files on the local disk. Based on your */etc/package.proto* file. Using the one above will cause package to load the Apache Web Server, and the PHP4 module.

¹ See Appendix A: Additional Resources for more information on Package

\$>/etc/mpp-package

Step II is now complete

III. Modify /etc/httpd.conf

The file */usr/www/conf/httpd.conf* controls aspects of the Apache Web Server², in our case we wish to be sure that the Web Server understands how to handle files with a **.php** extension, as well as to add and load the **php4** module. The elements that have to be added to the file include:

```
AddModule mod_php4.c
LoadModule php4_module libexec/libphp4.so
AddType application/x-httpd-php .php
DirectoryIndex index.html index.php
php_flag asp_tags on
```

We will discuss each of these statements in the order that we will need to put them in the file. At the end of this section we will see an complete example *httpd.conf*.

AddModule mod_php4.c

This statement indicates to the Apache Web Server that we wish to add the *mod_php4.c* module to our web server instance. In the *httpd.conf* file there is a section that has a number of these lines. I usually put a new *AddModule* statement on the last line of the *AddModule* section.

LoadModule php4_module libexec/libphp4.so

This statement tells Apache that the *php4_module* is invoked by using the file in the *libexec* directory. This file is called *libphp4.so*, the full path based on our model is: */usr/www/libexec/libphp4.so*. Just as in the last line of the *AddModule* section, I put a new *LoadModule* line at the end of the current *LoadModule* section.

AddType application/x-httpd-php .php

This *AddType* line is an indication to the Apache Web Server that files ending with a **.php** extension should be run using the **php** module that we have previously loaded. Once again I place this declaration at the end of any current *AddType* statements within the file.

² See Appendix A: Additional Resources for more information on Apache Configuration

DirectoryIndex index.html index.php

The *DirectoryIndex* statement is a simple directive to Apache that tells it that both index.html, and index.php files may be present, an important note here. If both files exist the first one in the list (in this case index.html will be invoked)

php_flag asp_tags on [optional]

This directive enables the developer to use the *asp format tags for php scripts* to escape html, and separate html content from php programming

```
<?php echo "hello world"; ?> # php type tag
```

```
<% echo "hello world"; %> # asp type tag
```

IV. Create a simple .php file

At this point the Apache Web Server should be in place with the PHP4 module, let's create a simple php file in /usr/www/tree directory (the default DocumentRoot Directory for Andrew Linux). This will allow us to test the Apache php Installation.

Sample test.php

```
<html>
<head>
<title>This is a sample php file</title>
</head>
<body>
<?php
echo "<center><h1>Hello World of PHP</h1></center>";
phpinfo();
?>
</body>
</html>
```

This simple php script puts a bold and centered "Hello World of PHP" on your screen followed by the output of the **phpinfo()** subroutine. This subroutine displays information about the **php** installation.

You reach this by typing:

<http://yourmachinename.andrew.cmu.edu/test.php>

in a browser window. Your results should look similar to the following:

This is a sample php file

http://shiloh.andrew.cmu.edu/test.php

Hello World of PHP

PHP Version 4.2.3

System	Linux sourcefour.andrew.cmu.edu 2.4.22-ac3 #5 SMP-i686+-MP (020) Thu Sep 18 11:55:37 EDT 2003 i686 i686 i386 GNU/Linux
Build Date	Sep 29 2003 23:26:07
Configure Command	'./configure' '--enable-shared=max' '--with-layout=GNU' '--prefix=/usr/php' '--with-apxs=/usr/www/bin/apxs' '--with-config-file-path=/usr/www/conf' '--enable-memory-limit' '--with-xml' '--with-imap=/usr/local' '--with-kerberos=/usr/local' '--with-imap-ssl' '--enable-dba' '--with-db3' '--with-gdbm' '--with-mysql=/usr/local' '--with-ldap=/usr/local' '--with-snmp' '--enable-ucd-snmp-hack' '--with-openssl' '--with-gd=/usr/local' '--with-jpeg-dir=/usr/local/lib' '--with-png-dir=/usr/local/lib' '--with-zlib-dir=/usr/local/lib' '--with-xpm-dir=/usr/local/lib' '--with-freetype-dir=/usr/local' '--with-ttf' '--enable-gd-native-ttf'
Server API	Apache
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/www/conf/php.ini
Debug Build	no
Thread Safety	disabled

This program makes use of the Zend Scripting Language Engine:
Zend Engine v1.2.0, Copyright (c) 1998-2002 Zend Technologies

Powered by
Zend

The above image does not contain all of the information about the php installation, you are likely to have to scroll quite a bit to get all of the information depending on the size of your php installation and the number of database engines and options configured.

Once you have seen the page above or your version from your machine, php installed on your system.

Appendix A—Additional Resources

Additional information on package can be found at:

http://www.cmu.edu/computing/documentation/andrw_package/package.html

Additional information and documentation on Apache can be found at:

<http://acs-wiki.andrew.cmu.edu/twiki/bin/view/Andrewenv/SoftwareHowTos>

<http://www.apache.org>

<http://www.php.net>