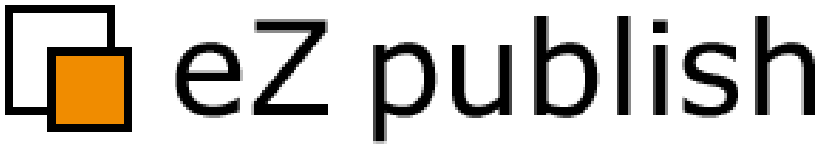
eZ publish Installation Guide



28th February 2001

Contents

Chapter 1

Introduction

> **“He who asks is a fool for five minutes, but he who does not ask remains a fool forever.”**
> -

eZ publish is a content management system, among a lot of other things. This installation manual will try to cover the job of installing eZ publish on your server.

This manual covers installation on a Red Hat Linux system; most of what is described here can also be applied to other installations, especially if your system uses RPM for installation. For other systems you would need to do a lot of compiling yourself to make this work, or apply the system’s own package manager.

Finding packages can be done dirctly from vendor sites, though you might not be guaranteed that you’ll find the package you need. In such instances you need to download the source directly from the software developer.

Different distribution sites for different Unix systems are:

- Debian http://www.debian.org/distrib/ftplist
- IRIX http://freeware.sgi.com/
- Red Hat Linux http://www.redhat.com/apps/download
- SuSE Linux http://www.suse.com/us/support/download/index.html
- Sun http://www.sunfreeware.com/

The addresses to the software developers will be given where apropriate in the text.

A line starting with a hash-sign “#” are input from the user to the shell.

Chapter 2

Pre-requisites

2.1 Needed Privileges

For the standard installation (and for the moment the only method) of eZ publish you will need to have the following privileges on your system:

- Access to Apache's httpd.conf
- Access to compiler
- Access to Apache's modules
- Access to a MySQL database
- You might also need the privilege to add new libraries to your system.

You might also use other web servers than apache, but then you're on your own since we haven't tested eZ publish on other configurations. If you do try another web server, please keep a log of what you do and submit it to us (pkej@ez.no) for inclusion in future versions of this manual.

2.2 Needed Software

You also need to download and install the following packages, if they aren't present on your system already:

- MySQL (http://www.mysql.com) version 3.23 or later.
- libXml (http://xmlsoft.org/#Downloads) version 2.2.7
- libQdom () version
- ImageMagick (http://www.imagemagick.org/) newest version
- Apache (http://httpd.apache.org/) latest 1.3 release.
- Any and all modules you need for apache in addition to mod_php. (http://modules.apache.org/)
- PHP (http://www.php.net/) version 4.0.4pl1 or later, you need the source code version.
- eZ publish (http://developer.ez.no/) verision 2.0 or later stable releases.

The libraries and php will appear pre-compiled for Linux i386 on http://developer.ez.no in the future. The software is listed in the order of installation.

2.3 Which Software is Already Installed?

2.3.1 Systems Using RPM

RPM is a system for distributing pre-compiled software. The packages also contain pre-configured settings and initialisation files, leaving almost nothing to the user, except deciding what to install.

To check if a package is available on your system you can run the following command (RPM based systems "rpm -qa | grep <name of program/library>". If you need to know where you can find the different files from that package you can follow up on the previous command with the following "rpm -ql <rpm name>". RPM name is one of the returned names from the previous command, example:

```
# pkej@vogol:/etc/httpd > rpm -qa | grep libxml
libxml-1.8.7-80
libxmld-1.8.7-80
# pkej@vogol:/etc/httpd > rpm -ql libxml-1.8.7-80
/usr/bin/xml-config
/usr/lib/libxml.so.1
/usr/lib/libxml.so.1.8.7
/usr/share/doc/packages/libxml
/usr/share/doc/packages/libxml/AUTHORS
/usr/share/doc/packages/libxml/COPYING
/usr/share/doc/packages/libxml/COPYING.LIB
/usr/share/doc/packages/libxml/NEWS
/usr/share/doc/packages/libxml/README
/usr/share/doc/packages/libxml/TODO
```

2.4 FreeBSD

When installing and compiling PHP on a FreeBSD system you might encounter an error when using –with-dom which says you have a conifgure error on the lib. It turns out that the current port of libxml installs itself as /usr/local/lib/libxml2.a|so and it goes unrecognised by configure. You can easily get around this problem by linking the libs to libxml.a|so.

2.5 IRIX

By accessing the software manager (you must be root) you can get a list of installed software, scroll or search that list to find the packages you're interested in. Double click on the tabs to the left to get information about where specific files are installed.

2.6 RAQ 3

There is a separate chapter at the end of this manual describing installation on a RAQ 3 server. It was kindly provided by Chris Mason,

2.7 Other Systems

On other systems you should read the documentation for that system to learn how to find out what software is already installed.

You could try to use the command “find” to find the software. It is used thus: “find . -name *<program name>*” from the /usr/, /local/ , /lib/, /share/ directories. In extreme cases you could try from the root of the system, but this will take a long time and will also hog resources on your computer. Therefore we urge you to learn how to use the proper installation features of your system to find the software already installed.

2.8 Installation of Required Software

If you’ve found pre-compiled versions of all the software packaged for use with an installation tool, you just have to install that software using the tool. Instructioins for its usage is often found using the command “man <installation tool name>” or by reading your system’s documentation or the supplier’s website.

If you’ve had to download source code you will find instructions on how to compile and install the software you’ve downloaded at the software developer’s website. This requires a bit of knowledge and you should only undertake this if you feel confident about the job.

This manual will only cover configuration of the software needed and compilation of PHP to use the other software.

Chapter 3

Compile Configuration

3.1 PHP

3.1.1 Unpacking

After you have downloaded PHP you need to unpack it somewhere where you can compile and configure the software. To unpack run the command:

```
# tar zxvf php-4.0.x.tar.gz
```

Where the x is the version of php you've downloaded. Then you need to move into the directory you extracted php into:

```
# cd php-4.0.x
```

3.1.2 Configuration

You'll need either an apache module or a command line version of PHP to use eZ publish on your website. We recommend you use PHP as an apache module. You will also need the command line version if you want to use the cron jobs for periodical updates of the eZ news feed module.

Thus for our recommended installation of PHP you need both the command line and module versions of PHP.

3.1.2.1 Common

Both the command line and apache module versions need to have the following configurations added to the configuration tool:

–enable-trans-sid This lets PHP use session id's which don't rely on cookies. It does not disable normal cookie based sessions.
(http://www.php.net/manual/en/install.configure.php#install.configure.enable-trans-sid)

–with-mysql This tells PHP that the mysql functionality should be used.
(http://www.php.net/manual/en/install.configure.php#install.configure.with-mysql)

–enable-magic-quotes This tells PHP to enable magic quotes by default. you can also turn this feature on and off on a directory by directory basis in either the ".htaccess" files (if you use them) or in the setup of the virtual server in "httpd.conf".
(http://www.php.net/manual/en/install.configure.php#install.configure.enable-magic-quotes)

–with-dom This configures PHP to include libxml.
(http://www.php.net/manual/en/install.configure.php#install.configure.with-dom)

–with-qtdom This configures PHP to include libqdom. It isn't up on the PHP site with a link, but it works as –with-dom.

You should also go through the web page: http://www.php.net/manual/en/install.configure.php and make sure that there isn't other functionality you would like to have included.

3.1.2.2 Command Line

The default is to create a command line version of PHP. Therefore you don't need to add more configuration options for this.

3.1.2.3 Apache Module

To build an apache module you need to add:

–with-apxs This compiles PHP as an apache module.
(http://www.php.net/manual/en/install.configure.php#install.configure.with-apxs)

3.1.2.4 Other Web Servers

We haven't tested our software with other web servers than apache. If you need to try out other web servers, read this document http://www.php.net/manual/en/install.configure.php#install.configure.servers to learn how you configure for the web server you will be using.

3.1.2.5 Creating the Configuration

Now you just have to run the "./configure" program with the apropriate configuration directives which we discussed in the preceeding sections, for an apache module you'd do the following:

```
   # ./configure --enable-trans-sid --with-mysql --with-magic-quotes
--with-apxs --with-dom --with-qtdom
```

Remember that to compile a script/cgi version you'd need to change that line to:

```
    # ./configure --enable-trans-sid --with-mysql --with-magic-quotes
--with-dom --with-qtdom
```

3.1.3 Compilation

To compile you need to run the command "make":

```
    # make
```

3.1.4 Installation

To install your new PHP package you need to run the following command:

```
    # make install
```

Chapter 4

Apache Configuration

For the moment we have only one solution for configuring apache.

4.1 Dual Virtual Host

4.1.1 Configuring Through httpd.conf

This set up is based on having two different virtual hosts for your administration back-end and the main site. The main site would typically be known as “www.yoursite.com” and the administration would be “admin.yoursite.com”; the names are up to you, theoretically you could have different names, for example “mysite.yoursite.com” and “administration.mysite.com”.

The virtual host is configured through the “httpd.conf” file which is the main configuration of Apache. Following is an example of such a host, remember to exchange everything within brackets (“[“ and “]”) with your preferred and local settings and also remove the brackets.

4.1.1.1 User Side

```
# User site
<VirtualHost [your.domain.com]>
  <Directory [/your/apache/documentroot/publish_dist]>
    Options FollowSymLinks Indexes ExecCGI
    AllowOverride None
  </Directory>
  RewriteEngine On
  RewriteRule ^/filemanager/filedownload/([^/]+)/(.*)$
  [/your/apache/documentroot]/publish_dist/ezfilemanager/files/$1
  [T="application/oct-stream",S=1]
  # The lines above should appear on the same
  # line in your configuration file!
  RewriteRule !\.(gif|css|jpg|png)$ [/your/apache/documentroot]/publish_dist/index.php
  ServerAdmin ce@ez.no
  DocumentRoot [/your/apache/documentroot]/publish_dist
  ServerName [your.domain.com]
  ServerAlias publish
</VirtualHost>
```

4.1.1.2 Admin Side

```
# Admin site
```

```
<VirtualHost admin.yourdomain.org>
  <Directory [/your/apache/documentroot/admin]>
    Options FollowSymLinks Indexes ExecCGI
    AllowOverride None
  </Directory>
  RewriteEngine On
  RewriteRule !\.(gif|css|jpg|png)$ [/your/apache/documentroot/admin/index.php]
  ServerAdmin [your_mail@domain.no]
  DocumentRoot [/your/apache/documentroot/admin]
  ServerName [admin.yourdomain.org]
  ServerAlias [admin.yourdomain.org]
</VirtualHost>
```

The format of the "httpd.conf" file is covered at http://httpd.apache.org/docs/ for a complete understanding of the above information you'll need to read that documentation.

The rewrite rules does the following:

```
RewriteRule ^/filemanager/filedownload/([^/]+)/(.*)$
[/your/apache/documentroot]/publish_dist/ezfilemanager/files/$1
[T="application/oct-stream",S=1]
```

This says that everything served from "/filemanager/filedownload/" should be redirected to fetch information from "publish_dist/ezfilemanager/files". In other words, when people downloads a file from the filemanager, the file is served from the directory specified in the second part.

The "^" just after "RewriteRule" says that evertything which starts with this, in other words it is a start of line marker. When working with an URL that is from the root of your site, ie. the part from the first slash after your domain name.

The "$" sign is used to mark the end of line, in order to remember the full line.

The part `"[T="application/oct-stream",S=1]"` means that everything which is matched shall be of the specific mime type ("application/oct-stream", ie. binary download). The "S=1" part means that if we match this rule, we should skip one rule ahead before trying to match again.

The next rewrite rule:

```
RewriteRule !\.(gif|css|jpg|png)$ [/your/apache/documentroot]/publish_dist/index.php
```

is found in both sites (admin and user). This means that every file, except gif, css, jpg and png (and files matched against the previous rule when in the user site) should be redirected to the file in the second part, ie. the index.php file. The reason for this is that we don't want anyone trying to get direct access to anything which might be sensitive, or revealing about the site's operation.

If you didn't compile PHP with magic quotes; or other software relies on PHP not using magic quotes you can add the following line into each virtual host section:

```
php_value magic_quotes_gpc 1
```

NOTE: It isn't possible to use the form http://mysite.com/admin at all; since the admin module assumes that the url "/" is the start of the admin pages. If you do change eZ publish code in order to do this anyway; please send the code to bf@ez.no for future inclusion.

4.1.2 Configuring Through .htaccess

Instead of using httpd.conf and rewrite rules in the virtual hosts, you can also do the rewrite rules in the .htaccess filesm directory specific configuration files.

Note: You must set up apache to accept this. You still need two domains for this operation!

4.1.2.1 User Side

In your main directory (/path/to/index.php/) create a file called ".htaccess" containing the following text:

```
RewriteEngine On
RewriteRule ^/filemanager/filedownload/([^/]+)/(.*)$
/path/to/website/ezfilemanager/files/$1
[T="application/oct-stream",S=1]
# The lines above should appear on the same
# line in your configuration file!
RewriteRule !\.(gif|css|jpg|png)$ /path/to/website/index.php
```

4.1.2.2 Admin Side

In your admin subdomain home directory, create a file with the following text:

```
RewriteEngine On
RewriteRule !\.(gif|css|jpg|png)$ /path/to/website/admin/index.php
```

Note the extra "\" in the rewrite rule. Its slightly different to the line used in the httpd.conf method.

Chapter 5

eZ publish Installation

5.1 Database

Now you need to create a database in MySQL, the default name we use is publish, but you can change that to whatever pleases you.

```
# mysql -uroot -p publish < sql/publish.sql
```

Add a publish user in MySQL. To add a user you can use the MySQL client to log on to mysql and then create the user:

```
# mysql>grant all on publish.* to publish@localhost
identified by "secret";
```

where secret is your password. Then you need to add the default eZ publish data into your newly created database:

```
# mysql -uroot -p publish < sql/publish.sql
```

5.2 Program Files

The next step is to install the eZ publish package in your document root directory. First you need to unpack the software in a temporary directory:

```
# cd /tmp
# tar zxvf /path/to/ezpublish-2.0.tar.gz
```

The next step is to move the files to your document root:

```
# mv /tmp/publish_dist /your/apache/documentroot
```

When all this is done you need to tell eZ publish a little about the site you're running. You'll need to edit the "site.ini" file which you will find in the document root:

```
# cd /your/apache/documentroot
# vi site.ini
```

Instead of vi you can use your preferred text editor. You'll need to add information about the username, hostname and password of your database. More information on what you can do with "site.ini" can be found in the "eZ publish Customisation Guide".

The next important step is to run the script modfix. This script will create symbolic links needed and set permissions.

```
# ./modfix.sh
```

Chapter 6

Now What?

After installing eZ publish you can test your site through the URL http://www.yoursite.com/ and you can administrate your site from the URL http://admin.yoursite.com/, of course, if you did anything different the names of the admin and the public site might be different.

NOTE: The default user name and password for your site will be admin/publish. Remember to change the password.

The next manual you should read is the “eZ publish Customisation Guide”, it tells you how to configure the software to use the functionality you want, as well as how you change the templates to suit your needs.

When you’re finished with the design and the initial testing you can head over to http://zez.org/ for articles about community building as well as programming, or you can visit http://developer.ez.no for updates, articles about eZ publish and how to work with it, as well as keeping abreast of new developments.

Chapter 7

Installing on RAQ 3

Installing ezPublish on raq3 without messing up the GUI or voiding the warranty.

First, add the domain into the DNS, but do not create a virtual site.

Log in by telnet (install SSH unless you are desperate to get hacked).

Put the publish files in the directory you want to use, I used /home/sites/extrasites/mysite/web

Install mysql 3.23 or later by rpm, there is one out there. MySQL (http://www.mysql.com) version 3.23 or later if you want to compile

Now you need to create a database in MySQL, the default name we use is publish, but you can change that to whatever pleases you.

```
# mysql -uroot -p publish < sql/publish.sql
```

Add a publish user in MySQL. To add a user you can use the MySQL client to log on to mysql and then create the user:

```
# mysql>grant all on publish.* to publish@localhost
identified by "secret";
```

where secret is your password. Then you need to add the default eZ publish data into your newly created database:

```
# mysql -uroot -p publish < sql/publish.sql
```

Then get:

- http://www.freesoftware.com/pub/infozip/zlib/ (zlib.tar.gz)
- http://www.boutell.com/gd (gd-1.8.4.tar.gz)
- ftp://ftp.uu.net/graphics/jpeg/jpegsrc.v6b.tar.gz (jpegsrc.v6b.tar.gz)
- http://www.php.net (php-4.0.4pl1.tar.gz)

Delete all gd.h files on your system. You can find them using:

```
# find / -name gd.h
```

If there are more than one, then delete all of them.

Now add the following line to the /etc/ld.so.conf file:

```
/usr/local/lib
```

Save the file, and run:

```
# /sbin/ldconfig
```

This was an important part, because Apache needs this dir to find the correct modules.
Extract the zlib archive:

```
# tar -zxvf zlib.tar.gz # cd zlib-1.1.3
```

And install it:

```
# ./configure --shared
# make
# make install
```

Now install the JPEG-6b, doing the following:

```
# tar -zxvf jpegsrc.v6b.tar.gz
# cd jpeg-6b
# ./configure --enable-shared
# make
# make install
```

Install the PNG library

```
# wget http://www.libpng.org/pub/png/src/libpng-1.0.9.tar.gz
```

Then compile the package.
Get Imagemagick ImageMagick (http://www.imagemagick.org/) newest version Download and then:

```
# tar -zxvf Imagemagick-xxx
# cd Imagemagick-xxx
# ./configure
# make
# make install
```

Then go one directory back, and extract the GD archive using:

```
# tar -zxvf gd-xxx
# cd gd-xxx
```

Now edit the Makefile (using vi or pico) and check which modules you want. I removed the Freetype Library (-DHAVE_LIBFREETYPE / -lfreetype). After making the changes save the file and go back to the shell. Now compile GD:

```
# make
# make install
```

If this is giving any errors, just remove the modules you don't have (but don't remove the JPEG lib - we need that one ! :)))
Now go back one dir, and extract PHP4:

```
# tar -zxvf php-4.0.4pl1.tar.gz
# cd php-4.0.4pl1
```

First remove any cache:

```
# rm config.cache
# make clean
#./configure --with-xml --with-mysql \
--with-apxs=/usr/sbin/apxs \
--with-system-regex \
--with-zlib \
--enable-safe-mode \
--with-gdbm \
--enable-sysvsem \
--with-ftp \
--with-config-file-path=/etc/httpd/conf/ \
--enable-magic-quotes \
--with-exec-dir=/usr/sbin/httpd \
--with-dom \
--enable-trans-sid
# make
# make install
```

run /sbin/ldconfig again.
Apache: (Your milage may vary, be wary of paths)
edit /etc/httpd/conf/httpd.conf and add the Loadmodules lines like this:

```
# Extra Modules
LoadModule php_module modules/mod_php.so
LoadModule php3_module modules/libphp3.so
LoadModule perl_module /usr/lib/apache/libperl.so
LoadModule php4_module /usr/lib/apache/libphp4.so
LoadModule php4_module lib/apache/libphp4.so
```

Reconstruction of the complete module list from all available modules
(static and shared ones) to achieve correct module execution order.
[WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE UPDATE THIS, TOO]

```
ClearModuleList
# Extra Modules
AddModule mod_php.c
AddModule mod_php3.c
AddModule mod_perl.c
AddModule mod_php4.c
```

Add the second line below line below the rewrite stuff, above the <Virtualhost> directives.
NameVirtualHost 216.97.67.4 Include /etc/httpd/conf/extrasites.conf <VirtualHost 216.97.67.4>

create this include file and in it put the apache vitual server directives for your site.
For example:

```
# User site
<VirtualHost yourIP>
  ServerName [yourdomain.org]
  ServerAlias [www.yourdomain.org]
  <Directory [/your/site/root/]>
    Options FollowSymLinks Indexes ExecCGI
    AllowOverride None
  </Directory>
  RewriteEngine On
  RewriteRule !\.(gif|css|jpg|png)$ [/your/site/root/index.php]
  ServerAdmin [your_mail@domain.no]
  DocumentRoot [/your/site/root/]
</VirtualHost>

# Admin site
<VirtualHost admin.yourdomain.org>
  <Directory [/your/site/root/admin]>
    Options FollowSymLinks Indexes ExecCGI
    AllowOverride None
  </Directory>
  RewriteEngine On
  RewriteRule !\.(gif|css|jpg|png)$ [/your/site/root/admin/index.php]
  ServerAdmin [your_mail@domain.no]
  DocumentRoot [/your/site/root/admin]
  ServerName [admin.yourdomain.org]
  ServerAlias [admin.yourdomain.org]
</VirtualHost>
```

restart apache:

```
# /etc/rc.d/init.d/httpd stop
```

wait a few seconds then

```
# /etc/rc.d/init.d/httpd start
```

Then chown httpd.httpd * on both the domain and admin.domain directories to get it to work.
If all is well, your site should work.