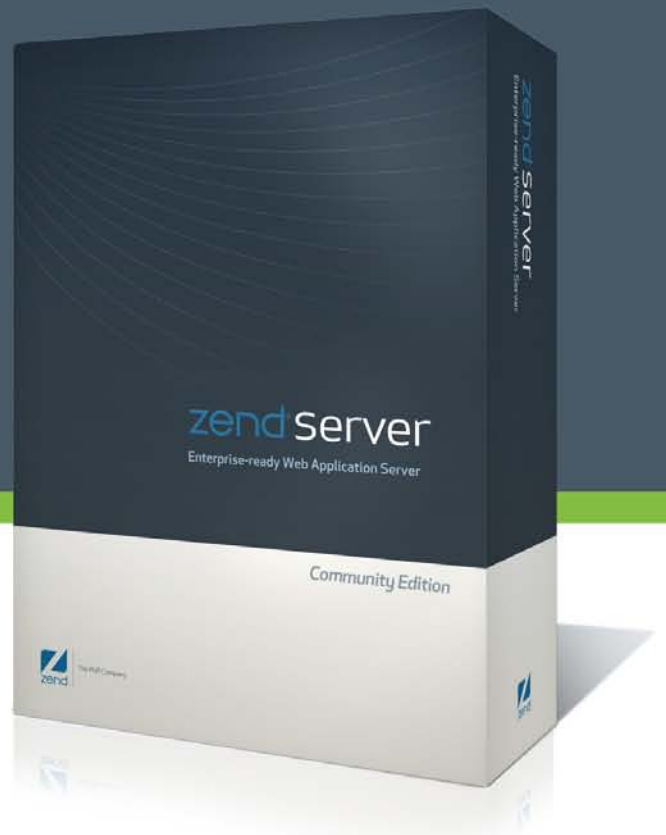




The PHP Company

Zend Server Community Edition 5.6 Installation Guide

By Zend Technologies



Abstract

This is the Installation Guide for Zend Server Version 5.6.

The information in this document is subject to change without notice and does not represent a commitment on the part of Zend Technologies Ltd. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's personal use, without the written permission of Zend Technologies Ltd.

All trademarks mentioned in this document, belong to their respective owners.

© 1999-2012 Zend Technologies Ltd. All rights reserved.

Zend Server Community Edition Installation Guide, issued January 2012.

DN: ZS-IG-030112-5.6-01

Table of Contents

Zend Server Installation Guide.....	1
Installing for the First Time	1
Installation Directories	1
Choosing Which Distribution to Install	2
Supported Operating Systems	3
Tarball Installation	4
Tarball Installation	4
Installing Zend Server Community Edition	4
Uninstalling Zend Server Community Edition.....	6
DEB Installation.....	7
Automatically Installing Zend Server Community Edition.....	7
Manually Installing Zend Server Community Edition.....	8
Additional Packages	9
Post Installation Configuration.....	10
Upgrading Zend Server Community Edition.....	11
Uninstalling Zend Server Community Edition.....	12
RPM Installation (RHEL, CentOS, Fedora and OEL)	13
Automatically Installing Zend Server Community Edition.....	13
Manually Installing Zend Server Community Edition.....	14
Additional Packages	15
Post Installation Configuration.....	16
Enabling RPM Package Signature Validation	17
Upgrading Zend Server Community Edition.....	18
Uninstalling Zend Server Community Edition.....	19
RPM (SLES and OpenSUSE) Installation.....	20
Automatically Installing Zend Server Community Edition.....	20
Manually Installing Zend Server Community Edition.....	21
To set up repositories in SLES 11 and OpenSUSE	21
To set up repositories in SLES 10	22
To install Zend Server Community Edition using zypper.....	23
Basic Usage	23
Additional Packages	23
Upgrading Zend Server Community Edition.....	25
Uninstalling Zend Server Community Edition.....	26
Mac OS X Installation.....	27

Zend Server Community Edition Installation Guide

Installing Zend Server Community Edition	27
Installing Zend Controller.....	28
Post Installation Configuration.....	28
Upgrading Zend Server	29
Uninstalling Zend Server Community Edition.....	29
Windows Installation	30
Installing Zend Server Community Edition	31
Installing with the Native Windows Installer.....	31
Installing Additional Components	32
Installed Components.....	33
Java Bridge	33
Locating Installed Components	33
Start Menu Options.....	33
Upgrading Zend Server	34
Uninstalling Zend Server Community Edition.....	34
PHP Extension List	35
Zend Server Community Edition Extension List - PHP 5.2	36
Common Extensions.....	36
Extra / Additional Extensions	40
Standalone Extensions	41
Zend Server Community Edition Extension List - PHP 5.3	42
Common Extensions.....	42
Extra / Additional Extensions	46
Standalone Extensions	47
Using PECL.....	48
Installing Zend Server Community Edition Additional Extensions.....	48
Uninstalling Zend Server Community Edition Additional Extensions	49
Registering Zend Server Community Edition.....	50
Setup Wizard	50
Upgrading	54
Upgrading	54
Upgrading your RPM Installation to Merge Configuration Files During Upgrade.....	54
Upgrading to a Newer Version of Zend Server Community Edition.....	54
Zend Server	54
Zend Server CE	55
Upgrading from Zend Server CE to Zend Server	56
Upgrading from Zend Server Community Edition CE to Zend Server from within the Product	57

Manual Rollback	59
Upgrading PHP 5.2 to PHP 5.3	59
RPM	59
SLES 11	60
DEB	60
Windows and Mac OS X	60
Migrating to Zend Server Community Edition from Zend Platform/Zend Core	61
Downgrading/Rollback to an Older Version	61
Post Installation	62
Package Setup and Control Scripts	62
Windows: Package Setup and Control Scripts	63
Starting Services on MS Windows	63
Controlling Zend Server Community Edition from Startup	63
Additional Information	64
Linux Mac: Package Setup and Control Scripts	65
Controlling Zend Server Community Edition Components from the Command Line	65
Command Line Actions	65
Controlling Zend Server Community Edition from Startup	66
Log Rotation	67
Configuring Log Rotation for Zend Server Community Edition Logs	67
To Disable Log Rotation	69
Ports and Services	70
Web Server Ports	70
Firewall Settings	70
Zend Server Administration Interface (Web GUI)	71
Debugging and Profiling	71
Zend Java Bridge	72
Installed Components	73
Installation Directories	73
Additional Information	78
Glossary	81

Zend Server Installation Guide

The following instructions describe how to obtain and install Zend Server Community Edition. A summary of the procedure follows and later sections provide the details.

If you plan to upgrade an existing version of Zend Server Community Edition to a newer version rather than install Zend Server Community Edition for the first time, see the section on "[Upgrading Zend Server Community Edition](#)" for information about upgrade procedures and about issues that you should consider before upgrading.

Installing for the First Time

To determine if Zend Server Community Edition is supported on your platform of choice see the list in [Choosing Which Distribution to Install](#)".

Please note that not all platforms are equally suitable for running Zend Server Community Edition.

Note:

If you are installing Zend Server to be added as a node to a cluster using Zend Server Cluster Manager, you must install Zend Server with Session Clustering and the component must be turned on.

Installation Directories

Not all users decide to install their software in the same location. To reflect this actuality, all paths in this document have been replaced with the following prefix: <install_path>. This represents the location of the installed files. If you used the default settings, the location should be as follows:

- Windows: C:\Program Files\Zend\ZendServer
- Windows 64 bit C:\Program Files (x86)\Zend\ZendServer
- DEB/RPM: /usr/local/zend
- Tarball: /usr/local/zend
- Mac: /usr/local/zend
- For Zend Server Community Edition installation directories, see the Zend Server for IBM i Installation Guide.

DN: ZS-IG-011211-5.6-06

Choosing Which Distribution to Install

Zend Server Community Edition is available, in several distribution formats. You can choose from the pre-packaged distribution called [Tarball](#) or install using a Package Manager (RPM and DEB). When in doubt, use the [Tarball](#) distribution.

The distributions for the following product versions are:

1. [DEB](#) and [RPM](#) - Those wanting to use the DEB and RPM should define the Zend Server Community Edition repository (see the DEB and RPM sections for how to define the repository).
2. [Windows](#) - Download the package from zend.com.

Zend Server Community Edition includes the following additional installation options:

1. [Tarball](#) - Download the package from zend.com
2. [DMG](#) - For Mac OSX Download the package from zend.com.

Choose the most suitable type of installation according to your operating system by selecting it from the table below.

If your Operating System does not appear in the list below, click here to be directed to the [Tarball Installation](#). If you are unable to complete the installation, please refer to our Best Practices to see if these were already handled. Only if there is no article on the subject please see the [Zend Support Center](#) for further assistance.

Supported Operating Systems

Package Name	Operating System	Installation Type
Linux	CentOS	RPM
	Debian	DEB
	Fedora	RPM
	Oracle Linux	RPM
	RHEL	RPM
	SUSE	RPM (SLES and Open SUSE)
	Ubuntu	DEB
	All Linux distributions	Tarball
Mac	Mac OS X	DMG
Windows x86 - 32	Windows XP Professional	EXE
	Windows Server	EXE
	Windows Vista*	EXE
Windows x86 - 64	Windows Vista*	EXE
	Windows Server	EXE
	Windows XP Professional	EXE

*All flavors except Home Basic

Tarball Installation

The Tarball installation is a generic Linux Tarball installation that includes Zend Server Community Edition, Apache 2.2, and The latest available version of PHP - PHP 5.2 or PHP 5.3. This method is suitable for all machines, specifically older versions and homegrown Linux-based operating systems that are not specified in the list of supported operating systems.

Requirement: GLIBC >= 2.3.

Note:

This package includes the Java Bridge component. This component requires Sun Microsystems JRE 1.4 (or later) installed on your computer. Therefore, if you do not already have JRE installed, install it before using the Java Bridge. More information about JREs and the latest updates can be found on the Sun Microsystem website: <http://java.sun.com> or http://java.com/en/download/linux_manual.jsp.

Installing Zend Server Community Edition

This procedure describes how to install Zend Server Community Edition using the general Tarball installation.

Before beginning the installation, download the package from zend.com: You will have to log in with your zend.com account to begin the download process.



To install Zend Server Community Edition :

1. Extract by running the command:

```
tar -xzvf <package name>
```

2. CD (change the directory) to the extracted directory and run the installer with the command

```
./<Package_Name>/install.sh
```

3. Specify the prefix (/usr/local/by default): A new directory called zend is created under it (making the path using the default location: <install_path>).

After installing, a completion notification appears, informing you about the installation and what to do next. You can access the (Web) Administration Interface from: <https://localhost:10082/ZendServer>. Upon initial login, you are prompted to define your password.

Note:

To access the Administration Interface from a remote browser, make sure the IP is configured as an allowed IP in the Debugger tab. In the address, replace <LocalHost> with the IP of the machine on which Zend Server Community Edition is installed.

If you intend to use PHP and other tools provided by Zend Server Community Edition (pear and pecl) from the command line (PHP CLI), it is recommended that you add the `<install_path>/bin` directory to your `$PATH` environment variable.

This can be done in two ways:

- Per user profile
- For all users

The following procedure is intended for use with `bash`. If you are using a different shell, adjust the procedure accordingly.



To add the `<install_path>/bin` directory to your `$PATH` environment variable per user profile:

1. Using a text editor, open `.bashrc` (located in your home directory).
2. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace `<install_path>` with your Zend Server Community Edition installation path.

3. Save the file.
4. In order for this to take effect, close and reopen your shell or run the following command:

```
source ~/.bashrc
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.



To add the `<install_path>/bin` directory to your `$PATH` environment variable for all users:

1. Log in as root or use `sudo` to execute the following commands.
2. Using a text editor, open `/etc/profile`.
3. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace `<install_path>` with your Zend Server Community Edition installation path.

4. Save the file.
5. In order for this to take effect, close and reopen your shell or run the following command:

```
source /etc/profile
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.

Uninstalling Zend Server Community Edition

The following instructions describe how to uninstall the Tarball package:



To completely uninstall:

```
# <install_path>/bin/zendctl.sh stop
```

Ensure all daemons are stopped by running:

```
# <install_path>/bin/zendctl.sh status
```

```
# rm -rf <install_path>
```

DEB Installation

This method uses "aptitude" to handle the installations, upgrades and additional packages. Alternatively, you may choose any other tool that supports the DEB packaging format (i.e., Synaptic, Kpackage, etc). This method downloads files from the Internet and therefore requires that you have an active Internet connection, access to your distribution's repositories and root privileges for the server.

Note:

This procedure requires root privileges.

To acquire root privileges in Ubuntu, run the following command and type your password:

```
$ sudo -s.
```

Automatically Installing Zend Server Community Edition

The following procedure describes how to run a script that will automatically create your DEB or RPM repositories and install Zend Server.



1. Download the package called "Linux x86 Installer (RPM/DEB Setup Script)" from zend.com - <http://www.zend.com/products/server-ce/downloads>
2. Locate and extract the package:
ZendServer-X.X.X-RepositoryInstaller-linux.tar.gz
3. To change to the directory with the installer scripts run:
cd ZendServer-RepositoryInstaller-linux/
4. Depending on the PHP version, you want to use, run one of the following commands:
 - For Zend Server Community Edition with PHP 5.2 Support run:
install_zs.sh 5.2 ce
 - For Zend Server Community Edition with PHP 5.3 Support run:
install_zs.sh 5.3 ce

After installing, a completion notification will appear, with a notice that the servers have started.

To access the Administration Interface (Web) open your browser at: <https://localhost:10082/ZendServer> (secure) or <http://localhost:10081/ZendServer>.

Upon initial log in, you will be prompted to define your password.

Manually Installing Zend Server Community Edition

To install Zend Server Community Edition, the first thing you have to do is to setup the repository for downloading the Zend Server Community Edition package.



To setup the environment:

1. Define a repository by opening the following file: `/etc/apt/sources.list` and adding the line:

```
deb http://repos.zend.com/zend-server/deb server non-free
```

3. Add Zend's repository public key by running:

```
# wget http://repos.zend.com/zend.key -O- | apt-key add -
```

If you are using `sudo` to run each command the next command requires using `sudo` following the `|` (pipe) symbol as follows:

```
# wget http://repos.zend.com/zend.key -O- | sudo apt-key add -
```

4. To synchronize with Zend's repository run:

```
# aptitude update
```

Now you can use "aptitude" to handle the installations, upgrades and additional packages.



To install:

1. Once the repository is set up, run the appropriate command according to the product version and PHP support you require:

To install **Zend Server Community Edition** with **PHP 5.2** run:

```
# aptitude install zend-server-ce-php-5.2
```

To install **Zend Server Community Edition** with **PHP 5.3** run:

```
# aptitude install zend-server-ce-php-5.3
```

2. Each package locates and downloads all relevant packages from the web.

The actual installation will require your conformation.


After installing, a completion notification will appear, with a notice that the servers have started.


To access the Administration Interface (Web) open your browser at: <https://localhost:10082/ZendServer> (secure) or <http://localhost:10081/ZendServer>.

Upon initial log in, you will be prompted to define your password.

Additional Packages

There are additional packages that can be added after installing Zend Server Community Edition, using `'aptitude install'`.

Important Note:
 After installing a component on top of an existing Zend Server installation, you must click  for the changes to take effect.

You can remove each package individually with the following command `'apt-get remove'` and then the package name of the component you want to remove. To complete the removal process in the Zend Server Administration interface click .

Description	PHP 5.2	PHP 5.3
Additional PHP extensions	php-5.2-extra-extensions-zend-server	php-5.3-extra-extensions-zend-server
Java bridge package (Requires Sun JRE 1.5, 1.6 or later installed on your computer. Therefore, if you do not already have JRE installed please install it before using the Java Bridge. More information about JRE's and the latest updates are found in the SUN Website: http://java.sun.com/javase/downloads/index.jsp (jre-6u21-linux-x64.bin)	php-5.2-java-bridge-zend-server	php-5.3-java-bridge-zend-server
The Zend Guard Loader for running PHP, encoded with Zend Guard.	php-5.2-loader-zend-server	php-5.3-loader-zend-server
A phpMyadmin meta package that installs phpMyAdmin and attaches it to the Administration Interface via a link from the Dashboard.	phpmyadmin-zend-server	phpmyadmin-zend-server
Zend Framework's bundled Dojo.	zend-server-framework-dojo	zend-server-framework-dojo
Zend Framework's extra components.	zend-server-framework-extras	zend-server-framework-extras
Full PHP sources, patched by Zend	php-5.2-source-zend-server	php-5.3-source-zend-server
Zend Server Control Panel	control-panel-zend-server	control-panel-zend-server

The following extensions require the IBM DB2 runtime client (RTCL):		
PHP extension that enables access to the IBM DB2 Universal Database, IBM Cloudscape and Apache Derby databases.	php-5.2-ibmdb2-zend-server	php-5.3-ibmdb2-zend-server
PHP pdo_ibm extension.	php-5.2-pdo-ibm-zend-server	php-5.3-pdo-ibm-zend-server

You can download and install IBM's Runtime Client libraries from

<http://ftp.software.ibm.com/software/data/db2/express/>

Post Installation Configuration

If you intend to use PHP and other tools provided by Zend Server Community Edition (pear and pecl) from the command line (PHP CLI), it is recommended that you add the <install_path>/bin directory to your \$PATH environment variable.

This can be done in two ways:

- Per user profile
- For all users

The following procedure is intended for use with *bash*. If you are using a different shell, adjust the procedure accordingly.



To add the <install_path>/bin directory to your \$PATH environment variable per user profile:

1. Using a text editor, open *.bashrc* (located in your home directory).
2. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace <install_path> with your Zend Server Community Edition installation path.

3. Save the file.
4. In order for this to take effect, close and reopen your shell or run the following command:

```
source ~/.bashrc
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.



To add the `<install_path>/bin` directory to your `$PATH` environment variable for all users:

1. Log in as root or use sudo to execute the following commands.
2. Using a text editor, open `/etc/profile`.
3. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace `<install_path>` with your Zend Server Community Edition installation path.

4. Save the file.
5. In order for this to take effect, close and reopen your shell or run the following command:

```
source /etc/profile
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.

Upgrading Zend Server Community Edition

The following instructions describe how to upgrade Zend Server Community Edition using `'aptitude'`.

For more information on upgrading your Zend Server Community Edition, including additional upgrade scenarios, see [Upgrading](#).

If you are using Zend Server Cluster Manager see [Upgrading a Zend Server Cluster](#).



To perform these actions you must have root privileges.



To upgrade all packages installed run:

```
# aptitude update
# aptitude upgrade
```

To upgrade only Zend packages, run:

```
# aptitude install `dpkg --get-selections|grep zend| awk -F " " '{print $1}' |xargs`
```

The upgrade process locates newer packages and downloads them.

Uninstalling Zend Server Community Edition

The following instructions describe how to delete or uninstall using '*aptitude*'.



To perform these actions you must have root privileges.



To uninstall Zend Server Community Edition (leaving the configuration files in place) run:

```
# aptitude remove '~nzend.* '
```

To delete Zend Server Community Edition from the system with no traces left run:

```
# aptitude purge '~nzend.* '
```

RPM Installation (RHEL, CentOS, Fedora and OEL)

This method uses "yum" to handle all installations, upgrades and additional packages. Alternatively, you may choose any other tool that supports the RPM packaging format (e.g. Kpackage, etc).

This method downloads files from the Internet and therefore requires that you have an active Internet connection, access to your distribution's repositories and root privileges for the server.

PHP Note:

The Zend Server installation package will replace your distribution's PHP - this may create conflicts between RPM packages. If you cannot install one of Zend Server's components, it is recommended that you remove your distribution's PHP packages and try to install again.

SELinux Note:

SELinux users must change their system settings to permissive mode before starting the Zend Server Community Edition installation procedure, by executing the following command:

```
# setenforce permissive
```

Automatically Installing Zend Server Community Edition

The following procedure describes how to run a script that will automatically create your DEB or RPM repositories and install Zend Server.



1. Download the package called "Linux x86 Installer (RPM/DEB Setup Script)" from zend.com - <http://www.zend.com/products/server-ce/downloads>
2. Locate and extract the package:
ZendServer-X.X.X-RepositoryInstaller-linux.tar.gz
3. To change to the directory with the installer scripts run:
cd ZendServer-RepositoryInstaller-linux/
4. Depending on the PHP version, you want to use, run one of the following commands:
 - For Zend Server Community Edition with PHP 5.2 Support run:
install_zs.sh 5.2 ce
 - For Zend Server Community Edition with PHP 5.3 Support run:
install_zs.sh 5.3 ce

After installing, a completion notification will appear, with a notice that the servers have started.

To access the Administration Interface (Web) open your browser at: <https://localhost:10082/ZendServer> (secure) or <http://localhost:10081/ZendServer>.

Upon initial log in, you will be prompted to define your password.

Manually Installing Zend Server Community Edition

To install Zend Server Community Edition, first setup the repository for downloading the Zend Server Community Edition package.



To setup the environment:

Set up your Zend Server Community Edition repository by creating:
`/etc/yum.repos.d/zend.repo` and adding the following content:

```
[Zend]
name=Zend Server
baseurl=http://repos.zend.com/zend-server/rpm/$basearch
enabled=1
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key

[Zend_noarch]
name=Zend Server - noarch
baseurl=http://repos.zend.com/zend-server/rpm/noarch
enabled=1
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key
```

Now you can use 'yum' to handle installations or any other tool that supports the RPM packaging format.

To install:



1. Once the environment is setup, run the appropriate command according to the product version and PHP support you require:

To install **Zend Server Community Edition** with **PHP 5.2** run:

```
# yum install zend-server-ce-php-5.2
```

To install **Zend Server Community Edition** with **PHP 5.3** run:

```
# yum install zend-server-ce-php-5.3
```

2. To clean your packages cache and ensure retrieval of updates from the web, run:

```
yum clean all
```

After installing, a completion notification will appear, with a notice that the servers have started.

To access the Administration Interface (Web) open your browser at: <https://localhost:10082/ZendServer> (secure) or <http://localhost:10081/ZendServer>.

Upon initial log in, you will be prompted to define your password.

Additional Packages

There are additional packages that can be added after installing Zend Server Community Edition, using, `'yum install'`.

Important Note:

After installing a component on top of an existing Zend Server installation, you must click



for the changes to take effect.

You can remove each package individually with the following command `'yum remove'` and then the package name of the component you want to remove. To complete the removal process in the Zend Server Administration interface click "Restart PHP".

Description	PHP 5.2	PHP 5.3
Additional PHP extensions	php-5.2-extra-extensions-zend-server	php-5.3-extra-extensions-zend-server
Java bridge package (Requires Sun JRE 1.5, 1.6 or later installed on your computer. Therefore, if you do not already have JRE installed please install it before using the Java Bridge. More information about JRE's and the latest updates are found in the SUN Website: http://java.sun.com/javase/downloads/index.jsp (jre-6u21-linux-x64-rpm.bin)	php-5.2-java-bridge-zend-server	php-5.3-java-bridge-zend-server
The Zend Guard Loader for running PHP, encoded with Zend Guard.	php-5.2-loader-zend-server	php-5.3-loader-zend-server
A phpMyadmin meta package that installs phpMyAdmin and attaches it to the Administration Interface via a link from the Dashboard.	phpmyadmin-zend-server-php-5.2	phpmyadmin-zend-server-php-5.3
Zend Framework's bundled Dojo.	zend-server-framework-dojo	zend-server-framework-dojo
Zend Framework's extra components.	zend-server-framework-extras	zend-server-framework-extras
Full PHP sources, patched by Zend	php-5.2-source-zend-server	php-5.3-source-zend-server
Zend Server Control Panel	control-panel-zend-server	control-panel-zend-server

The following extensions require the IBM DB2 runtime client (RTCL):		
PHP extension that enables access to the IBM DB2 Universal Database, IBM Cloudscape and Apache Derby databases.	php-5.2-ibmdb2-zend-server	php-5.3-ibmdb2-zend-server
PHP pdo_ibm extension.	php-5.2-pdo-ibm-zend-server	php-5.3-pdo-ibm-zend-server
Informix client	php-5.2-pdo-informix-zend-server	php-5.3-pdo-informix-zend-server

You can download and install IBM's Runtime Client libraries from

<http://ftp.software.ibm.com/software/data/db2/express/>

CentOS and RHEL4 and 5 Note:

The `phpmyadmin-zend-ce` package depends on the availability of `phpMyAdmin` from your distribution's repositories. The default CentOS repositories for example do not offer `phpMyAdmin` and therefore require that you manually add the `rpmforge` repositories to your `yum` repositories list. For information on how to do this for CentOS see: <http://wiki.centos.org/AdditionalResources/Repositories/RPMForge>

Post Installation Configuration

If you intend to use PHP and other tools provided by Zend Server Community Edition (`pear` and `pecl`) from the command line (PHP CLI), it is recommended that you add the `<install_path>/bin` directory to your `$PATH` environment variable.

This can be done in two ways:

- Per user profile
- For all users

The following procedure is intended for use with `bash`. If you are using a different shell, adjust the procedure accordingly.



To add the `<install_path>/bin` directory to your `$PATH` environment variable per user profile:

1. Using a text editor, open `.bashrc` (located in your home directory).
2. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace `<install_path>` with your Zend Server Community Edition installation path.

3. Save the file.
4. In order for this to take effect, close and reopen your shell or run the following command:

```
source ~/.bashrc
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.



To add the <install_path>/bin directory to your \$PATH environment variable for all users:

1. Log in as root or use sudo to execute the following commands.
2. Using a text editor, open /etc/profile.
3. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
```

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace <install_path> with your Zend Server Community Edition installation path.

4. Save the file.
5. In order for this to take effect, close and reopen your shell or run the following command:

```
source /etc/profile
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.

Enabling RPM Package Signature Validation

Zend Server's RPM packages are signed using GPG. This can be used to validate package integrity and authenticity during installation. This is an optional step which is recommended for organizations with strict security policies.



To enable RPM package signature validation:

1. Start the manual installation process as described in ([link to manual install](#))
2. After creating the file /etc/yum.repos.d/zend.repo, change the value of all of the *gpgcheck* occurrences to '1'
3. After each *gpgcheck* occurrence add the following line:
gpgkey=http://repos.zend.com/zend.key
4. Continue the installation process

Before installing the first packages, *yum* will prompt you to approve importing the Zend GPG key from the URL above.

After following this procedure, whenever upgrading and installing Zend Packages *yum* will verify the packages before installing them.

Upgrading Zend Server Community Edition

The following procedure describes how to upgrade Zend Server Community Edition using 'yum'.

For more information on upgrading your Zend Server Community Edition, including additional upgrade scenarios, see [Upgrading](#).

If you are using Zend Server Cluster Manager see [Upgrading a Zend Server Cluster](#).

RPM Upgrade Note:

After upgrading, you will need to manually start your server by running the command:
<install_path>/bin/zendctl.sh start.



To perform these actions you must have root privileges.



To upgrade, run:

To upgrade all Zend related packages run the following command according to the package you have installed:

```
# yum update \*zend\*
```

To update any and all files in your system that are managed by 'yum' (not just Zend products) run:

```
# yum update
```

To update a specific component (in this example it is php-mcrypt) run the following command according to the package you have installed:

To upgrade **Zend Server Community Edition** with **PHP 5.2** run:

```
# yum update php-mcrypt-zend-server-ce-php-5.2
```

To upgrade **Zend Server Community Edition** with **PHP 5.3** run:

```
# yum update php-mcrypt-zend-server-ce-php-5.3
```

The upgrade process locates any components of the product version that are newer and downloads them.

Uninstalling Zend Server Community Edition

The following procedure describes how to uninstall Zend Server Community Edition:



To uninstall run:

```
zendctl.sh stop
```

And then:

```
yum remove -y 'deployment-daemon-zend-server' && yum remove -y  
'*zend*'
```

This will stop the Zend Server Community Edition daemons and remove the program, including any additional packages that were installed.

When uninstalling, the configuration files are not removed. They remain in the same location with an additional suffix: `.rpmsave` so that they can be reused in a newer installation. For example: a file called `example.ini` is renamed to `example.ini.rpmsave`, after you run the uninstall.

RPM (SLES and OpenSUSE) Installation

This section describes the installation process of Zend Server Community Edition on SLES and OpenSUSE, using zypper – the default package manager for SUSE.

This method downloads files from the Internet and therefore requires that you have an active Internet connection and root privileges for the server.

SELinux Note:

SELinux users need to change their system settings to permissive mode before starting the Zend Server Community Edition installation procedure, by executing the following command:

```
# setenforce permissive
```

Automatically Installing Zend Server Community Edition

The following procedure describes how to run a script that will automatically create your DEB or RPM repositories and install Zend Server.



1. Download the package called "Linux x86 Installer (RPM/DEB Setup Script)" from zend.com - <http://www.zend.com/products/server-ce/downloads>
2. Locate and extract the package:
ZendServer-X.X.X-RepositoryInstaller-linux.tar.gz
3. To change to the directory with the installer scripts run:
cd <Install_Path>/ZendServer-RepositoryInstaller-linux/
4. Depending on the PHP version, you want to use, run one of the following commands:

- **For Zend Server Community Edition with PHP 5.2 Support run:**
`install.sh 5.2 ce`
- **For Zend Server Community Edition with PHP 5.3 Support run:**
`install.sh 5.3 ce`

After installing, a completion notification will appear, with a notice that the servers have started.

To access the Administration Interface open your browser at: <https://localhost:10082/ZendServer> (secure) or <http://localhost:10081/ZendServer>.

Upon initial log in, you will be prompted to define your password.

Manually Installing Zend Server Community Edition

There are two methods for setting up repositories one method is for SLES 11 and OpenSUSE and the other for SLES 10.

To set up repositories in SLES 11 and OpenSUSE

The following commands are run in the shell as **root** or using **sudo**.



To set up the repositories:

1. Set up your Zend Server repository by creating:
/etc/zypp/repos.d/zend.repo and adding the following content:

On 32 bit systems run:

```
[Zend]
name=Zend Server
baseurl=http://repos.zend.com/zend-server/sles/ZendServer-$basearch
type=rpm-md
enabled=1
autorefresh=1
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key

[Zend_noarch]
name=Zend Server - noarch
baseurl=http://repos.zend.com/zend-server/sles/ZendServer-noarch
type=rpm-md
enabled=1
autorefresh=1
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key
```

On 64 bit systems run:

```
[Zend]
name=Zend Server
baseurl=http://repos.zend.com/zend-server/sles/ZendServer-
$basearch
type=rpm-md
enabled=1
autorefresh=1
```

```
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key

[Zend_noarch]
name=Zend Server - noarch
baseurl=http://repos.zend.com/zend-server/sles/ZendServer-noarch
type=rpm-md
enabled=1
autorefresh=1
gpgcheck=1
gpgkey=http://repos.zend.com/zend.key
```

2. Accept importing the Zend Key to your trusted keys list

Your repository is now ready for Zend Server Community Edition to be installed.

To set up repositories in SLES 10

The following commands are run in the shell as **root** or using **sudo**.



To set up the repositories:

1. Run the following command:

```
zypper service-add http://repos.zend.com/zend-
server/sles/ZendServer-noarch ZendServer-noarch
```

2. Answer the questions prompted as follows:
 - Accept importing the Zend Key to your trusted keys list
 - Accept importing the Zend Key into your key ring
3. Run the following command to add the Zend Server Community Edition repositories to your system:

On 32 bit systems run:

```
zypper service-add http://repos.zend.com/zend-
server/sles/ZendServer-i586 ZendServer-i586
```

On 64 bit systems run:

```
zypper service-add http://repos.zend.com/zend-
server/sles/ZendServer-x86_64 ZendServer-x86_64
```

Your repository is now ready for Zend Server Community Edition to be installed.

To install Zend Server Community Edition using zypper



To Install Zend Server Community Edition:

1. Once the environment is setup, run the appropriate command according to the product version and PHP support you require:
2. To install **Zend Server Community Edition** with **PHP 5.2** run:

```
# zypper install zend-server-ce-php-5.2
```

3. To install **Zend Server Community Edition** with **PHP 5.3** run:

```
# zypper install zend-server-ce-php-5.3
```

zypper automatically downloads and installs the selected packages.

Basic Usage

The Zend Server Community Edition Administration Interface is accessed from `https://localhost:10082/ZendServer` after the installation.

You should refer to the on-line help available through the product or to the "[Package Setup And Control Scripts](#)" guide for additional usage information.

Additional Packages

There are additional packages that can be added after installing Zend Server Community Edition, using '*zypper install*'.

Important Note:

After installing a component on top of an existing Zend Server installation, you must click

for the changes to take effect.

You can remove each package individually with the following command '*zypper remove*' and then the package name of the component you want to remove. To complete the removal process in the Zend Server Administration interface click "Restart PHP".

Description	PHP 5.2	PHP 5.3
Additional PHP extensions	php-5.2-extra-extensions-zend-server	php-5.3-extra-extensions-zend-server
Java bridge package (Requires Sun JRE 1.5, 1.6 or later installed on your computer. Therefore, if you do not already have JRE installed please install it before using the Java Bridge. More information about JRE's	php-5.2-java-bridge-zend-server	php-5.3-java-bridge-zend-server

and the latest updates are found in the SUN Website: http://java.sun.com/javase/downloads/index.jsp (jre-6u21-linux-x64.bin)		
The Zend Guard Loader for running PHP, encoded with Zend Guard.	php-5.2-loader-zend-server	php-5.3-loader-zend-server
A phpMyadmin meta package that installs phpMyAdmin and attaches it to the Administration Interface via a link from the Dashboard.	phpmyadmin-zend-server-php-5.2	phpmyadmin-zend-server-php-5.3
Zend Framework's bundled Dojo.	zend-server-framework-doj	zend-server-framework-doj
Zend Framework's extra components.	zend-server-framework-extras	zend-server-framework-extras
Full PHP sources, patched by Zend	php-5.2-source-zend-server	php-5.3-source-zend-server
Zend Server Control Panel	control-panel-zend-server	control-panel-zend-server
The following extensions require the IBM DB2 runtime client (RTCL):		
PHP extension that enables access to the IBM DB2 Universal Database, IBM Cloudscape and Apache Derby databases.	php-5.2-ibmdb2-zend-server	php-5.3-ibmdb2-zend-server
PHP pdo_ibm extension.	php-5.2-pdo-ibm-zend-server	php-5.3-pdo-ibm-zend-server

You can download and install IBM's Runtime Client libraries from

<ftp://ftp.software.ibm.com/software/data/db2/express/>

Upgrading Zend Server Community Edition

The following procedure describes how to upgrade Zend Server Community Edition using *zypper*.

For more information on upgrading your Zend Server Community Edition, including additional upgrade scenarios, see [Upgrading](#).

If you are using Zend Server Cluster Manager see Upgrading a Zend Server Cluster.

RPM Upgrade Note:

After upgrading, you will need to manually start your server by running the command:
<install_path>/bin/zendctl.sh start.



To perform these actions you must have root privileges.



To upgrade on SLES 11 and OpenSUSE run:

To upgrade all Zend related packages run the following command according to the package you have installed:

```
# zypper update '*zend*'
```

To upgrade on SLES 10 and OpenSUSE run:

To update any and all files in your system that are managed by 'zypper' (not just Zend products) run:

```
# zypper update -t package '*zend*'
```

To update a specific component (in this example it is php-mycrypt) run the following command according to the package you have installed:

The upgrade process locates any components of the product version that are newer and downloads them.

Uninstalling Zend Server Community Edition

The following procedure describes how to uninstall Zend Server Community Edition using *zypper*.

Run the commands in the shell as **root** or using *sudo*.



To uninstall Zend Server Community Edition on SLES 11 and OpenSUSE run:

```
zypper remove *zend*
```

To uninstall Zend Server Community Edition on SLES 10 run:

```
rpm -qa --qf "%{name}\n" | grep zend | xargs zypper remove
```

When uninstalling, configuration files are not removed. They remain in the same location and are renamed with an additional suffix (.rpmsave) so that they can be reused in a newer installation. For example: a file called example.ini is renamed to example.ini.rpmsave after running the un-install.

Mac OS X Installation

This section describes the process for installing Zend Server Community Edition on Mac OS X from a .dmg disk image. The disk image contains a .pkg installer for Zend Server Community Edition (including MySQL and phpMyAdmin) and the Zend Controller App. which you may optionally install after installing Zend Server Community Edition.

Note

If you are upgrading Zend Server Community Edition from an existing installation you must first perform the procedure described in "[Upgrading](#)".

Requirements:

- Mac OS X 10.4 (Tiger) or 10.5 (Leopard) both on Intel
- An account with administrator privileges
- A minimum of 200 megabytes of available disk space is required to unpack and install Zend Server Community Edition.
- The Oracle extensions (oci8, pdo_oci) and Oracle Instant Client library can only be installed on Mac OS X version 10.5 and above.

Installing Zend Server Community Edition

The following procedure describes how to install Zend Server Community Edition on Mac OS X.



To install Zend Server Community Edition :

1. After downloading the package, locate the downloaded file and double-click it to reveal the package file ZendServer.pkg. Double-click the package file to start the installation process.
2. Follow the instructions presented by the Installer. Pay attention to the Important Information presented during installation process.
3. If necessary, click "Change Install Location" to define a different location for installing Zend Server Community Edition .
4. Zend Server Community Edition is installed to /usr/local/zend. Additionally, ZendServer.app will be placed under /Applications through which you can access the Administration interface.

When you open the Administration Interface for the first time, you will be prompted to set your Zend Server Community Edition password.

This password is required for logging in to the Administration Interface every time you use Zend Server Community Edition . You should set the same password in the Zend Controller configuration when you launch it for the first time.

You can access your Apache's document root at `/usr/local/zend/apache2/htdocs`. Apache is configured to run on port 10088 by default (accessible at <http://localhost:10088/>).

Installing Zend Controller

The following procedure describes how to install Zend Controller.



To install Zend Controller:

To install Zend Controller, drag it from the disk image to `/Applications`.

After the installation, Zend Controller should be configured to access your Zend Server installation by setting your Zend Server password. You can access the Zend Controller configuration window by selecting Preferences from the Zend Controller menu.

Post Installation Configuration

If you intend to use PHP and other tools provided by Zend Server Community Edition (`pear` and `pecl`) from the command line (PHP CLI), it is recommended that you add the `<install_path>/bin` directory to your `$PATH` environment variable.

This can be done in two ways:

- Per user profile
- For all users

The following procedure is intended for use with `bash`. If you are using a different shell, adjust the procedure accordingly.



To add the `<install_path>/bin` directory to your `$PATH` environment variable per user profile:

1. Using a text editor, open `.bashrc` (located in your home directory).
2. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace `<install_path>` with your Zend Server Community Edition installation path.

3. Save the file.
4. In order for this to take effect, close and reopen your shell or run the following command:

```
source ~/.bashrc
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.



To add the <install_path>/bin directory to your \$PATH environment variable for all users:

1. Log in as root or use sudo to execute the following commands.
2. Using a text editor, open /etc/profile.
3. Add the following lines to the end of the file:

```
PATH=$PATH:<install_path>/bin
```

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<install_path>/lib
```

Replace <install_path> with your Zend Server Community Edition installation path.

4. Save the file.
5. In order for this to take effect, close and reopen your shell or run the following command:

```
source /etc/profile
```

You can now run the PHP binary provided by Zend Server Community Edition without typing its full path.

Upgrading Zend Server

Upgrading a Windows installation of Zend Server is accomplished by going to Zend's [download page](#) and selecting the relevant installation. When upgrading Zend Server Cluster Manager, upgrade the cluster manager, and then the nodes. For more information see [Upgrading a Zend Server Cluster](#).

Uninstalling Zend Server Community Edition

You can uninstall Zend Server, by running `sudo /usr/local/zend/bin/uninstall.sh` from a terminal window.

The uninstall script will perform the following:

- Stop all Zend Server processes
- Delete all Zend Server installed files
- Remove the .app from /Applications
- Remove Zend users

You must have administrator privileges to run the uninstall procedure.

Windows Installation

This section describes the three available processes for installing Zend Server Community Edition on Windows: Either via a native Windows installer, an unattended Installation or a Silent Installation.

Note

If you are upgrading Zend Server Community Edition from an existing installation older than Zend Server Community Edition Version 5.5, you must first perform the procedure described in “Upgrading”.

To run Zend Server Community Edition on Windows, you need the following:

- Generally, you should install Zend Server Community Edition on Windows using an account that has administrator rights. Otherwise, you may encounter problems with certain operations such as editing the PATH environment variable or accessing the Service Control Manager.
- Enough space on the hard drive to unpack and install (generally a minimum of 200 megabytes is recommended.)

Zend Server Community Edition for Windows is available in a binary distribution that contains a setup program .exe file. The .exe file installs everything you need to start using Zend Server immediately.

If you are encountering problems with Internet Explorer 7 running on Windows 2008 Server, see the following troubleshooting topic: [Windows: Internet Explorer Blocking Zend Server](#)

IIS Note:

Zend Server Community Edition running with IIS does not provide URL rewrite capabilities. If you require such capabilities (for example, when using Zend Framework based applications that use the default MVC components) refer to the following troubleshooting article.

Installing Zend Server Community Edition

Installing with the Native Windows Installer

The following procedure describes how to install Zend Server Community Edition on Windows using a binary distribution.

Note:

Users of previous versions of Zend Server Community Edition need to shut down and remove their existing Zend Server Community Edition installations manually before installing Zend Server Community Edition. See Section [“Upgrading”](#), for more information on upgrading from a previous version.



To install Zend Server Community Edition:

1. After completing the download, double-click on the .exe file to start the installation process.
2. There are three installation types available: Typical, Full, and Custom.
 - The **Typical** installation type installs the most common options and is recommended for most users. The installed components are Zend Optimizer+, Zend Loader, Zend Debugger, Zend Cache, Zend Framework and Oracle OCI Driver.
 - The **Full** installation type installs all components included in the installation package. The components are Zend Optimizer+, Zend Loader, Zend Debugger, Zend Cache, Java Server, Zend Framework, Oracle OCI Driver, phpMyAdmin, IBM DB2 RTCL and MySQL. The full installation package requires an Internet connection while running the installation, to download online components.
 - The **Custom** installation type gives you complete control over which packages you wish to install and the installation path that is used. The components are Zend Optimizer+, Zend Loader, Zend Debugger, Zend Cache, Java Server, Zend Framework, Oracle OCI Driver, phpMyAdmin, IBM DB2 RTCL and MySQL.
3. Click the **NEXT** button to advance to the Confirmation dialog.
4. If you choose the **Custom** installation type, click the **NEXT** button to advance to the **“Destination Location”** dialog.
5. Select a Web server on which to install the PHP and the Installation Location. Click the **NEXT** button to advance to the Select Features dialog.
6. Select the features to install by double-clicking the check box next to each feature. A single click on a feature in the list displays a description for the feature. Click the **NEXT** button to advance to the “Administration Interface Password” screen and from there, to the Confirmation dialog.

7. The Confirmation Dialog

Once you choose an installation type and choose your installation components, you advance to the confirmation dialog.

Your installation type and installation path are displayed for you to review.

8. To install Zend Server Community Edition (if you are satisfied with your settings), click the **INSTALL** button. To change your settings, click the **BACK** button.
9. To exit the Zend Server Community Edition Installation Wizard without installing Zend Server Community Edition, click the **CANCEL** button.

10. The Custom Installation

The custom installation installs select components and provides an option to choose the Web server on which to install PHP. After confirming the installation, if the port number is already in use, you are asked to specify a different port number. The selected Web server is configured to the port specified after completing the installation.

A browser opens after the installation, to display the Administration Interface's login screen. Use the password you specified in the installation process to log in. If it was selected during the installation, a shortcut is added to your desktop, otherwise, bookmarking the page at this point will help you to easily locate the link.

Note:

For information on running the installer in "Silent Mode" see the following Knowledge Base article:
<http://kb.zend.com/index.php?View=entry&EntryID=464>

Installing Additional Components

While running the Installer in custom mode, you can choose not to install certain components. If at any time, you want to add them or remove them, save the installation file and re-run in Modify mode or, if you did not keep the Installer file, go to the Control Panel, click Add/Remove programs and select "change" to run the Installer.

Important Note:

After installing a component on top of an existing Zend Server installation, you must click

A green button with a circular refresh icon and the text "Restart PHP".

for the changes to take effect.

Installed Components

Java Bridge

The Java Bridge is set to run by default and it requires that you have SUN Microsystems JRE 1.4 (or later) installed on your computer. Therefore, if you do not already have JRE installed, install it before using the Java Bridge. The Installer is set to detect your JVM: if you do not have one, the Installer prompts you to identify its location. Clicking **NO** continues the installation without the Java components. More information about JREs and the latest updates can be found on the Sun Microsystems website: <http://java.sun.com>.

Locating Installed Components

Zend Server Community Edition installs to the directory: *<install_path>\Zend Server*.

If you choose to install Apache from the Zend Server Community Edition installation, an additional folder containing your installed Apache is added to the installation path:

<install_path>\Zend Server

<install_path>\Apache2.2

If you choose to install phpMyAdmin from the Zend Server Community Edition installation, an additional folder containing phpMyAdmin is added to the installation path:

<install_path>\Zend Server

<install_path>\phpMyAdmin

Start Menu Options

The Installation Wizard creates a new entry in the Windows START menu under a Zend Server Community Edition menu heading.

The following entries are created within the new START menu section:

- Change Password - deletes your current password. Clicking this option automatically deletes your password and opens a new password definition page.
- Help and Reference - Opens the online help in a browser
- Uninstall - Initiates the Wizard based uninstall process
- Zend Server Community Edition - Opens the Zend Server Community Edition Administration Interface
- Zend Control Panel - Opens the Zend Controller

Upgrading Zend Server

Upgrading a Windows installation of Zend Server is accomplished by going to Zend's [download page](#) and selecting the relevant installation. When upgrading Zend Server Cluster Manager, upgrade the cluster manager, and then the nodes. For more information see [Upgrading a Zend Server Cluster](#).

Uninstalling Zend Server Community Edition

The following instructions describe how to uninstall Zend Server Community Edition:



To uninstall:

1. Use the Windows Control Panel: **Start | Control Panel | Add or Remove Programs**.
2. In the **Add or Remove Programs** dialog, locate and click the Zend Server Community Edition package in the list.
3. Click "Remove".
The Installer runs in uninstall mode.
4. Follow the instructions and click "Finish" to complete the uninstallation process.

This will stop the Zend Server Community Edition services and remove the program, including any additional packages that were installed.

PHP Extension List

Zend Server Community Edition, supports two PHP versions, PHP 5.2 and PHP 5.3. Each PHP version has its own list of extensions as follows:

[Zend Server CE PHP 5.2 Extensions](#)

[Zend Server CE PHP 5.3 Extensions](#)

Zend Server Community Edition Extension List - PHP 5.2

Common Extensions

Common extensions are installed and enabled by default in typical installations

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
bcmath	Enabled	Enabled	Enabled	Enabled	Arbitrary precision mathematics functions based on the bcmath (Binary Calculator) library
bz2	Enabled	Enabled	Enabled	Enabled	The bzip2 functions are used to transparently read and write bzip2 (.bz2) compressed files and streams
calendar	Enabled	Enabled	Enabled	Enabled	The calendar extension provides functions that simplify conversion between different calendar formats
com_dotnet	Not Shipped	Not Shipped	Not Shipped	Built-in	Component Object Model - An interface to Microsoft's COM / .NET environment
ctype	Enabled	Enabled	Enabled	Built-in	Character Classifications - Checks whether a character or string falls into a certain character class according to the current locale
curl	Enabled	Enabled	Enabled	Enabled	Enables you to connect to and communicate with different types of servers using various protocols - for example HTTP and FTP
date	Built-in	Built-in	Built-in	Built-in	Enables various date and time related functions that can handle retrieving the time, date formatting and more
dom	Built-in	Built-in	Built-in	Built-in	Enables operating on an XML document using the Document Object Model (DOM) API
exif	Enabled	Enabled	Enabled	Enabled	Enables access to image EXIF (Exchangeable Image File Format) meta data
filter	Built-in	Built-in	Built-in	Built-in	Provides a set of functions for validating and filtering data coming from insecure sources, such as user inputs
ftp	Enabled	Enabled	Enabled	Built-in	Provides low-level client access to FTP (File Transfer Protocol) servers
gd	Enabled	Enabled	Enabled	Enabled	Enables creation, manipulation and streaming of images and graphics in various formats
gettext	Enabled	Enabled	Enabled	Enabled	Provides a set of functions that allow

					internationalization of PHP applications through the GNU gettext API
hash	Built-in	Built-in	Built-in	Built-in	Enables direct or incremental processing of arbitrary length messages using a variety of hashing algorithms
iconv	Built-in	Built-in	Built-in	Built-in	Enables conversion between different character sets using the iconv library
imap	Enabled	Enabled	Enabled	Enabled	Provides mail and news access through the IMAP, POP3 and NNTP protocols
intl	Enabled	Enabled	Enabled	Enabled	Provides Unicode and global localization support to PHP applications using the ICU library
json	Enabled	Enabled	Enabled	Built-in	Implements the JavaScript Object Notation (JSON) data-interchange format
ldap	Enabled	Enabled	Enabled	Enabled	Provides access to LDAP (Lightweight Directory Access Protocol) based directory servers; Based on the OpenLDAP library
libxml	Built-in	Built-in	Built-in	Built-in	Provides basic API and infrastructure for other XML processing extensions
mbstring	Enabled	Enabled	Enabled	Enabled	Enables manipulation of strings encoded in multi-byte character encoding schemes
mcrypt	Enabled	Enabled	Enabled	Enabled	Provides support for multiple encryption algorithms using the mcrypt library
mhash	Enabled	Enabled	Enabled	Enabled	Provides support for multiple hashing algorithms using the mhash library. Can be used to create checksums, message digests, message authentication codes, and more
mime_magic	Enabled	Enabled	Enabled	Enabled	Enables automatic MIME-type detection based on various patterns in files
mysql	Enabled	Enabled	Enabled	Enabled	Provides legacy access to MySQL database servers. For new applications it is recommended to use the 'mysqli' extension
mysqli	Enabled	Enabled	Enabled	Enabled	MySQL Improved - Provides access to MySQL database servers. Enables the functionality provided by MySQL 4.1 and above
oci8	Enabled	Enabled	Disabled	Enabled	Oracle Call Interface - Provides access to Oracle database servers, supporting many of the advanced features provided by Oracle servers
openssl	Built-in	Built-in	Built-in	Built-in	This module utilizes the OpenSSL library for generation and verification of signatures and for

Zend Server Community Edition Installation Guide

					encrypting and decrypting data and streams
pcre	Built-in	Built-in	Built-in	Built-in	Provides a set of functions for string matching and manipulation based on Perl Compatible Regular Expressions syntax
pdo	Built-in	Built-in	Built-in	Built-in	Base PDO (PHP Data Objects) Driver - Defines a lightweight, consistent interface for accessing databases in PHP
pdo_mysql	Enabled	Enabled	Built-in	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to MySQL database servers
pdo_oci	Enabled	Enabled	Disabled	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to Oracle database servers using the OCI library
pdo_pgsql	Enabled	Enabled	Built-in	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to PostgreSQL database servers
pdo_sqlite	Built-in	Built-in	Built-in	Built-in	PDO (PHP Data Objects) driver that enable access from PHP to SQLite database files
pgsql	Enabled	Enabled	Enabled	Enabled	Provides access to PostgreSQL database servers
posix	Enabled	Enabled	Enabled	Not Shipped	Contains an interface to functions defined in the IEEE 1003.1 (POSIX.1) standards document which are not accessible through other means
reflection	Built-in	Built-in	Built-in	Built-in	Adds the ability to reverse-engineer classes, interfaces, functions and methods as well as extensions
session	Built-in	Built-in	Built-in	Built-in	Enables data persistence between consecutive requests of the same user session
simplexml	Built-in	Built-in	Built-in	Built-in	The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators
soap	Enabled	Enabled	Enabled	Enabled	The SOAP extension can be used to implement SOAP Servers and Clients
sockets	Enabled	Enabled	Enabled	Enabled	The socket extension implements a set of low-level socket communication functions, providing the possibility to act as a socket server as well as a client
spl	Built-in	Built-in	Built-in	Built-in	SPL is a collection of interfaces and classes that can be used to solve standard problems
sqlite	Enabled	Enabled	Enabled	Enabled	Enables usage of the SQLite Embeddable SQL Database Engine. Can be used for SQL database

					access without running a separate RDBMS process
standard	Built-in	Built-in	Built-in	Built-in	Standard PHP functions
tidy	Enabled	Enabled	Enabled	Enabled	Tidy HTML Clean and Repair - enables you to not only clean and otherwise manipulate HTML documents, but also traverse the document tree
tokenizer	Enabled	Enabled	Enabled	Enabled	The tokenizer functions provide an interface to the PHP tokenizer embedded in the Zend Engine. Using these functions you may write your own PHP source analyzing or modification tools without having to deal with the language specification at the lexical level
win32service	Not Shipped	Not Shipped	Not Shipped	Enabled	Allows management of Windows Services using PHP API
xml	Built-in	Built-in	Built-in	Built-in	Enables the creation of event-based XML document parsers using the SAX XML interface
xmlreader	Enabled	Enabled	Enabled	Enabled	The XMLReader extension is an XML Pull parser. The reader acts as a cursor going forward on the document stream and stopping at each node on the way.
xmlwriter	Enabled	Enabled	Built-in	Enabled	Provides a non-cached, forward-only writer for generating streams or files containing XML data in an efficient manner
xsl	Enabled	Enabled	Enabled	Enabled	The XSL extension implements the XSL standard, performing XSLT transformations using the libxslt library
zip	Enabled	Enabled	Enabled	Enabled	ZIP Archives - Enables you to transparently read ZIP compressed archives and the files inside them
zlib	Built-in	Built-in	Built-in	Built-in	Enables you to transparently read and write gzip (.gz) compressed files, through versions of most of the filesystem functions which work with gzip-compressed files

Extra / Additional Extensions

Extra extensions are shipped by Zend and can easily be installed but are not installed by default in typical installations

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
fileinfo	Enabled	Disabled	Disabled	Not Shipped	Allows retrieval of information regarding many different file types. This information includes file type and encoding, as well as more specific information such as dimensions, quality or length
gmp	Enabled	Disabled	Disabled	Disabled	These functions allow you to work with arbitrary-length integers using the GNU MP library
imagick	Enabled	Disabled	Disabled	Disabled	Enables image creation and manipulation using the ImageMagick API
memcache	Enabled	Disabled	Disabled	Disabled	Provides access to memcached - a highly efficient memory based caching daemon
ming	Enabled	Disabled	Disabled	Disabled	Provides a set of functions that can be used to create SWF ("Flash") format animations; Based on the open-source Ming library
mssql	Enabled	Disabled	Disabled	Disabled	Provides access to MS SQL Server database; Based on the open-source FreeTDS library
odbc	Enabled	Disabled	Disabled	Disabled	Provides access to several database servers through the Unified ODBC interface
pcntl	Enabled	Disabled	Disabled	Not Shipped	Process Control Functions - Process Control support in PHP implements the Unix style of process creation, program execution, signal handling and process termination
pdo_dblib	Enabled	Disabled	Disabled	Not Shipped	PDO (PHP Data Objects) driver that enable access from PHP to MSSQL and other databases accessible through the FreeDTS interface
pdo_odbc	Enabled	Disabled	Disabled	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to different databases through ODBC drivers or through the IBM DB2 Call Level Interface (DB2 CLI) library
shmop	Enabled	Disabled	Disabled	Disabled	Shared Memory - Shmop is an easy-to-use set of functions that allows PHP to read, write, create and delete Unix shared memory

					segments
sysvmsg	Enabled	Disabled	Disabled	Not Shipped	Enables System V messages support - The messaging functions may be used to send and receive messages to/from other processes. They provide a simple and effective means of exchanging data between processes, without the need for setting up an alternative using Unix domain sockets
sysvsem	Enabled	Disabled	Disabled	Not Shipped	Enables System V semaphore support - Semaphores may be used to provide exclusive access to resources on the current machine, or to limit the number of processes that may simultaneously use a resource
sysvshm	Enabled	Disabled	Disabled	Not Shipped	Enables System V shared memory support - Shared memory may be used to provide access to global variables
uploadprogress	Enabled	Disabled	Disabled	Disabled	Provides a mechanism for keeping track and showing progress when uploading files
wddx	Enabled	Disabled	Disabled	Disabled	WDDX (Web Distributed Data Exchange) - These functions are intended for work with the WDDX data exchange format
xmlrpc	Enabled	Disabled	Disabled	Disabled	Provides a set of functions that can be used to write XML-RPC servers and clients

Standalone Extensions

Standalone extensions are provided by Zend but are not included in any of the meta-packages / categories above.

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
ibm_db2	Enabled	Disabled	Not Shipped	Disabled	Provides functions that enable you to access the IBM DB2 Universal Database, IBM Cloudscape, and Apache Derby databases using the DB2 Call Level Interface
pdo_ibm	Enabled	Disabled	Not Shipped	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to IBM databases
pdo_informix	Enabled	Not Shipped	Not Shipped	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to Informix database servers
sqlsrv	Not Shipped	Not Shipped	Not Shipped	Disabled	Provides access to Microsoft SQL Server 2005 database servers. Developed by Microsoft

Zend Server Community Edition Extension List - PHP 5.3

Common Extensions

Common extensions are installed and enabled by default in typical installations

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
bcmath	Enabled	Enabled	Enabled	Enabled	Arbitrary precision mathematics functions based on the bcmath (Binary Calculator) library
bz2	Enabled	Enabled	Enabled	Enabled	The bz2 functions are used to transparently read and write bz2 (.bz2) compressed files and streams
calendar	Enabled	Enabled	Enabled	Enabled	The calendar extension provides functions that simplify conversion between different calendar formats
cgi-fcgi	Not Shipped	Not Shipped	Not Shipped	Built-in	PHP CGI/FastCGI Server API
com_dotnet	Not Shipped	Not Shipped	Not Shipped	Built-in	Component Object Model - An interface to Microsoft's COM / .NET environment
Core	Built-in	Built-in	Built-in	Built-in	Core PHP functionality
ctype	Enabled	Enabled	Enabled	Built-in	Character Classifications - Checks whether a character or string falls into a certain character class according to the current locale
curl	Enabled	Enabled	Enabled	Enabled	Enables you to connect to and communicate with different types of servers using various protocols - for example HTTP and FTP
date	Built-in	Built-in	Built-in	Built-in	Enables various date and time related functions that can handle retrieving the time, date formatting and more
dom	Built-in	Built-in	Built-in	Built-in	Enables operating on an XML document using the Document Object Model (DOM) API
ereg	Built-in	Built-in	Built-in	Built-in	Provides a set of string pattern matching functions using POSIX extended regular expressions.
exif	Enabled	Enabled	Enabled	Enabled	Enables access to image EXIF (Exchangeable Image File Format) meta data
fileinfo	Enabled	Enabled	Enabled	Enabled	Allows retrieval of information regarding many different file types. This information includes file

					type and encoding, as well as more specific information such as dimensions, quality or length
filter	Built-in	Built-in	Built-in	Built-in	Provides a set of functions for validating and filtering data coming from insecure sources, such as user inputs
ftp	Enabled	Enabled	Enabled	Built-in	Provides low-level client access to FTP (File Transfer Protocol) servers
gd	Enabled	Enabled	Enabled	Enabled	Enables creation, manipulation and streaming of images and graphics in various formats
gettext	Enabled	Enabled	Enabled	Enabled	Provides a set of functions that allow internationalization of PHP applications through the GNU gettext API
hash	Built-in	Built-in	Built-in	Built-in	Enables direct or incremental processing of arbitrary length messages using a variety of hashing algorithms
iconv	Built-in	Built-in	Built-in	Built-in	Enables conversion between different character sets using the iconv library
imap	Enabled	Enabled	Enabled	Enabled	Provides mail and news access through the IMAP, POP3 and NNTP protocols
intl	Enabled	Enabled	Enabled	Enabled	Provides Unicode and global localization support to PHP applications using the ICU library
json	Enabled	Enabled	Enabled	Built-in	Implements the JavaScript Object Notation (JSON) data-interchange format
ldap	Enabled	Enabled	Enabled	Enabled	Provides access to LDAP (Lightweight Directory Access Protocol) based directory servers; Based on the OpenLDAP library
libxml	Built-in	Built-in	Built-in	Built-in	Provides basic API and infrastructure for other XML processing extensions
mbstring	Enabled	Enabled	Enabled	Enabled	Enables manipulation of strings encoded in multi-byte character encoding schemes
mcrypt	Enabled	Enabled	Enabled	Enabled	Provides support for multiple encryption algorithms using the mcrypt library
mysql	Enabled	Enabled	Enabled	Enabled	Provides legacy access to MySQL database servers. For new applications it is recommended to use the 'mysqli' extension
mysqli	Enabled	Enabled	Enabled	Enabled	MySQL Improved - Provides access to MySQL database servers. Enables the functionality provided by MySQL 4.1 and above
mysqlnd	Not Shipped	Not	Not	Built-in	MySQL Native Driver - provides a PHP-native

Zend Server Community Edition Installation Guide

		Shipped	Shipped		
					underlying infrastructure for all MySQL extensions
oci8	Enabled	Enabled	Disabled	Enabled	Oracle Call Interface - Provides access to Oracle database servers, supporting many of the advanced features provided by Oracle servers
openssl	Built-in	Built-in	Built-in	Built-in	This module utilizes the OpenSSL library for generation and verification of signatures and for encrypting and decrypting data and streams
pcre	Built-in	Built-in	Built-in	Built-in	Provides a set of functions for string matching and manipulation based on Perl Compatible Regular Expressions syntax
pdo	Built-in	Built-in	Built-in	Built-in	Base PDO (PHP Data Objects) Driver - Defines a lightweight, consistent interface for accessing databases in PHP
pdo_mysql	Enabled	Enabled	Built-in	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to MySQL database servers
pdo_oci	Enabled	Enabled	Disabled	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to Oracle database servers using the OCI library
pdo_pgsql	Enabled	Enabled	Built-in	Enabled	PDO (PHP Data Objects) driver that enable access from PHP to PostgreSQL database servers
pdo_sqlite	Built-in	Built-in	Built-in	Built-in	PDO (PHP Data Objects) driver that enable access from PHP to SQLite database files
pgsql	Enabled	Enabled	Enabled	Enabled	Provides access to PostgreSQL database servers
phar	Enabled	Enabled	Enabled	Enabled	Allows running of complete PHP applications out of .phar package files
posix	Enabled	Enabled	Enabled	Not Shipped	Contains an interface to functions defined in the IEEE 1003.1 (POSIX.1) standards document which are not accessible through other means
reflection	Built-in	Built-in	Built-in	Built-in	Adds the ability to reverse-engineer classes, interfaces, functions and methods as well as extensions
session	Built-in	Built-in	Built-in	Built-in	Enables data persistence between consecutive requests of the same user session
simplexml	Built-in	Built-in	Built-in	Built-in	The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators
soap	Enabled	Enabled	Enabled	Enabled	The SOAP extension can be used to implement SOAP Servers and Clients

sockets	Enabled	Enabled	Enabled	Enabled	The socket extension implements a set of low-level socket communication functions, providing the possibility to act as a socket server as well as a client
spl	Built-in	Built-in	Built-in	Built-in	SPL is a collection of interfaces and classes that can be used to solve standard problems
sqlite	Enabled	Enabled	Enabled	Enabled	Enables usage of the SQLite Embeddable SQL Database Engine. Can be used for SQL database access without running a separate RDBMS process
standard	Built-in	Built-in	Built-in	Built-in	Standard PHP functions
tidy	Enabled	Enabled	Enabled	Enabled	Tidy HTML Clean and Repair - enables you to not only clean and otherwise manipulate HTML documents, but also traverse the document tree
tokenizer	Enabled	Enabled	Enabled	Enabled	The tokenizer functions provide an interface to the PHP tokenizer embedded in the Zend Engine. Using these functions you may write your own PHP source analyzing or modification tools without having to deal with the language specification at the lexical level
win32service	Not Shipped	Not Shipped	Not Shipped	Enabled	Allows management of Windows Services using PHP API
xml	Built-in	Built-in	Built-in	Built-in	Enables the creation of event-based XML document parsers using the SAX XML interface
xmlreader	Enabled	Enabled	Enabled	Enabled	The XMLReader extension is an XML Pull parser. The reader acts as a cursor going forward on the document stream and stopping at each node on the way.
xmlwriter	Enabled	Enabled	Built-in	Enabled	Provides a non-cached, forward-only writer for generating streams or files containing XML data in an efficient manner
xsl	Enabled	Enabled	Enabled	Enabled	The XSL extension implements the XSL standard, performing XSLT transformations using the libxslt library
zip	Enabled	Enabled	Enabled	Enabled	ZIP Archives - Enables you to transparently read ZIP compressed archives and the files inside them
zlib	Built-in	Built-in	Built-in	Built-in	Enables you to transparently read and write gzip (.gz) compressed files, through versions of most of the filesystem functions which work with gzip-compressed files

Extra / Additional Extensions

Extra extensions are shipped by Zend and can easily be installed but are not installed by default in typical installations

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
gmp	Enabled	Disabled	Disabled	Disabled	These functions allow you to work with arbitrary-length integers using the GNU MP library
imagick	Enabled	Disabled	Disabled	Disabled	Enables image creation and manipulation using the ImageMagick API
memcache	Enabled	Disabled	Disabled	Disabled	Provides access to memcached - a highly efficient memory based caching daemon
mssql	Enabled	Disabled	Disabled	Disabled	Provides access to MS SQL Server database; Based on the open-source FreeTDS library
odbc	Enabled	Disabled	Disabled	Disabled	Provides access to several database servers through the Unified ODBC interface
pcntl	Enabled	Disabled	Disabled	Not Shipped	Process Control Functions - Process Control support in PHP implements the Unix style of process creation, program execution, signal handling and process termination
pdo_dblib	Enabled	Disabled	Disabled	Not Shipped	PDO (PHP Data Objects) driver that enable access from PHP to MSSQL and other databases accessible through the FreeDTS interface
pdo_odbc	Enabled	Disabled	Disabled	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to different databases through ODBC drivers or through the IBM DB2 Call Level Interface (DB2 CLI) library
shmop	Enabled	Disabled	Disabled	Disabled	Shared Memory - Shmop is an easy-to-use set of functions that allows PHP to read, write, create and delete Unix shared memory segments
sysvmsg	Enabled	Disabled	Disabled	Not Shipped	Enables System V messages support - The messaging functions may be used to send and receive messages to/from other processes. They provide a simple and effective means of exchanging data between processes, without the need for setting up an alternative using Unix domain sockets
sysvsem	Enabled	Disabled	Disabled	Not	Enables System V semaphore support -

	Enabled	Disabled	Disabled	Shipped	Semaphores may be used to provide exclusive access to resources on the current machine, or to limit the number of processes that may simultaneously use a resource
sysvshm	Enabled	Disabled	Disabled	Not Shipped	Enables System V shared memory support - Shared memory may be used to provide access to global variables
uploadprogress	Enabled	Disabled	Disabled	Disabled	Provides a mechanism for keeping track and showing progress when uploading files
wddx	Enabled	Disabled	Disabled	Disabled	WDDX (Web Distributed Data Exchange) - These functions are intended for work with the WDDX data exchange format
xmlrpc	Enabled	Disabled	Disabled	Disabled	Provides a set of functions that can be used to write XML-RPC servers and clients

Standalone Extensions

Standalone extensions are provided by Zend but are not included in any of the meta-packages / categories above.

Extension	Linux (DEB/RPM)	Linux (tarball)	Mac OS X	Windows	Description
ibm_db2	Enabled	Disabled	Not Shipped	Disabled	Provides functions that enable you to access the IBM DB2 Universal Database, IBM Cloudscape, and Apache Derby databases using the DB2 Call Level Interface
pdo_ibm	Enabled	Disabled	Not Shipped	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to IBM databases
pdo_informix	Enabled	Not Shipped	Not Shipped	Disabled	PDO (PHP Data Objects) driver that enable access from PHP to Informix database servers
sqlsrv	Not Shipped	Not Shipped	Not Shipped	Disabled	Provides access to Microsoft SQL Server 2005 database servers. Developed by Microsoft

Using PECL

This is relevant only for Linux and Mac OS X



[Not applicable for IBM i]

[PECL](#) is the online repository for PHP extensions. PECL includes a directory of known extensions, including many additional extensions that are not bundled with the default PHP distribution or with Zend Server Community Edition.

Zend Server Community Edition includes a command line tool, *pecl*, that automates the download, compilation and installation of additional extensions from PECL.

Note:

The default Zend Server installation does not include the complete set of build tools that may be required to compile PHP extensions using *pecl*.

Make sure you have a C compiler (such as *gcc*) before using *pecl*.

Installing Zend Server Community Edition Additional Extensions

The following commands will install additional extensions using *pecl*.



To get a list of available extensions, run:

```
# <install_path>/bin/pecl list-all
```

To install an extension, run:

```
# <install_path>/bin/pecl install $extension_name
```

Note:

Make sure to verify that all required dependencies for compiling an extension are met. For example, to compile the *newt* extension, you must ensure that the *ncurses* library is available on the same machine.

Uninstalling Zend Server Community Edition Additional Extensions

The following commands will uninstall additional extensions using *pecl*.



To remove an extension, run:

```
# <install_path>/bin/pecl uninstall $extension_name
```

To get a list of commands, run without arguments:

```
# <install_path>/bin/pecl
```

Registering Zend Server Community Edition

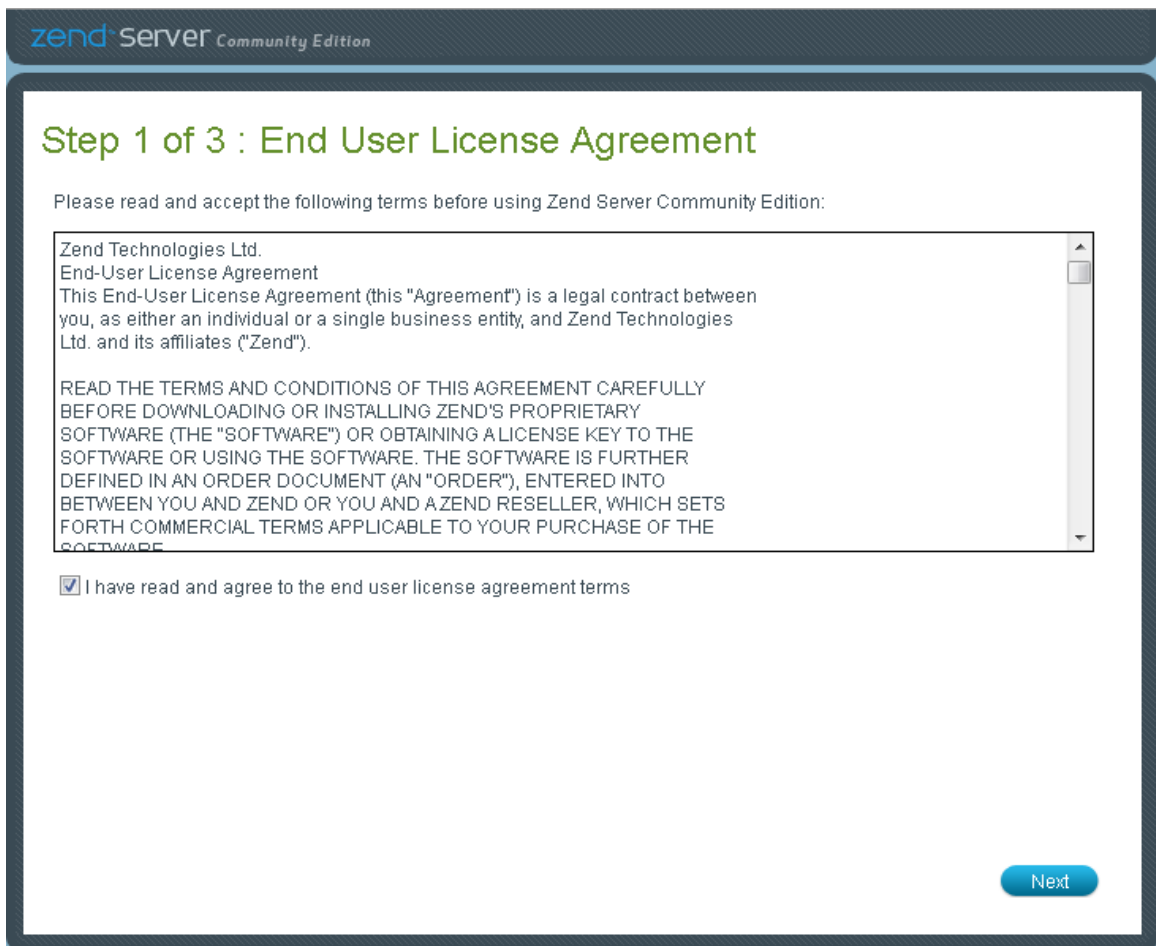
After installing your version of Zend Server, complete the short Setup wizard to begin working.

Setup Wizard

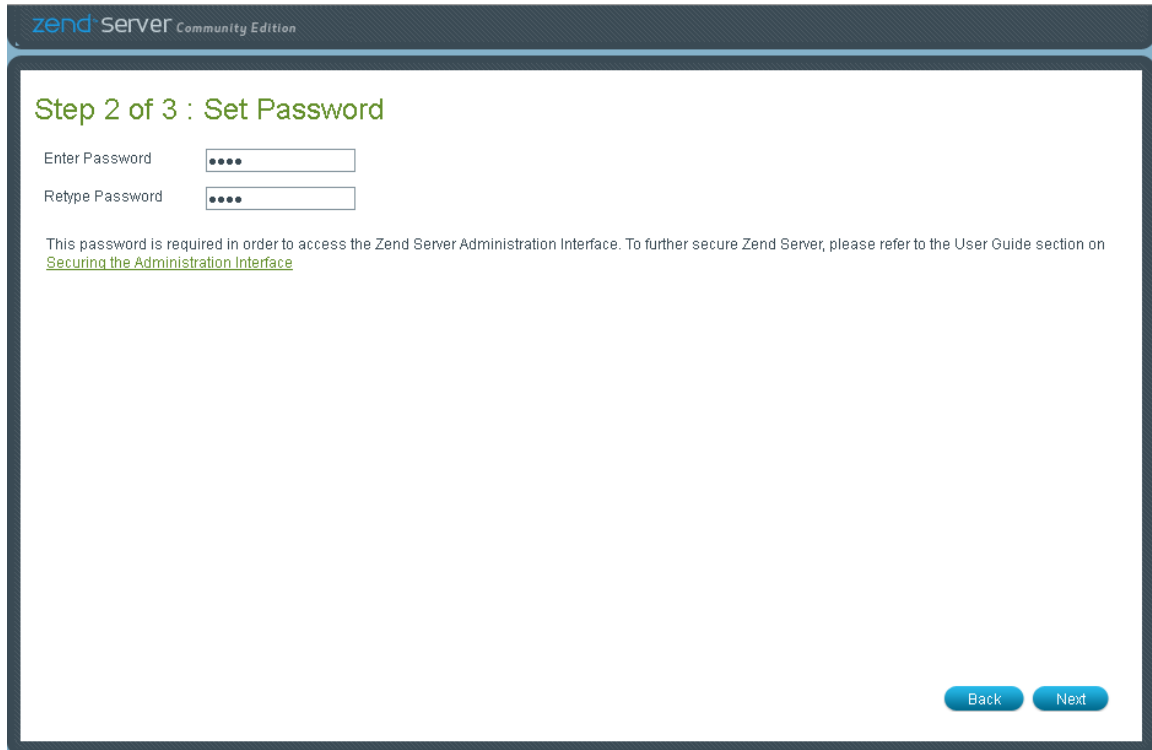


To setup Zend Server Community Edition:

1. The first time Zend Server Community Edition runs after installation, the Setup Wizard is displayed.



2. Read and accept the agreement on the License Agreement page, and click **Next**. The Set Password page is displayed.



The screenshot shows the 'Set Password' page in the Zend Server Community Edition installation wizard. The page title is 'Step 2 of 3 : Set Password'. It features two input fields: 'Enter Password' and 'Retype Password', both with masked characters (dots). Below the fields, there is a note: 'This password is required in order to access the Zend Server Administration Interface. To further secure Zend Server, please refer to the User Guide section on [Securing the Administration Interface](#)'. At the bottom right, there are two buttons: 'Back' and 'Next'.

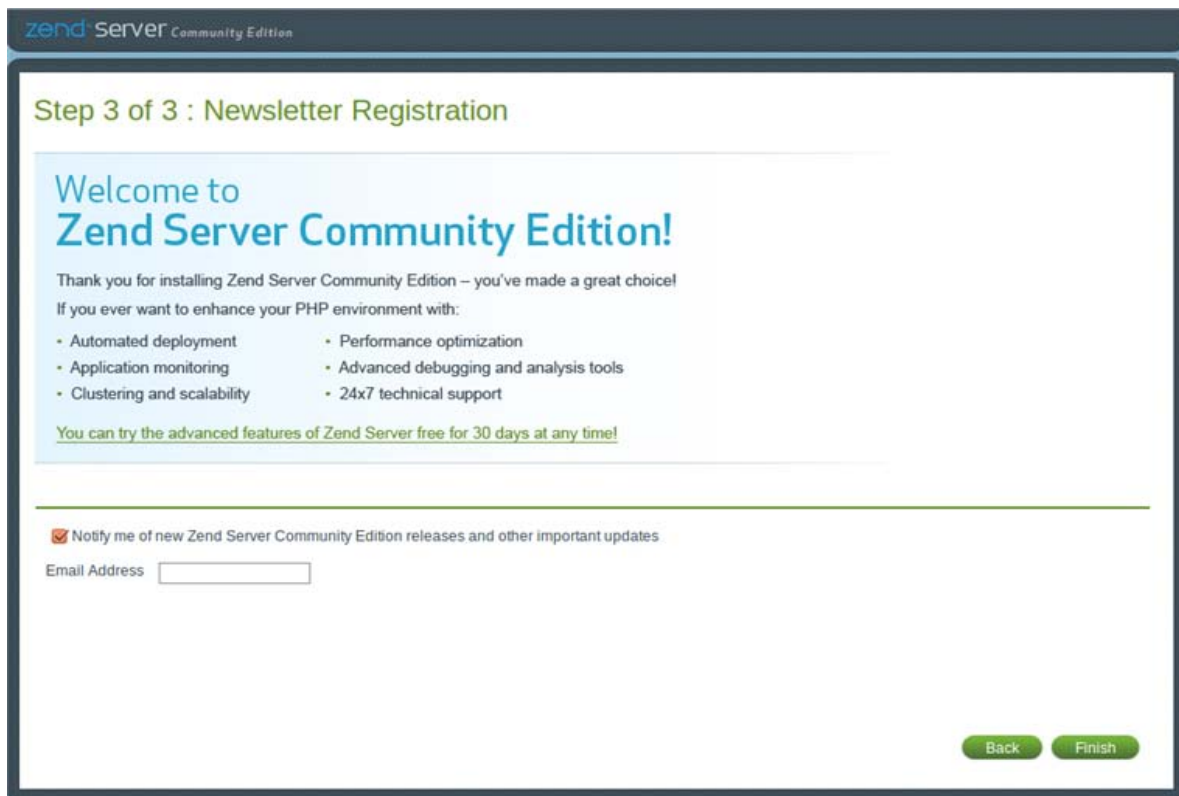
3. Enter a password for Zend Server Community Edition.
Your password is used to log in to the Administration Interface, either from the main login page accessed from your browser or from the Zend Controller. Passwords must be between 4 - 20 characters long and can be changed at a later stage from within the product. For more information, see Password Management.

Note:

If you are using the Zend Controller locally or remotely (i.e., Zend Server Community Edition and Zend Controller are located on separate machines), make sure that the Zend Controller settings match your Zend Server settings. Click here for instructions on how to change your Zend Controller settings according to your operating system.

4. Click **Next**.

The Licensing and Newsletter Signup page appears.



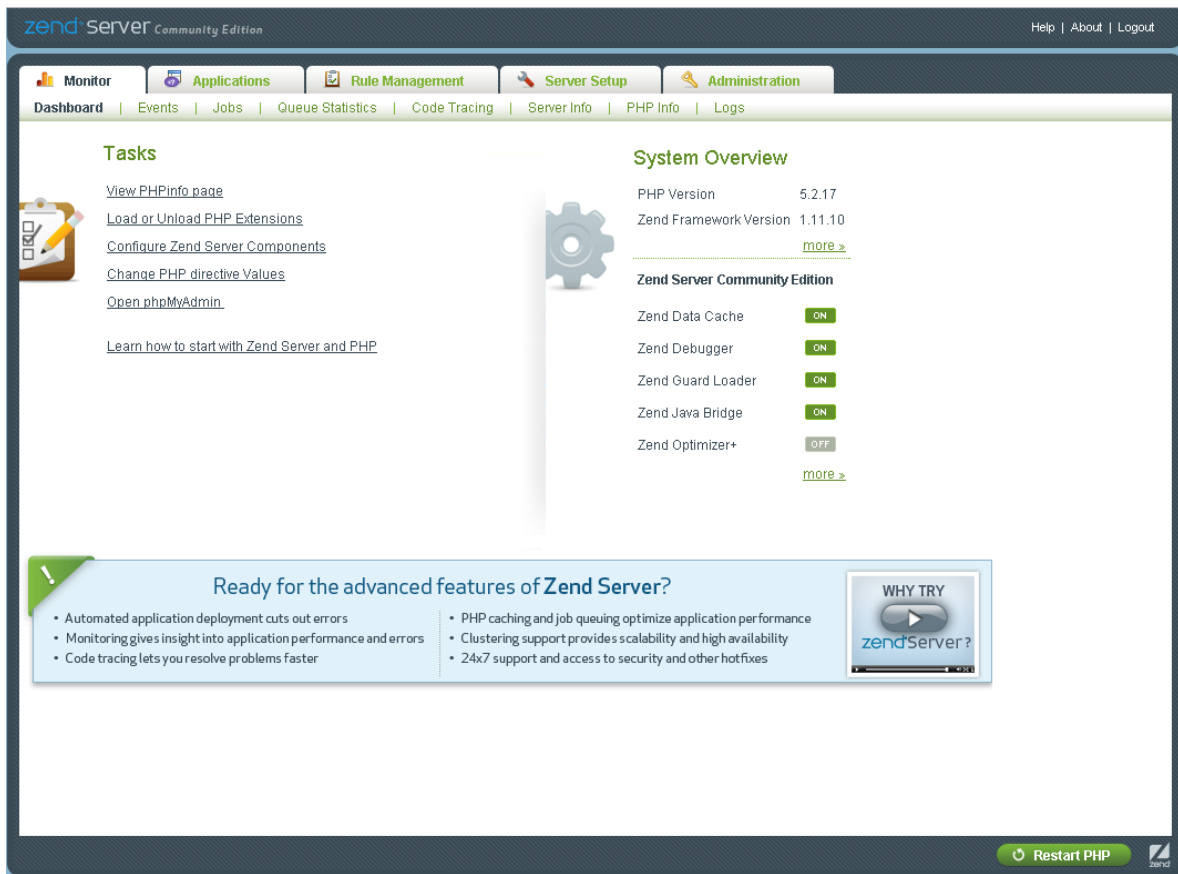
5. Mark the check box below, and enter an Email address to receive Zend Server Community Edition updates.

Note:

If you wish to try the advanced features that Zend Server Community Edition offers, click the link on the 'Welcome to Zend Server Community Edition' banner to go to the [Zend Server Community Edition Download page](#) and download an Evaluation license. For more information on upgrading to Zend Server Community Edition, see [Upgrading](#).

6. Click **Finish** to complete the Wizard..

Zend Server Community Edition runs with the Administration Interface Dashboard displayed.



To get started with Zend Server Community Edition, see [Working with Zend Server](#).

Upgrading

The information in this section refers to the different upgrade options available to users, according to the product currently installed and available installation type. If you are using Zend Server Cluster Manager see [Upgrading a Zend Server Cluster](#).

Upgrading your RPM Installation to Merge Configuration Files During Upgrade

To control what yum will do with configuration changes when installing or updating packages, use *yum-merge-conf*.



To install, run `# yum install yum-merge-conf`.

To use the plugin, pass `--merge-conf` to yum, in addition to the regular flags that you use.

Upgrading to a Newer Version of Zend Server Community Edition

The following instructions pertain to the process of installing a newer version of the same product, i.e., a newer version of Zend Server Community Edition or a newer version of Zend Server Community Edition CE (Community Edition). For example, upgrading from version 5.03 to 5.04 or from version 5.0 to 5.1.

Zend Server

Installation Type	Zend Server	Configuration Information	Comments
RPM	+		See Upgrading a RPM Installation .
DEB	+	Handles all configuration upgrades.	See Upgrading a DEB Installation .
Tarball	-		See Tarball Installation .
Mac OS X	-		See Upgrading a Mac OS X Installation .
Windows	+	Upgrades include existing configurations.	The installer automatically identifies if it is a new installation or an upgrade. See Upgrading a Windows Installation .
SLES and OpenSUSE	+	Upgrades all Zend related packages.	See Upgrading a RPM (SLES and OpenSUSE) Installation .

Zend Server CE

Installation Type	Community Edition	Configuration Information	Comments
RPM	+		See Upgrading a RPM Installation .
DEB	+	Handles all configuration upgrades	See Upgrading a DEB Installation .
Tarball	+	A separate backup of the ZendServer/etc/ directory is created.	Manually restore the old configuration file if you want to keep configurations. The configuration file is placed in the same location as the <i>.conf</i> file and is renamed with a timestamp. To use the backup rename to <i>.conf</i> .
Mac OS X	+	When upgrading, the following information will be preserved.	<p>apache2/htdocs apache2/conf apache2/conf.d Except for apache2/conf.d/zendserver_gui.conf etc/php.ini etc/conf.d/*.ini All directories under share/pear/ (but not .php files) gui/application/data/zend-server-user.ini gui/application/data/logfiles.xml MySQL data directory MySQL my.cnf file</p>
Windows	+	Upgrades include existing configurations.	The installer automatically identifies if it is a new installation or an upgrade.
SLES and OpenSUSE	+	Upgrades all Zend related packages.	See Upgrading a RPM (SLES and OpenSUSE) Installation .

Upgrading from Zend Server CE to Zend Server

This procedure can be performed using the Zend Repository Installer script (available in Zend Server's [Download Page](#) | Linux Tab | "Zend Server (DEB/RPM Installer Script)") by running the command `./install.sh <php version>` in the script's directory after download.

Installation Type	Availability	Configuration Information	Comments
RPM	+		You must first add the new repository to your sources.list. Use <i>yum-merge-conf</i> to manage configurations (see above).
DEB	+	Handles all configuration upgrades	You must first add the new repository to your sources.list.
Tarball	-		
Mac OS X	-		
Windows	+	Configuration information is imported from: ZendServer\etc\ ZendServer\GUI\application\data\ Apache2\conf/	
SLES and OpenSUSE	+	Upgrades all Zend related packages.	

RPM Upgrade Note:

After upgrading from the Community Edition to Zend Server, you will need to manually start your server by running the command: `<install_path>/bin/zendctl.sh start`.

Upgrading from Zend Server Community Edition to Zend Server from within the Product

Alternatively, you have the option to upgrade to Zend Server from within the product.



To upgrade to Zend Server:

1. Go to the [Zend Server Download page](#) to receive a valid license.
2. In your product, go to **Administration | License and Password**.

The screenshot shows the Zend Server Administration interface. The top navigation bar includes 'Monitor', 'Applications', 'Rule Management', 'Server Setup', and 'Administration'. The 'Administration' tab is active, and the 'License and Password' sub-tab is selected. The main content area is titled 'Activate Zend Server Features' and contains the following sections:

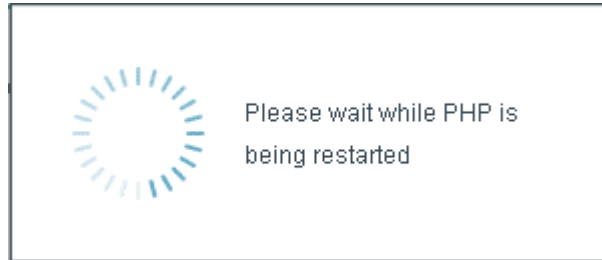
- Get a free 30-day trial license**: A green button.
- Zend Server License**: Two input fields for 'Order number' and 'License key'.
- Update License**: A button.
- Change Password**: Three input fields for 'Enter your current password:', 'Choose a new password:', and 'Confirm your new password:'. Below them is a 'Change Password' button.
- Zend Server Feature Content**: A section with a toggle for 'Zend Server feature information displayed:' set to 'Yes' (radio button selected). Below it is an 'Update' button.

At the bottom right of the interface, there is a 'Restart PHP' button and a Zend logo.

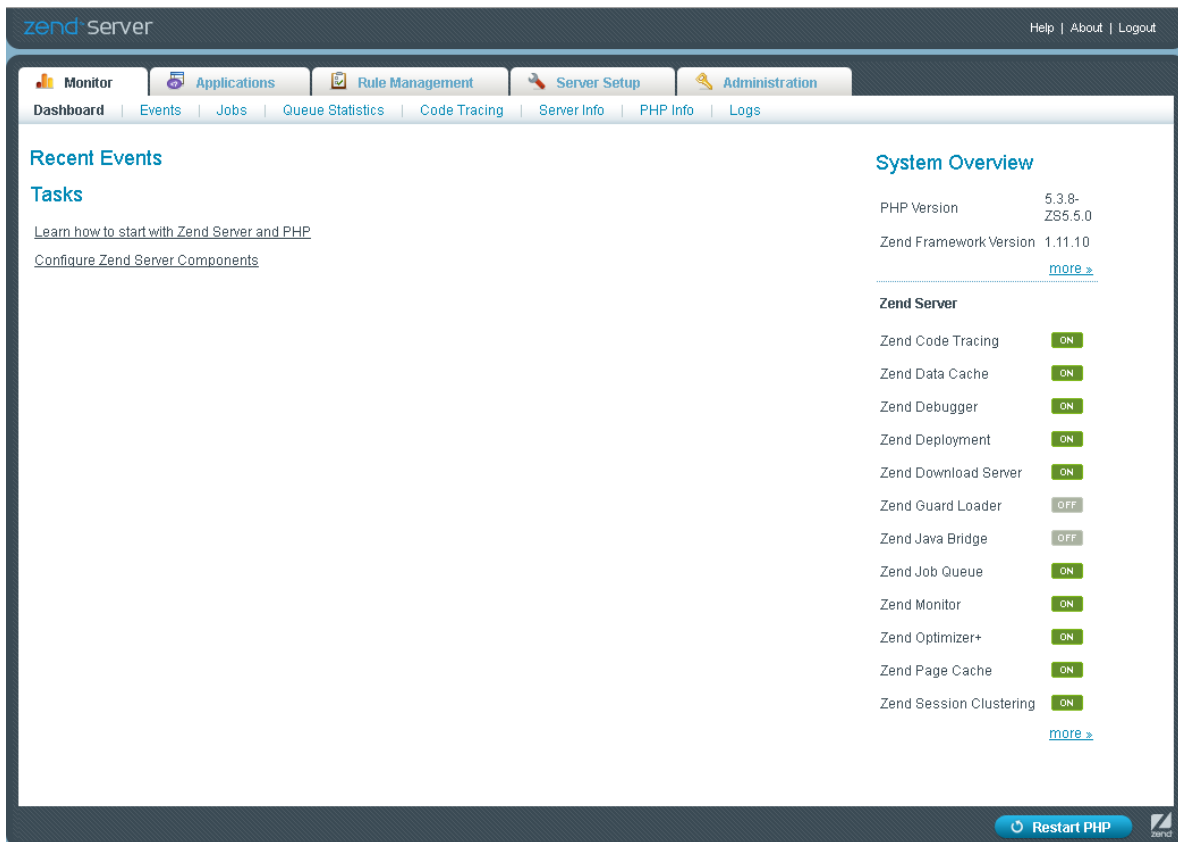
2. In the Zend Server License area, enter your license details as follows:
 - Order number- Enter your license number/name
 - License Key- Enter your license key

3. Click [Update License](#) to apply new licensing details.

The progress indicator is displayed indicating that the validation of the license details is taking place.



4. Zend Server Community Edition reloads the Administration Interface with all Zend Server Community Edition features enabled.



To get started with the newly available Zend Server features, see [Getting Started](#) and [Working with Zend Server](#).

For more information on registering Zend Server and license expiration, see [Registering Zend Server](#).

Manual Rollback

Alternatively, you can remove Zend Server and reinstall the Community Edition. To preserve your configurations, back up your configuration files before you remove Zend Server and return them after you reinstall the Community Edition.

The recommended directories to backup are:

- **In Windows:**
 - ZendServer\etc\
 - ZendServer\GUI\application\data\
 - Apache2\conf\
- **In Linux:**
 - ZendServer/etc/
 - ZendServer/GUI/application/data/
 - Apache2/conf/

Upgrading PHP 5.2 to PHP 5.3

This procedure describes how to manually update your Zend Server PHP 5.2 to PHP 5.3.

Note:

This procedure can also be performed using the Zend Repository Installer script (available in Zend Server's [Download Page](#) | Linux Tab | "Zend Server (DEB/RPM Installer Script)") by running the command `./upgrade_zs_php.sh <php version>` in the script's directory after download.

RPM



To update your PHP on RPM:

1. Remove Zend Server by running: `yum remove -y "zend-server*-php-5.*" && yum remove -y "**zend**"`
2. Restore the configuration files by running from the shell: `for conf in $(find /usr/local/zend/ -name "*.rpmsave"); do echo "Restoring $conf"; mv $conf `echo $conf | sed "s@.rpmsave@@"`; done`
3. Update the repositories by running: `yum clean all`
4. Install the PHP 5.3 version of all packages you installed for 5.2, for example: `yum install zend-server-php-5.3 php-5.3-extra-extensions-zend-server`

SLES 11



To update your PHP on SLES:

1. Remove Zend Server by running: `zypper -n remove "*zend*"`
2. Restore the configuration files by: running from the shell: `for conf in $(find /usr/local/zend/ -name "*.rpm*save");do echo "Restoring $conf";mv $conf `echo $conf|sed "s@.rpm*save @@";done`
3. Update the repositories by running: `zypper refresh`
4. Install the PHP 5.3 version of all packages you installed for 5.2, for example: `zypper install zend-server-php-5.3`

DEB



To update your PHP on DEB:

1. Update the repository by running: `aptitude update`
2. Install the PHP 5.3 version of all packages you installed for 5.2, for example: `aptitude install zend-server-php-5.3 php-5.3-extra-extensions-zend-server php-5.3-java-bridge-zend-server php-5.3-loader-zend-server`

Windows and Mac OS X



To update your PHP on Windows or Mac OS X, simply download the PHP 5.3 version of Zend Server from the product's [Downloads Page](#) on top of your existing one. There is no need to remove the prior version in order to update.

Migrating to Zend Server Community Edition from Zend Platform/Zend Core

Currently there is no automated process for upgrading from Zend Platform/Zend Core to Zend Server Community Edition. However, it is possible to install Zend Server Community Edition on a separate machine and manually transfer part of the Zend Platform/Zend Core configurations to the machine running Zend Server Community Edition. This may help save some time configuring your Zend Server Community Edition environment.

The following configuration files can be manually copied and used to replace Zend Server Community Edition configuration files:

- Apache configuration files
- php.ini

After transferring any configuration from Zend Platform/Zend Core to Zend Server Community Edition it is highly recommended to thoroughly test your applications before permanently applying these changes.

Downgrading/Rollback to an Older Version

Rollback is the process of reverting to an older version of Zend Server Community Edition.

Currently there is no automated process for this. Therefore, users who want to rollback should first uninstall their current version and only then install an older version of Zend Server Community Edition.

Note:

This does not refer to Updates. You can rollback to remove recently added updates. For more information on updating go to Update Notifications.

Post Installation

Package Setup and Control Scripts

Package setup and control scripts, refer to the management of the different components included in Zend Server Community Edition . A list of the components that are installed and running on **your** system can be found in the Administration Interface in **Server Setup | Components**.

Which components are installed depends on the chosen installation method and product version.

[For information on Windows click here](#)

[For information on Linux and Mac OS X click here](#)

Windows: Package Setup and Control Scripts

The following section describes how Start/Stop services on MS Windows.

Starting Services on MS Windows

All Zend Server Community Edition component services are managed by the MS Windows Service Manager.

To Start/Stop (run) or restart any of the services go to **Start | Control Panel | Administrative Tools | Services**.

The installed components that have services are displayed with a Zend Prefix.

Additional components can be added by running the installer in modify mode:



To add components:

1. Use the Windows Control Panel **Start | Control Panel | Add or Remove Programs**.
2. In the **Add or Remove Programs** dialog, locate and click on the Zend Server Community Edition package in the list.
3. Click **Change**.
The installer will start to run in modify mode.
4. Click **Modify** and select the components you want to add in the Custom Setup dialog.

This adds (and can also remove) any additional packages that are selected.

Controlling Zend Server Community Edition from Startup

Zend Server Community Edition by default is installed to start at boot time. If you are running Zend Server Community Edition on Apache the Apache Web Server manager will also be started. If you are using IIS it will run according to the settings you defined. For both webservers the default is to load automatically at boot time.

You can, if you want to, change what is loaded at boot time. You can even control which specific services are started or not when Zend Server Community Edition starts - this is useful if, for example, if you are not using [Job Queue](#) and do not want this service to run for nothing.

The Zend Server Community Edition services are controlled via the MS Windows service manager and are as follows:

- Java Bridge - Not installed by default
- MySQL(Optional) – Automatically starts according to standard MySQL settings.

Additional Information

- **Apache Web Server** - The service name is Apache2.2-Zend, and it automatically starts after reboot.

To change the setting to run or not run at boot time use the MS Windows service manager.

The command to start/stop the service from the command line is to first go to the directory `<install_dir>\Zend\Apache2\bin` and run `httpd.exe -k [start|stop] -n Apache2.2-Zend`.

-or -

- **IIS(5,6,7)** – depends on user settings. The command line to start/stop service (all versions): `net [start|stop] w3svc`.

More info about changing IIS configurations can be found in <http://www.iis.net/>.

Linux Mac: Package Setup and Control Scripts

The following section describes how to do control Zend Server Community Edition components from the command line.

Controlling Zend Server Community Edition Components from the Command Line

The setup and control scripts control the optional components that come with Zend Server Community Edition for the DEB, Tarball and RPM packages.



To control the Administration Interface's dedicated server, run:

```
# <install_path>/bin/lighttpdctl.sh stop/start/restart:
```

To set the Administration Interface's password, run:

```
# <install_path>/bin/gui_passwd.sh
```

To setup the Java Bridge, run:

```
#<install_path>/bin/setup_jb.sh
```

To control (start/stop) the Java Bridge daemon, run:

```
# <install_path>/bin/java_bridge.sh stop/start/restart
```

Command Line Actions

The following lists the possible actions that can be done to the Zend Server Community Edition components from the command line:

Usage: `<install_path>/bin/zendctl.sh <action>`.

Zend Server Community Edition

start - Start all Zend Server Community Edition daemons

stop - Stop all Zend Server Community Edition daemons

restart - Restart all Zend Server Community Edition daemons

version - Print Zend Server Community Edition version

status - Get Zend Server Community Edition status

Apache

start-apache - Start Apache only

stop-apache - Stop Apache only

restart-apache - Restart Apache only

LightHttpd

start-lighttpd - Start lighttpd only

stop-lighttpd - Stop lighttpd only

restart-lighttpd - Restart lighttpd only

Java Bridge

setup-jb - Setup Java bridge

Running *zendctl.sh* will show a list of uses and only after running *setup_jb.sh* will the following additional options be available:

start-jb - Start Java bridge only

stop-jb - Stop Java bridge only

restart-jb - Restart Java bridge only

Controlling Zend Server Community Edition from Startup

Zend Server Community Edition by default is installed to start at boot time. You can, if you want to, change that, using native OS tools (using *chkconfig* or *update-rc.d*). You can even control which specific daemons are started or not when Zend Server Community Edition starts - this is useful if, for example, if you are not using Job Queue and do not want this daemon to run for nothing.

The Zend Server Community Edition daemons are controlled via: */usr/local/zend/bin/zendctl.sh* which is symlinked to */etc/init.d/zend-server*

Zend Server Community Edition postinstall scripts call:

- Debian and Ubuntu: *update-rc.d zend-server defaults*
- CentOS, Fedora and OEL): */sbin/chkconfig --add zend-server*
- MAC (the procedure is different): */Library/StartupItems/ZendServer_init/*

Log Rotation

This Item is only relevant for Linux and Mac OS X.

In production environments, it is important to periodically compress/archive or truncate log file contents. Controlling your log file size prevents unnecessary disk consumption due to bloated log files. The following instructions describe how to override the native Zend Server Community Edition log rotation mechanism and use *logrotate*.

Note:

logrotate is not part of the Zend Server Community Edition product. To add this component you can use *yum* and *aptitude*, according to your distribution's repository.

For example, to locate the package in *yum*, run `# yum search logrotate`. You can do the same with *aptitude* too. If your distribution does not include this package in its repositories, you can download the source from here: <https://fedorahosted.org/logrotate/>.

Configuring Log Rotation for Zend Server Community Edition Logs

The following procedure describes how to configure automatic log rotation for all Zend Server Community Edition logs, using the *logrotate* daemon. For full details on the third party *logrotate* utility, see the [utility's man page](#) (`man logrotate`).

This daemon is installed by default, or can be easily installed on all Zend Server Community Edition supported Linux distributions.

Before using *logrotate*, make sure that the Zend component internal log rotation is disabled (see [To Disable Log Rotation](#)).



To configure log rotation:

1. Log in as root or use sudo to execute the following commands.
2. Create a file called zendserver using a text editor and save it in /etc/logrotate.d/, with the following content:

```
/usr/local/zend/var/log/*.log {  
  
    size 5M  
  
    missingok  
  
    rotate 10  
  
    compress  
  
    delaycompress  
  
    copytruncate  
  
}
```

3. Save the file.

You have now created a configuration file for all the files located in /usr/local/zend/var/log/ that will be picked-up by *logrotate*. **According to the above mentioned configuration, *logrotate* checks the file size of each log, every time it is executed (through a daily cron job on most systems).** If the file size exceeds 5Mb, the log file is archived by moving the content to a new file and truncating the log file. The new file's name is the same name with an additional number added to the file name. According to the example, *logrotate* created up to ten backup files. After exceeding ten files, the oldest file is deleted and replaced with new content.

The code example describes the following settings:

- *size 5M* = file size to rotate
- *missingok* = if the file is not found, do not generate an error.
- *rotate 10* = keep up to ten backup files.
- *compress* = compress archive log files, using gzip.
- *delaycompress* = do not compress the newest file created.
- *copytruncate* = rotation method. In this case, copy the content to a new file and truncate the active log.

To Disable Log Rotation

The following procedure describes how to disable Zend component internal log rotation.



To disable log rotation, set the `log_rotation_size` directive to 0.

The default log rotation directive value is 10mb.

Ports and Services

Web Server Ports

According to your choice of operating system and installation method, your web server on which Zend Server is installed, will be listening on a different port.

- **Linux using DEB or RPM packages:**
Zend Server will be installed on the distribution supplied web server which listens to port 80 by default. This can be set to another port by editing your Apache configuration.
- **Linux using the Tarball installation and Mac OS X:**
Zend Server will be installed on the bundled Apache web server which listens on port 10088 by default. This can be set to another port by editing your Apache configuration file at `/usr/local/zend/apache2/conf/httpd.conf` and changing the value of the directive `Listen`.
- **Windows on IIS:**
Zend Server will be installed on the built-in IIS web server which listens to port 80 by default. This can be set to another port from the IIS configuration Manager.
- **Windows on Apache:**
Zend Server will be installed on the bundled Apache web server which listens on port 80 by default. unless a different port was selected during installation. This can be set to another port from your Apache configuration file at `<install_dir>\Apache2\conf\httpd.conf` and changing the value of the directive `Listen`.

Firewall Settings

The following document describes the minimal set of ports that must be opened in your firewall in order for the different Zend Server and Zend Server Cluster Manager components to function.

If you are not using one of the Zend Server or Zend Server Cluster Manager components listed below, you are not required to open any of its related ports.

Note

Most ports can be configured. Each component's relevant documentation includes configuration details.

Zend Server Administration Interface (Web GUI)

Function	Port / Protocol	Comments
Web GUI Access	TCP/10081 (HTTP) from client browser to Zend Server and from Zend Server Cluster Manager to Zend Server	Should be open for administrative access, and between Zend Server Cluster Manager and cluster members. May be blocked if HTTPS (10082) is always used.
Web GUI Access	TCP/10082 (HTTPS) from client browser to Zend Server and from Zend Server Cluster Manager to Zend Server	Linux only. Should be open for administrative access, and between Zend Server Cluster Manager and cluster members. May be blocked if HTTPS is never used.
Local Web Server Control for Zend Server GUI	TCP/10083 (HTTP) on localhost on Zend Server	Must be open for localhost access only

Debugging and Profiling

Function	Port / Protocol	Comments
Debugging / Profiling in open LAN	TCP/10137 (Proprietary Debugger Protocol) from Zend Server (the debugging server) to Zend Studio (client machine)	Zend Studio must be able to accept incoming connections from server. Will work when server and client are in the same LAN. If machines are separated by NAT routers or Firewalls, usage of tunneling or SSH port forwarding is required.
Tunneling	TCP/80 (HTTP, persistent connection) from Zend Studio to Zend Server.	Linux / Mac only. Required to bypass NAT routers or Firewalls between Zend Studio and Zend Server. Connection starts as HTTP but is kept alive after HTTP request ends, and will be used to tunnel debugging traffic.

Studio Settings Auto-Detection	TCP/20080 (HTTP) on localhost on the client's machine	No interaction with the server is required - sent using AJAX to http://localhost:20080 by the user's browser, in order to check Zend Studio configuration before debugging events. Not used if Studio Settings auto-detection is turned off.
--------------------------------	---	--

Zend Java Bridge

Function	Port / Protocol	Comments
Java Bridge	TCP/10001 (Proprietary JB Protocol) on localhost from Zend Server to local JVM	

Installed Components

The following text provides a description of each of the Zend Server Community Edition components that are installed in your environment along with the installation location of each component.

Installation Directories

Not all users decide to install their software in the same location. To reflect this actuality, all paths in this document have been replaced with the following prefix: <install_path>. This represents the location of the installed files. If you used the default settings, the location should be as follows:

- Windows: C:\Program Files\Zend\ZendServer
- Windows 64 bit C:\Program Files (x86)\Zend\ZendServer
- DEB/RPM: /usr/local/zend
- Tarball: /usr/local/zend
- Mac: /usr/local/zend
- For Zend Server Community Edition installation directories, see the Zend Server for IBM i Installation Guide.

Component	L o a d e d	Descripti on	Installation Path	Comments
PHP	+	The Zend certified version of PHP 5.2.x or 5.3.x that includes commonly used and Zend extensions.	Windows: <install_path>\bin RPM, DEB: <install_path>/lib/php/libphp5.so Tarball and Mac: <install_path>/apache2/modules/libphp5.so The extensions for all are under: <install_path>/lib/php_extensions	

Zend Optimizer+	+	Zend's extension for using opcode caching and optimizations for PHP.	Windows: <install_path>\lib\optimizerplus RPM, DEB , Tarball and Mac: <install_path>/lib/optimizerplus	
Zend Guard Loader	+	The Zend Guard Loader for running PHP, encoded with Zend Guard.	Windows: <install_path>\lib\loader RPM, DEB , Tarball and Mac: <install_path>/lib/loader	
Zend Debugger	+	Zend's extension for server side debugging, profiling and code coverage.	Windows: <install_path>\lib\debugger RPM, DEB , Tarball and Mac: <install_path>/lib/debugger	
Zend Cache	+	A Zend extension for PHP data caching and partial PHP output caching.	Windows: <install_path>\lib\datacache RPM, DEB , Tarball and Mac: <install_path>/lib/datacache	

<p>Java Bridge</p>	<p>+</p>	<p>Enables integration of Java libraries and classes within PHP applications.</p>	<p>Windows: <install_path>\lib\jbridge RPM, DEB , Tarball and Mac: <install_path>/lib/jbridge</p> <p>Java Server The Java PHP extension, Java daemon and setup files (not loaded by default). Windows: <install_path>\bin RPM, DEB , Tarball and Mac:</p> <p>PHP Extensions PHP 5.2: <install_path>/lib/jbridge/php.5.2.x/zendbridge.so PHP 5.3 <install_path>/lib/jbridge/php.5.3.x/zendbridge.so Java Daemon - <install_path>/lib/jbridge/jawamw.jar</p>	<p>Note: Requires SUN's JRE 1.4 or later or IBM's Java 1.4.2 or later. 64 bit JRE is not supported. For more information see: SUN Microsystems's website.</p>
<p>Zend Framework</p>	<p>+</p>	<p>Installs Zend's open-source framework for developing Web Applications and Web Services in PHP.</p>	<p>Windows: <install_path>\share\ZendFramework RPM, DEB , Tarball and Mac: <install_path>/share/ZendFramework</p>	<p>This installs libraries containing the Zend framework components.</p>

<p>Oracle Instant Client</p>	<p>+</p>	<p>This installs Oracle OCI (Oracle Instant Client Libraries) lightweight drivers for accessing Oracle Databases.</p>	<p>Windows: <install_path>\bin RPM, DEB , Tarball and Mac: The extension resides with the other extensions, the libraries it depends upon are in <install_path>/lib/</p> <p>Note: "Zend Server provides the Oracle Instant Client 'Basic Lite' package, which only includes English error messages, and support for ASCII, Unicode and Western European character sets. If you need support for other languages and character sets, please install one of the other Oracle Instant Client packages available from Oracle, such as here: http://www.oracle.com/technology/software/tech/oci/instclient/index.html. You should be able to install any recent version of the full Oracle Instant Client libraries as a drop-in replacement for the Lite version provided by Zend, without having to reinstall Zend Server." In Linux, make sure that you place the full version's shared libraries before the Lite version in 'LD_LIBRARY_PATH'.</p>	<p>Required for Oracle database access from PHP.</p>
<p>phpMyAdmin</p>	<p>-</p>	<p>A popular open-source management tool for handling MySQL Database over a Web interface.</p>	<p>Windows: <install_path>\..\phpMyAdmin Tarball : not bundled Mac: <install_path>/share/phpMyAdmin DEB and RPM: the distribution's default location.</p>	<p>Downloaded during installation. Only relevant for MySQL Database users.</p>

<p>IBM DB2 RTCL</p>	<p>-</p>	<p>This installs the IBM DB2 run Time Client libraries for managing Database access.</p>	<p>Windows: user defined location in a separate installer RPM, DEB , Tarball and Mac: IBM DB2 RTCL is not shipped with Zend Server Community Edition and can be downloaded from ftp://ftp.software.ibm.com/software/data/db2/express/</p>	<p>Downloaded during installation. Required for IBM DB2 access from PHP.</p>
<p>MySQL</p>	<p>-</p>	<p>Installs a complete MySQL database on the Web Server.</p>	<p>Windows: <install_path>\.MySQL Mac: <install_path>/mysql RPM, DEB , Tarball: Not Bundled MySQL server's user name and password Linux: Default - "root" Mac OS X: Default- "root" and no password Windows: Default - "root" and no password</p>	<p>Downloaded during installation. Usually the password is "root" for administrators). For more information see: Working with phpMyAdmin to Manage MySQL</p>

<p>Apache 2.2.x Web server</p>	<p>- /+</p>		<p>Windows: <install_path>\..\Apache2 Tarball and Mac: <install_path>/apache2 DEB and RPM: the distribution's Apache package.</p>	<p>Installed only if the option is selected. The alternative is to configure to an existing installation of IIS and then Apache will not be installed.</p>
--	-----------------	--	--	--

Additional Information

The following is additional useful information on Zend Server Community Edition installed components usage.

MySQL for Mac

By default, Zend Server Community Edition for Mac comes with MySQL installed in the following location:
/usr/local/zend/mysql/

Controlling MySQL

You can control the operation of MySQL by using the *mysql.server* in the bin directory:

/usr/local/zend/mysql/bin/mysql.server

The available commands are: start | stop | restart | reload | force-reload | status

Managing MySQL

You may manage the MySQL server by using the phpMyAdmin which is installed by default and may be accessed on Zend Server Community Edition's Dashboard:

- User- By default, 'root'
- Password- None

Note:

Access is possible only from the local machine.

Socket

The socket for MySQL is located in the following location:

/usr/local/zend/mysql/tmp/mysql.sock

Note:

For more information on using the MySQL DB, go to <http://www.mysql.com/> .

Glossary

E

Events: Events are a result of the monitoring component checking your system and collecting information about a specific occurrence that indicates that your environment is displaying uncharacteristic behavior.

G

GUI: Graphical user interface

H

HA: High Availability

J

Job Queue: A simple and manageable system for off-loading tasks from the synchronous user request, to an external, parallel PHP process.

Jobs: Jobs contain information about a specific Job (script to run) that you have set to run in your environment.

R

Recurring Jobs: Jobs that were generated based on a scheduling rule.

S

SC: Session Clustering

SCD: Session Clustering Daemon

Scheduling Rule: An entity defining the periodic re-occurrence of jobs of a specific type.

Z

ZSCM: Zend Server Cluster Manager