Updating PHP
on
Mac OS X Server 10.4.x

```
----------------------------------------------------------------
DISCLAIMER: The author(s) claim(s) no responsibility for any
damage that may occur from the use of any information found here
or found on links followed from this document. The author(s)
will take credit for the good stuff though.

All items and/or companies mentioned on this and other pages are
or may be trademarks, registered trademarks, or service marks of
the respective companies or individuals.
----------------------------------------------------------------
```

1. - Introduction

The purpose of this document is to provide instructions on how to update the version of PHP included with OS X 10.4.x Tiger Server. DO NOT USE this document to update PHP on 10.5.x Leopard Server. An separate tutorial is available.

It will guide you through updating to the latest version of PHP4 without breaking Server Admin functionality. Furthermore you will be shown how to update to the latest version of PHP5, while retaining most of Server Admin's functionality.

Although the Apple included version of PHP4 is suitable for most users, there are situations where one might need to update. A common reason is to include support for the GD2 libraries.

This document will require you to use the command line. If you do not feel comfortable with using the command line, you should look for a ready made installer package.

Why should I use the command line if there are ready made installer packages available? Short answer: Greater flexibility. Long answer: The packages available, reflect the creators' views on what should or shouldn't be included in the package. By understanding how to install things yourself, you can choose what to include in your PHP build.

This document is written for Mac OS X 10.4.x. However, it should apply to 10.3.x as well. Be aware though that I have not done any particular testing of this procedure on 10.3.x.

DISCLAIMER: Whatever you do based on this document, you do it at your own risk! Just in case you haven't understood: Whatever you do based on this document, you do it at your own risk!

This tutorial has been tested on a standard Mac OS X 10.4.x Server. If you have already tinkered with your system, be aware that things might differ. It is impossible for me to foresee all changes that one might have applied to a server.

This tutorial contains step-by-step instructions for the terminal. Although you could just type them in line by line, it is recommended you have a basic understanding of the terminal.

2. - What is installed as part of OS X Server

In order to better understand OS X 10.4.x the process it is useful to have a basic knowledge of the basic installation.

As of 10.4.8 and the latest security updates, Apple includes PHP 4.4.4 with its server operating system. The configuration used is as follows:

```
./configure --prefix=/usr --mandir=/usr/share/man --
infodir=/usr/share/info --disable-dependency-tracking --
with-apxs --with-ldap=/usr --with-kerberos=/usr --
enable-cli --with-zlib-dir=/usr --enable-trans-sid --
with-xml --enable-exif --enable-ftp --enable-mbstring --
enable-mbregex --enable-dbx --enable-sockets --with-
iodbc=/usr --with-curl=/usr --with-config-file-path=/
private/etc --sysconfdir=/private/etc --with-mysql=/usr
--with-mysql-sock=/var/mysql/mysql.sock --without-pear
```

Looks scary? Don't worry, it will become clearer later on. This line basically tells PHP what modules to install and what to omit. A sample of options can be found here:
http://ch2.php.net/manual/en/configure.php
For those who can't wait and know what they do: to configure a freshly downloaded PHP4 version to behave exactly as Apple intended it to, just use above configuration command.

3. - Requirements

Before you get started, you need to make sure some basic requirements are met:

- You have made a backup of your system.
- You have the latest version of Apple's Developer Tools (Xcode 2.4 or higher) installed.
  Dev Tools are available on your Server DVD and as a free download from Apple's Developer Connection.
- X11 SDK is installed (available on your OS X Developer Tools Disc or Image. This is different from the X11 client that comes with OS X.)
- You do have a backup
- You are running 10.4.x
- You have not manually updated anything related to PHP or Apache so far (if you have, you must know how to adapt these instructions to the changes you made).
- You have a PPC processor

- Not a requirement, but it is recommended you subscribe to our newsletter(s) or follow us on Twitter to be informed when updated versions of this tutorial become available:
  http://osx.topicdesk.com/newsletter/
  http://twitter.com/topicdesk/

4. - Getting and installing the latest version of PHP4

This chapter will guide you through replacing your current version of PHP4 with the latest version available. It will not add any functionality compared to the Apple installed version. It will only give you the peace of mind to have the latest version with the most recent security fixes. If you need extra functionality, wait for the following chapters.

So let's get going:
Make sure you are logged in as root.

Get and install the latest version of PHP by issuing the following commands (*in oblique type*). Issue them one after the other making sure you do not miss any dots or slashes. Also note that the download URL given will change in the future. In that case just replace the URL in this document with the current one. The URL used at the time this document was written refers to PHP 4.4.9. If you want to install a different version, adjust accordingly. Lines wrapping without line spacing are a single command.

```
gcc_select 3.3
```

(for PPC. Use "*gcc_select 4.0*" for Intel)

```
mkdir -p /SourceCache

cd /SourceCache

curl -O http://us3.php.net/distributions/
php-4.4.9.tar.gz

tar xzpf php-4.4.9.tar.gz

cd /SourceCache/php-4.4.9

sh

CFLAGS=-DBIND_8_COMPAT

export CFLAGS
```

(The following is one long line starting with './configure' and ending with 'pear')

```
./configure --prefix=/usr --mandir=/usr/share/man --
infodir=/usr/share/info --disable-dependency-tracking --
with-apxs --with-ldap=/usr --with-kerberos=/usr --
enable-cli --with-zlib-dir=/usr --with-xml --enable-exif
--enable-ftp --enable-mbstring --enable-mbregex --
enable-dbx --enable-sockets --with-iodbc=/usr --with-
curl=/usr --with-config-file-path=/private/etc --
sysconfdir=/private/etc --with-mysql=/usr --with-mysql-
sock=/var/mysql/mysql.sock --without-pear
```

(The above is one long line starting with './configure' and ending with 'pear')

```
make
```

(Up to this point, you have only built PHP4. Your current version has not been replaced. The next command will take care of this).

```
make install
```

```
sudo apachectl graceful restart
```

```
exit
```

You should now be all set and have an "Apple compliant" version of PHP4. Server Admin will still work as before.

/usr/bin/php -v will tell you the version.
/usr/bin/php -i will give you more info.

A more comfortable way of seeing the details of your PHP configuration can be achieved through a special .php file. Create a file called info.php with the following contents:

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

When done, place it in an accessible directory of your web-server and call it through your browser. Detailed version and configuration information will be displayed.

5. - Adding/removing features to/from PHP

Besides simply replacing PHP with the latest version, one can at the same time choose additional options.

The basic procedure is the same as outlined in chapter 4. The main difference is the addition/removal of configuration option. A configuration option is what follows the ./configure command. For example *--with-config-file-path=/private/etc* will tell PHP4 to look for its configuration file inside "*--with-config-file-path=/private/etc*".

Some options come as part of the PHP distribution and need only to be added/removed. Others like "*--with-gd*" for example rely on external components to be installed on your system. Simply adding the option to ./configure will not be enough. In this particular case you would tell PHP to add support for the GD Graphics Library which you would need to install first. A separate tutorial on installing the GD Graphics Library is available on http://osx.topicdesk.com/.

Let's take this as an example then. Once you have installed GD (following the other tutorial), you would repeat chapter 4 with one notable exception: The ./configure statement needs to be changed. Everything else is exactly the same.

(The following is one long line starting with './configure' and ending with 'gd')

```
./configure --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --disable-dependency-tracking --with-apxs --with-ldap=/usr --with-kerberos=/usr --enable-cli --with-zlib-dir=/usr --with-xml --enable-exif --enable-ftp --enable-mbstring --enable-mbregex --enable-dbx --enable-sockets --with-iodbc=/usr --with-curl=/usr --with-config-file-path=/private/etc --sysconfdir=/private/etc --with-mysql=/usr --with-mysql-sock=/var/mysql/mysql.sock --without-pear --with-jpeg-dir=/usr/local --with-png-dir=/usr/local --with-freetype-dir=/usr/X11R6 --with-gd=/usr/local
```

(The above is one long line starting with './configure' and ending with 'gd')

If you look at above ./configure statement we have added "--with-jpeg-dir=/usr/local --with-png-dir=/usr/local --with-freetype-dir=*/usr/X11R6* --with-gd=/usr/local". This tells PHP to use GD and where to look for the libraries required by GD.

Another possibility might be to remove the *--without-pear* option. This will tell PHP to keep using pear (for some reason Apple has removed Pear support in its latest security update).

As you might have guessed by now, it is quite simple to tailor PHP to your specific needs.

6. - Getting and installing the latest version of PHP5

Instead of PHP4 you may want to use PHP5. Be it because some software you need relies on it, be it because you can. If you do not a specific need, it is recommended you keep PHP4.

The basic steps required to install PHP5 are the same as for PHP5. Because PHP5 cannot be integrated 100% with Server Admin and may also create some incompatiblities (SquirrelMail for example would have to be updated to run with PHP5), we will install it alongside PHP4 rather than replace PHP4. Be aware though that only one version of PHP can be running at the same time!

So let's get going:
Make sure you are logged in as root.

Get and install the latest version of PHP5 by issuing the following commands (*in oblique type*). Issue them one after the other making sure you do not miss any dots or slashes. Also note that the download URL given will change in the future. In that case just change the URL with the current one. The URL used at the time this document was written refers to PHP 5.2.10. If you want to install a different version, adjust accordingly. Lines wrapping without line spacing are a single command.

gcc_select 3.3
(for PPC. Use "*gcc_select 4.0*" for Intel)

```
mkdir -p /SourceCache

cd /SourceCache

curl -O http://us3.php.net/distributions/php-5.2.10.tar.gz

tar xzpf php-5.2.10.tar.gz

cd /SourceCache/php-5.2.10

sh

CFLAGS=-DBIND_8_COMPAT

export CFLAGS
```

(The following is one long line starting with './configure' and ending with 'sock')

```
./configure --prefix=/usr/local/php5 --mandir=/usr/share/man --infodir=/usr/share/info --with-apxs --with-ldap=/usr --with-kerberos=/usr --enable-cli --with-zlib-dir=/usr --with-libxml-dir=/usr --enable-exif --enable-ftp --enable-mbstring --enable-sockets --with-iodbc=/usr --with-curl=/usr --with-config-file-path=/private/etc --with-mysql=/usr --with-mysql-sock=/var/mysql/mysql.sock
```

(The above is one long line starting with './configure' and ending with 'sock')

```
make
```

(Up to this point, you have only built PHP5. It has not been installed. The next command will take care of this).

```
make install

sudo apachectl graceful restart

exit
```

You should now be all set and have PHP 5 alongside of PHP4. Server Admin will still work as before.

/usr/local/php5/bin/php -v will tell you the version
/usr/local/php5/bin/php -i will give you more info.

A more comfortable way of seeing the details of your PHP configuration can be achieved through a special .php file. Create a file called info.php with the following contents:

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

When done, place it in an accessible directory of your web-server and call it through your browser. Detailed version and configuration information will be displayed.

As I mentioned before, only one version of PHP can be running at the same time. Some actions in Server Admin (like turning on webmail), will revert back to PHP4 as the running version.

Your best bet is to duplicate /etc/httpd/httpd.conf and create one version for PHP4 and one for PHP5. This will allow you to easily change/revert versions.

To this purpose issue:

```
sudo cp /etc/httpd/httpd.conf /etc/httpd/httpd.php4.conf
sudo cp /etc/httpd/httpd.conf /etc/httpd/httpd.php5.conf
```

Now edit /etc/httpd/httpd.php4.conf and make sure it contains

```
AddModule  mod_php4.c
LoadModule php4_module        libexec/httpd/libphp4.so
```

and that those lines are uncommented.

Next edit /etc/httpd/httpd.php5.conf and make sure it contains

```
AddModule  mod_php5.c
LoadModule php5_module        libexec/httpd/libphp5.so
```

and that those lines are uncommented.

Obviously you also need to make sure that each file only contains its respective reference to PHP4 or PHP5.

Now when you want to change version just issue:

```
sudo cp /etc/httpd/httpd.php5.conf /etc/httpd/httpd.conf
```

or

```
sudo cp /etc/httpd/httpd.php4.conf /etc/httpd/httpd.conf
```

(depending which version you want)
Then issue:

```
sudo apachectl graceful restart
```

7. - Caveats

- MySQL Passwords

Depending on which version of PHP you had originally installed, you may notice an incompatibility with mySQL authentication.

To solve it, log into MySQL at the command line by issuing:

mysql -u root -p

When asked enter your password (SQL root password)

When at the prompt

```
"mysql>"
SET PASSWORD FOR 'youruser'@'yourserver' =
OLD_PASSWORD('yourpassword');
```

(youruser, yourserver (or localhost) and yourpassword must reflect your configuration.)

- php.ini

You may have been using a php.ini file. Certain options inside php.ini may or may not cause issues with a later version you install. Make sure you double check it if something doesn't work.

- PHP4 and PHP5 can be installed but not used together.

See explanation at the end of chapter 6.

- Intel

The plain vanilla build will work for Intel Macs as well. However if you add options that rely on external libraries, you will need to make sure that those libraries are built for Intel as well.

That's all folks.
Hope this helps.
Have fun,
Alex

Doc. Version 1.3.3, 30.6.2009
Athanasios Alexandrides
Lugano, Switzerland
tutorials -at- topicdesk.com
