Zen Cart® Documentation

# Implementation Guide

## for Zen Cart® Version 2.0.0

| | |
|---|---|
| Document | Implementation Guide |
| Author | Zen Cart® Team |
| Document Revision | Document Rev 1.10.2 |
| Document Revision Date | 07 April 2024 |

## Table of Contents

***1. Introduction ........ 5***

***2. Installation Requirements ........ 5***

**2.1 Before Starting, Ask Yourself These Questions: ........ 5**

2.1.1 Do You Have A Domain? ........ 5
2.1.2 Am I Using A Wireless Network? ........ 5
2.1.3 Are You Using a Personal Firewall on Your Computers? ........ 5
2.1.4 Do You Have A Good Text Editor Program? ........ 5
2.1.5 Do You Have Access To Your Webhosting Control Panel to Create a MySQL Database and User? ........ 5
2.1.6 Do You Have Reliable FTP/SFTP Software? ........ 6

**2.2 Domain Name Requirements ........ 6**

**2.3 Server Hardware Requirements ........ 6**

**2.4 Server Software Requirements ........ 7**

**2.5 Other Installation Requirements ........ 8**

***3. Obtaining the Current Zen Cart® Release ........ 8***

**3.1 Verifying integrity using Hash Keys ........ 9**

**3.2 Patches ........ 9**

**3.3 Updates/Upgrades ........ 9**

**3.4 Notification of New Releases/Updates ........ 9**

***4. Unpacking and Uploading the Application Software Files ........ 9***

**4.1 Tools Required ........ 9**

**4.2 Unzipping/Unpacking ........ 10**

**4.3 Where Do I Upload To? ........ 10**

**4.4 Advanced Method ........ 10**

***5. Pre-Installation Actions ........ 11***

**5.1 New Installations ........ 11**

5.1.1 File/Folder Permissions ........ 11

**5.2 Upgrades ........ 12**

***6. Running the Web-Based Installer ........ 13***

**6.1 New Installs ........ 13**

6.1.1 Introduction ........ 13
6.1.2 Step 1 Welcome and System Inspection ........ 14
6.1.3 Step 2 System Setup ........ 17
6.1.4 Step 3 Database Setup ........ 19
6.1.5 Step 4 Admin Setup ........ 20
6.1.6 Step 5 Setup Finished ........ 22

6.2 Using zc_install to do The Database Upgrade Step of a Site Upgrade .......... 23
6.2.1 Introduction .......... 23
6.2.2 Upgrading configure.php files, if necessary .......... 23
6.2.3 Step 1 Welcome Screen and System Inspection .......... 24
6.2.4 Step 2 Version-Upgrade Checkboxes .......... 24
.......... 25
6.2.5 Step 3 Database-Upgrade Step Finished .......... 26
*7. Post-Installation Actions .......... 27*
7.1 Changing The Admin Directory Name for Security (By-Obscurity) .......... 27
7.2 Enabling SSL for HTTPS in your Admin .......... 27
7.3 Setting Directory and File Permissions .......... 27
7.4 Removing the Installation Directory .......... 28
7.5 Blocked Administration Access .......... 28
7.6 Removing Unnecessary Directories .......... 28
*8. Accessing the Administration Panel and Configuring Administrative Users and Passwords .......... 29*
8.1 Introduction .......... 29
8.2 Administrator Access to Credit Card Numbers .......... 29
8.3 Administrative User Access and PA-DSS requirements .......... 29
8.4 Users .......... 30
8.5 Profiles .......... 31
8.6 Admin Activity Logs .......... 32
8.6.1 Daily Log Review – Important Things To Monitor .......... 32
8.6.2 Review or Export Logs .......... 33
8.6.3 Purge Log History action .......... 34
8.6.4 PA-DSS Logging – Technical Details .......... 35
8.6.5 Centralized Logging .......... 35
*9. Code Customization, Addons, and Plugins .......... 37*
*10. Engaging 3rd-Party Consultants or Programmers .......... 38*
10.1 Webstore “Admin”/Backend access .......... 38
10.2 FTP Access .......... 38
10.3 Webhosting Account's Control Panel access .......... 38
10.4 Secure use of customer database and website files .......... 38
10.5 Two-Factor Authentication .......... 39
*11. Removing Old Non-PCI-Compliant Data .......... 40*
11.1 Removing Old Credit Card Data From Database Records .......... 40
11.2 Suggested Procedure For Secure Erasure of Old CHD data .......... 40
*12. Network Diagram .......... 42*
*13. Dataflow Diagram .......... 43*

*14. Notes about PA-DSS Compliance ........................................ 44*

14.1 Cardholder Data ........................................ 44

14.2 Cryptographic Keys and Key Management ........................................ 44

14.3 Protocols, Services, Dependent Software and Hardware ........................................ 44

14.4 Settings sensitive to PCI compliance ........................................ 45

*15. Additional Requirements for PA-DSS Compliance ........................................ 46*

15.1 Consequences of altering the system to store cardholder data ........................................ 46

15.2 Default Accounts ........................................ 46

15.3 Strong Authentication Controls ........................................ 47

15.4 Secure Access ........................................ 47

*16. Appendices ........................................ 49*

16.1 MySQL Root Password Reset ........................................ 49

16.2 Password Security in Zen Cart® ........................................ 49

16.3 Wireless (WiFi) Networks ........................................ 49

*17. Implementation Guide Changelog ........................................ 51*

.

# 1. Introduction

This Implementation Guide is meant to help you not only with important subjects related to installing or upgrading the Zen Cart® application but also to understand the issues related to securely implementing Zen Cart® in a manner that is PA-DSS compliant.

**PA-DSS**

It is a requirement of the PA-DSS that you follow the instructions in this Implementation Guide when installing or upgrading your Zen Cart® application.

**Note also, that this guide is written for the v1.5-v2.x series of Zen Cart®** unless otherwise noted.

NOTE: Although some screenshots in this guide may mention older Zen Cart versions, they are equally applicable.

# 2. Installation Requirements

## *2.1 Before Starting, Ask Yourself These Questions:*

### 2.1.1 Do You Have A Domain?

If No, stop and refer to section 2.2 for information about registering a domain for your website.

You need a domain name to host your webstore on a webserver.

### 2.1.2 Am I Using A Wireless Network?

If you are using a wireless network to access your online store, it MUST be configured securely. That means securing your wifi network with a strong complex password, and NOT using the one provided by default when resetting it or unboxing it. See the Appendix of this manual for additional requirements for properly securing your wireless network.

### 2.1.3 Are You Using a Personal Firewall on Your Computers?

For security, you should always use a personal firewall when accessing any online systems, especially your own online store's administration area.

### 2.1.4 Do You Have A Good Text Editor Program?

If no, stop … you will need a good Text Editing application such as Visual Studio Code, Sublime Text, Notepad++, UltraEdit, BBedit, Kedit, or maybe a more advanced IDE like PhpStorm.
This text editor application will be used for modifying the files if you customize the Zen Cart® software.

**NOTE: Do NOT use cPanel for editing files, nor Microsoft Word or other software designed for writing documents… you want a nice clean text-focused editor which doesn't add extra "junk" into the files, including "extra characters" and "fancy quotes" instead of straight quotes, etc.**

### 2.1.5 Do You Have Access To Your Webhosting Control Panel to Create a MySQL Database and User?

BEFORE YOU PROCEED TO INSTALLATION, make sure you have access to a MySQL database, and username/password to that database. You may need to create the database using your webhosting account's control panel. Contact your webhosting company for assistance. Zen Cart® cannot create the database for you. You must use a strong secure password.

(You need to grant the following permissions to your MySQL database user: SELECT, INSERT, UPDATE, DELETE, CREATE, ALTER, INDEX, DROP. If you must choose from more generic options such as with an hSphere host, this would be equivalent to "dba" access or at least "read/write".)

In a fully PCI-Compliant hosting setup, the database server would be behind a DMZ, on a separate server other than the main webserver. In this case the DMZ firewall will need to have port 3306 open for communication between the two servers' IP addresses. It may also be necessary to grant additional privileges to the database username you created in this step. Your server administrator (hosting company) can assist you with these configuration details.

You will also need to know the appropriate "host" address for the database server. If it is "localhost" then that means the database is on the same server as the webserver engine, which is usually not a PCI-Compliant configuration. Your hosting company / server administrator can provide you with the correct "host name" or IP address for the database server. You will use this information during initial setup via the zc_install script explained in the following sections.

### 2.1.6 Do You Have Reliable FTP/SFTP Software?

If No, stop. You need to obtain a reliable FTP software package such as FileZilla, WinSCP, or Transmit. This application is used to transfer files between your computer and your webserver.

("FTP" is a very common website acronym for "File Transfer Protocol")
("webserver" refers to the computer on the internet where you have your site/domain hosted)

You should use an FTP program capable of connecting in secure SFTP mode (or FTP-with-Implicit-TLS) when working with your website. Tutorials on how to use FTP/SFTP are available online from the vendor of your FTP software, or generically from any number of online reference websites.

*Whenever anyone mentions "FTP", you should use SFTP or FTP-with-Implicit-TLS instead.*
*This includes any subcontractors you hire to work on your website for you.*

**Why SFTP vs FTP?**
Plain FTP mode transfers files in plain-text over the internet, whereas SFTP ("Secure FTP") uses a secure encrypted connection for doing the transfer. This is important since the files you are transferring to/from your server may include sensitive information. Using an SFTP connection will cause your data to be encrypted as it is transferred, thus protecting it from prying eyes.
Many FTP programs capable of SFTP are available for free or for a modest fee from various online vendors. One very popular such application is FileZilla, which works on both Windows® and MacOS®. Some people prefer the more advanced look/feel of paid applications. The choice is yours.

NOTE: If your hosting company provides a file-upload service or FTP app that runs inside your browser, we strongly recommend that you DO NOT use that for uploading large amounts of files to your server. They may work for individual files, but are seldom reliable when uploading large numbers of files such as a fresh install of Zen Cart, since they will often timeout without showing any error, and leave you with a damaged set of files which operate unpredictably. Incomplete uploads are the most common cause of problems on new sites.

## *2.2 Domain Name Requirements*

You will need a registered domain name, connected to your webhosting account at your webhosting company. If you need to register a domain name, we recommend using https://hover.com

Temporary use of merely an IP address may work during initial installation, but to actually run your shop will require use of a domain name. If your domain is brand-new and is pending initial setup by your hosting company, a temporary domain name may be supplied to you so you can get started without waiting.
Changing the domain-name in Zen Cart® after initial setup will require manual editing of your configure.php files. An article on making such changes can be found at https://docs.zen-cart.com

## *2.3 Server Hardware Requirements*

Zen Cart® itself does not "require" any particular hardware, as long as the hardware you use for your hosting service supports the software requirements that follow.

However, you should be aware that some hardware configurations such as inadequate server RAM, slow server hard drives, excessively restrictive firewalls, etc, can adversely affect the operation of the Zen Cart® application.

## *2.4 Server Software Requirements*

See https://docs.zen-cart.com/user/first_steps/server_requirements/ for latest details.

Technically speaking, Zen Cart® v2.0.0 will work with the following **minimum** requirements:

PHP version >= 8.0, 8.1, 8.2, 8.3
MySQL version > 5.7.8+ or MariaDB 10.2.7+
Apache version > 2.2 or 2.4

However, for PA-DSS compliance, you must use **the latest stable versions** of PHP, MySQL and Apache. As of the date of this writing, the **recommended** versions for PA-DSS compliance are:

PHP version >= 8.3.4 or 8.2.17 (ref: https://php.net/ )

MySQL version >= 8.0.36 (ref: https://dev.mysql.com/downloads/mysql )

Apache version >= 2.4.59 (ref: https://httpd.apache.org )

*Note: While **we recommend the use of Apache as your web server software**, Zen Cart® will also work with other Web Servers (e.g. nginx), however some security features will cease to work. Additional information on this is provided in the next section regarding .htaccess.*

*The Zen Cart® PA-DSS certification is performed on Linux+Apache, so using it on nginx or IIS invalidates the certification for your site, and will require you to self-certify using your own Qualified Security Assessor.*

### *PHP Extensions*

You will also need to ensure that your PHP version has the following modules installed:

- cURL – Required for some shipping and payment methods. (eg: sudo apt-get install php-curl)
- OpenSSL support – Usually this is compiled into PHP and cURL upon install of PHP

### SSL Certificate for HTTPS

Unless you will have no customers accessing your site via the internet, you will want an SSL certificate added to your hosting account. A "shared" certificate may work, but dedicated is preferred as it is a more seamless experience for your customers and is much easier to configure. An SSL Certificate is used to encrypt sensitive data when transmitting from the customer's browser to your site

### Secure TLS (particularly support for TLS 1.2 and 1.3)

TLS is used to encrypt transmissions from your site to payment gateways or other systems like shipping-quote services, 3rd-party order fulfilment services, etc. It's important that your webserver be properly configured for TLS to be used with CURL, and not support old/insecure versions of SSL/TLS.

### SFTP or FTP-with-Implicit-SSL/TLS

You will also need to ensure that your hosting service allows you to use SFTP or FTP-with-implicit-TLS for transferring files to/from your hosting server. See section 2.1.6 above for more detail.

**MySQL**

This has already been accommodated since Zen Cart v1.5.7, but it is mentioned here for compatibility reasons:
As of MySQL 5.6 the NO_ZERO_DATE and NO_ZERO_IN_DATE modes are deprecated, and as of MySQL 5.7 they are included in STRICT_ALL_TABLES and STRICT_TRANS_TABLES modes. If your database contains fields with 0000-00-00 in the data or the schema, you may need to ensure that your MySQL server is configured with either STRICT_ALL_TABLES or STRICT_TRANS_TABLES mode … or make appropriate changes to your PHP files and the data already in your database tables, and your database schema. Some strict GROUP BY rules may also need addressing.

## *2.5 Other Installation Requirements*

**PA-DSS**

.htaccess requirements

Zen Cart® uses Apache .htaccess files to better protect some directories for security purposes. You should ensure that your Apache settings allow for the use of .htaccess files on your Web server (most do). If you are unsure please check with your Hosting provider.

Specifically, Apache must be configured with AllowOverride set to either '**All**' or at least both '**Limit**' and '**Indexes**' parameters, and preferably the '**Options**' parameter as well.

```
eg:  <directory /name-of-your-document-root-folder-here>
         AllowOverride All
#          Options -Indexes    # uncomment this to prevent all directory-listings
     </directory>
```

IIS or nginx

If you are not using Apache as the web server (e.g. you are using IIS or nginx) then you should take steps to protect the directories in a similar manner to the .htaccess files Zen Cart® suggests.  Inspect each .htaccess file in the distribution files and build corresponding rules appropriate for your webserver engine.

SSL Certificates for HTTPS

Your web server must be able to serve pages using SSL encryption and you should have an SSL certificate correctly installed for your domain. If you do not have SSL or are unsure, then once again you must confer with your Hosting provider.

TLS - Your webserver must be configured to use only secure versions of TLS.

Secure FTP

Your hosting service must also offer the ability to use SFTP or FTP-with-implicit-TLS for transferring files to/from the server.

# 3. Obtaining the Current Zen Cart® Release

The current release is obtainable via Github: https://github.com/zencart/zencart/releases
or via https://www.zen-cart.com/latest_https. The release is provided as a .zip file. For PCI Compliance, you should verify to be sure your browser is actually downloading via HTTPS.

### *3.1 Verifying integrity using Hash Keys*

Hash keys are a way of checking the validity of a zip file. We provide SHA256 hashes for the current release. The validation hashes can be seen below the download link on the home page of the Zen Cart® support website at https://www.zen-cart.com.

There is also information on how to use and check the hash keys in the following FAQ article: https://docs.zen-cart.com/user/installing/validate_sha/

### *3.2 Patches*

The normal distribution of updates is to release a new version with fixes included. In the rare occasion that a separate .zip file is released as a patch, the same hash-verification described above should be performed on the zip file before unzipping to install it on your website. Again, these zips will be released on Github.com.

### *3.3 Updates/Upgrades*

Updates must be performed manually. There is no built-in automated facility for upgrading the software. (The software can automatically bring your database content up-to-date when upgrading, but this ONLY handles the database data, and not the PHP files which run the actual software logic.)

**PCI-DSS and PA-DSS** requires that we specifically inform you that "the payment application does not receive automatic updates" and further requires that when you or your agent installs updates that they do so in a secure PCI-DSS and PA-DSS compliant manner, and that access for said purpose is limited to the duration of time required to perform said tasks and access is removed thereafter. See section 10 of this guide for more guidance.

### *3.4 Notification of New Releases/Updates*

There are numerous ways to become aware of a new release. But all updates are obtained manually by the merchant themselves, and transmitted to the server themselves. Installation is not automated.

- Notification in your Admin console. There is a button in the upper right corner of your admin console which can be used to check for new versions. In some cases this button may automatically display a message that a new version is available. This is done by querying the Zen Cart® versioning server to determine what the latest version number is. It does not transmit any specific information to the Zen Cart versioning server. It simply compares your version with the latest version available, and informs you if there is a difference.
- Notification posts on the Zen Cart® support site. You can subscribe yourself to these notices by registering on the Zen Cart® support site and clicking the Subscribe option (link) in the News And Announcements section.
- Alerts on Github.com or the Zen Cart® site, and the @ZenCart twitter feed.

## 4. Unpacking and Uploading the Application Software Files

### *4.1 Tools Required*

Before you can unpack and upload the files to your server, you will need two important tools:

- **An "unzip" utility**, such as 7-zip, WinZip, unRar, BetterZip, etc.
  Many unzip utilities are available for free, for various computer operating systems. Choose one that suits your computer best.
  IMPORTANT: When you unzip, you need to ensure that your unzip program retains the embedded file-structure. Usually that setting is properly "on" by default, but if it prompts you saying "xxxxxx file already exists – overwrite?" or similar, then it's most likely only extracting the "files" and not also the "folders. In that case you'll need to make appropriate adjustments to your unzip application settings before you can properly unzip the Zen Cart® files.

- **An FTP application capable of SFTP**.
  See section 2 for more information on FTP and SFTP.
  You must use strong secure passwords for your FTP/SFTP access to your webserver.

## 4.2 Unzipping/Unpacking

The Zen Cart® release is packaged as a zip file. You will need to unzip this file using an appropriate tool/application on your local computer, before uploading it to your web server.

NOTE: When you unzip the file it will create a folder, which will be named something like

zen-cart-v2.x.x-xxxxx

NOTE: You should not upload this "zen-cart-v2.x.x.-xxxxx..." directory to your web server, but rather the **contents** of that directory.

## 4.3 Where Do I Upload To?

The directory on your web server that you need to upload to is usually specific to the hosting platform you are using, and you will need to check those details with your host. Usually this will be the "public_html" or "www" or "httpdocs" or "http" folder, depending on how your hosting company has configured their server.

Basically, in your FTP application, look for a "public_html" or "www" or "htdocs" or "httpdocs" or "wwwroot" folder. These are the common folder names for what is referred to as the "webroot", which is where all website content is served from.

Your Zen Cart files (or **any** files to run your website, for that matter) need to be under that folder. If they're not, then you're going to encounter "not found" errors … because the content is not found!

## 4.4 Advanced Method

You can optionally save some time by uploading the zip file directly to your server (using FTP etc.), and unpacking it in situ if your hosting company provides you a means of unzipping files on the server side. Talk to your hosting company about this.

# 5. Pre-Installation Actions

## 5.1 New Installations

Before running the Zen Cart® installer you will need to address the following.

1) **MySQL Database**
   Ensure that you have created an empty MySQL database (and corresponding username and strong secure password) for use with Zen Cart®.
   How you do this depends on your hosting configuration. Usual methods include using cPanel or phpMyAdmin. If you are unsure, please check with your hosting company.
   **Strong secure passwords are required for PCI/PA-DSS compliance.**
   **For PCI compliance, you must NOT use the "root" account for your database credentials.**
   **Also, see section 2 for notes about MySQL configuration details needed for properly configuring it in a secure DMZ scenario, as required for PCI compliance.**

2) **configure.php files**
   Two files need to be created on the server. These are the configure.php files that will contain important information to identify the settings of your particular server and the location of the files that you just loaded. After they have been created, you will then need to change the permissions on these files.

For a new install, the simplest way to do this is to rename these two files:

- `/includes/dist-configure.php`
  to `/includes/configure.php`
- `/admin/includes/dist-configure.php`
  to `/admin/includes/configure.php`

3) Set file and folder permissions as explained in the next section.

### 5.1.1 File/Folder Permissions

Changing permissions on these files can often be done using your FTP application (unless you are hosted on a Windows server). Or, you can use a File Manager console provided by your hosting company's control panel. If you need help understanding the concepts of file permissions and some general guidance on making these changes, consult this tutorial article: https://docs.zen-cart.com/user/first_steps/how_do_i_install/

*In the following instructions, the "INSTALL_DIRECTORY" is the "webroot" folder into which you uploaded your Zen Cart files as explained in the previous section.*

4) Ensure that the INSTALL_DIRECTORY/includes/configure.php and INSTALL_DIRECTORY/admin/includes/configure.php are writeable. These file needs to be <u>writeable during the installation only</u>, so that the installer may save some important settings to them. Once the installation is complete you will need to make the files read only again.

5) Ensure that the directory INSTALL_DIRECTORY/cache is writeable. This directory needs to be writeable, as the Zen Cart® application may need to store some important files here (ie: Session and Cache data).

6) Ensure that the directory INSTALL_DIRECTORY/logs is writeable. This directory needs to be writeable, as the Zen Cart® application may need to store some important files here (ie: namely PHP error logs and debug output).

7) Ensure that the directory INSTALL_DIRECTORY/images is writeable. The images directory needs to be writeable to allow for the uploading of product and other images that you will use in your store. Your admin backend will be used for uploading product/category images here.

8) Ensure that the directory INSTALL_DIRECTORY/pub is writeable. The pub directory needs to be writeable to allow for the downloading of any virtual products that you sell, eg. Media files (mp3/mp4/pdf). *If you do not intend to ever sell these types of products then this directory does not need to be writeable.*

9) Ensure that the directory INSTALL_DIRECTORY/admin/images/graphs is writeable. This directory needs to be writeable to allow for the creation of graph images that represent the statistics for any banners you may serve. Ignore this if you're not using Banner Manager.

Note: After installation is complete, you will need to change some permissions again. More information is given later in this document *(See the section on **Post-Installation Actions**.)*

## 5.2 Upgrades

Upgrading your site between most versions is only as complex as the amount of customization you've made to your site. You'll need to consider upgrades to any plugins/addons you've installed, as well as changes you've made to any core files, and any changes made by overriding files with custom versions of those files.

A proper FULL UPGRADE between versions is essentially a rebuild of your current site using the new version. While that may sound daunting, it doesn't need to be. There are automated tools such as WinMerge, Araxis Merge or BeyondCompare which can help you quickly identify all the changes you made to your old site, so that you can easily re-make those changes to your new site.

Always read the "Whats New" and "Changed Files" logs located at https://docs.zen-cart.com/release or in the /docs/ folder of the new version's release-zip file, because specific upgrade instructions, if any, for each version will be listed there.

**Warning: You should never directly upgrade a live site. Always test it separately first!**

Your upgrade should be staged using a separate folder+database+phpVersion on your server, and then migrated to the live site only after you've tested to confirm that no problems have been introduced in your implementation of the upgrade. Consider using a subdomain for all this!
***A simplified guide for upgrading is found at*** https://docs.zen-cart.com/user/upgrading/detailed_upgrading/

Please see https://www.zen-cart.com/upgrades for guidance regarding upgrading your site.

# 6. Running the Web-Based Installer

## *6.1 New Installs*

### 6.1.1 Introduction

To run the Zen Cart® installation wizard, you will need to use your browser to access the web server where you installed Zen Cart®. The installation wizard is accessed from the folder /zc_install.

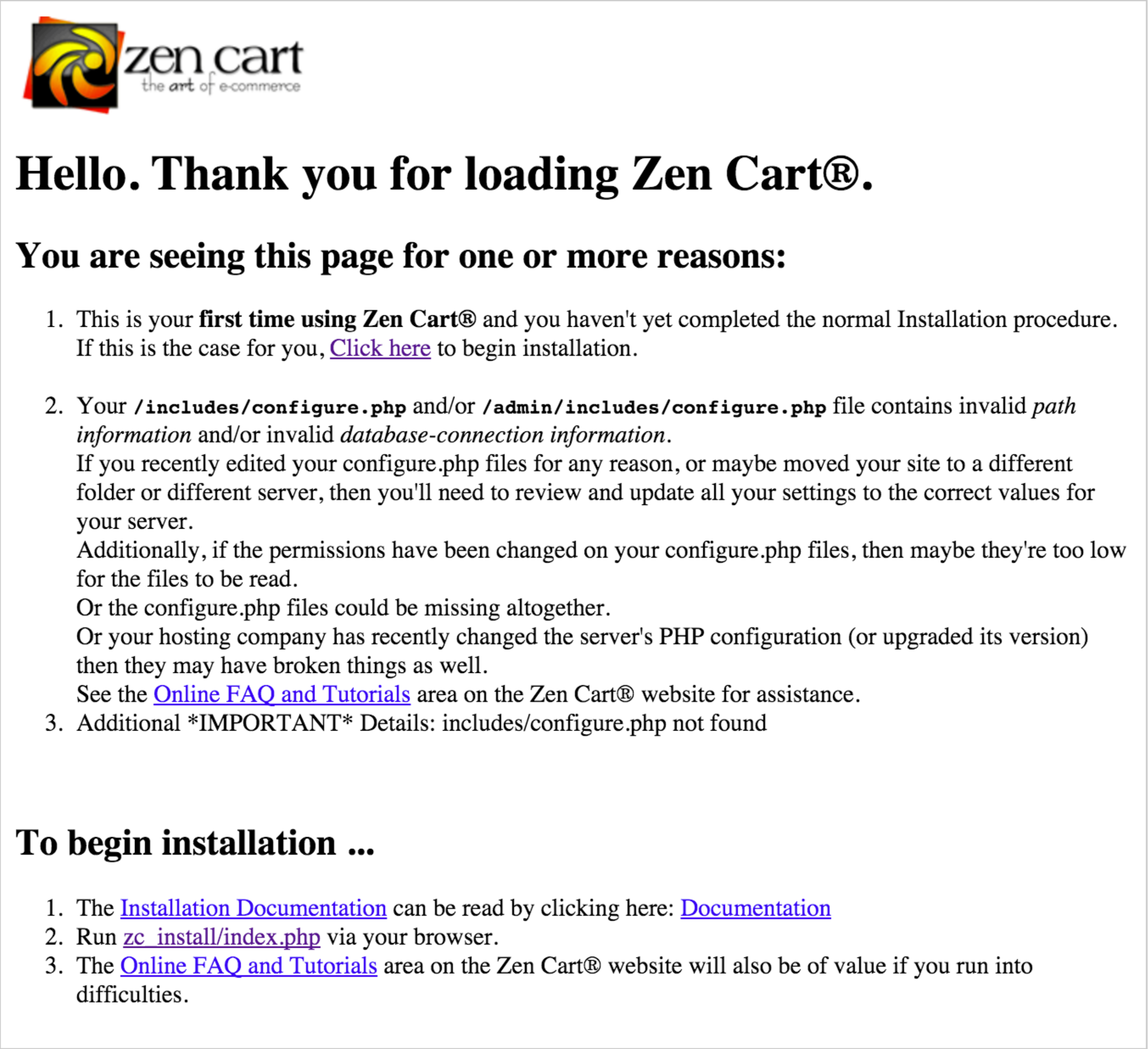
So if you have set your web server up to be accessed as https://www.MY_DOMAIN.com/

Then you would need to set your browser url to https://www.MY_DOMAIN.com/zc_install/

(If you get "not a secure site" errors when accessing via https, you should sort out the https/SSL issue with your hosting company before proceeding. While you "could" proceed via http instead of https, you will need to update configure.php files later to ensure using https, so it's best to fix the https access first before proceeding here. Most hosting companies provide free SSL/HTTPS nowadays, and offer easy configuration for it. Contact your hosting company for help.)

**PA-DSS**
To comply with PA-DSS you **MUST** enable SSL (ie: use HTTPS) on your webserver.

Note. If you attempt to load the URL where your store will ultimately reside, before running the Installation wizard, you may get a blank page, or a screen that looks like the screenshot below.


# Hello. Thank you for loading Zen Cart®.

## You are seeing this page for one or more reasons:

1. This is your **first time using Zen Cart®** and you haven't yet completed the normal Installation procedure. If this is the case for you, Click here to begin installation.

2. Your **`/includes/configure.php`** and/or **`/admin/includes/configure.php`** file contains invalid *path information* and/or invalid *database-connection information*.
   If you recently edited your configure.php files for any reason, or maybe moved your site to a different folder or different server, then you'll need to review and update all your settings to the correct values for your server.
   Additionally, if the permissions have been changed on your configure.php files, then maybe they're too low for the files to be read.
   Or the configure.php files could be missing altogether.
   Or your hosting company has recently changed the server's PHP configuration (or upgraded its version) then they may have broken things as well.
   See the Online FAQ and Tutorials area on the Zen Cart® website for assistance.
3. Additional *IMPORTANT* Details: includes/configure.php not found

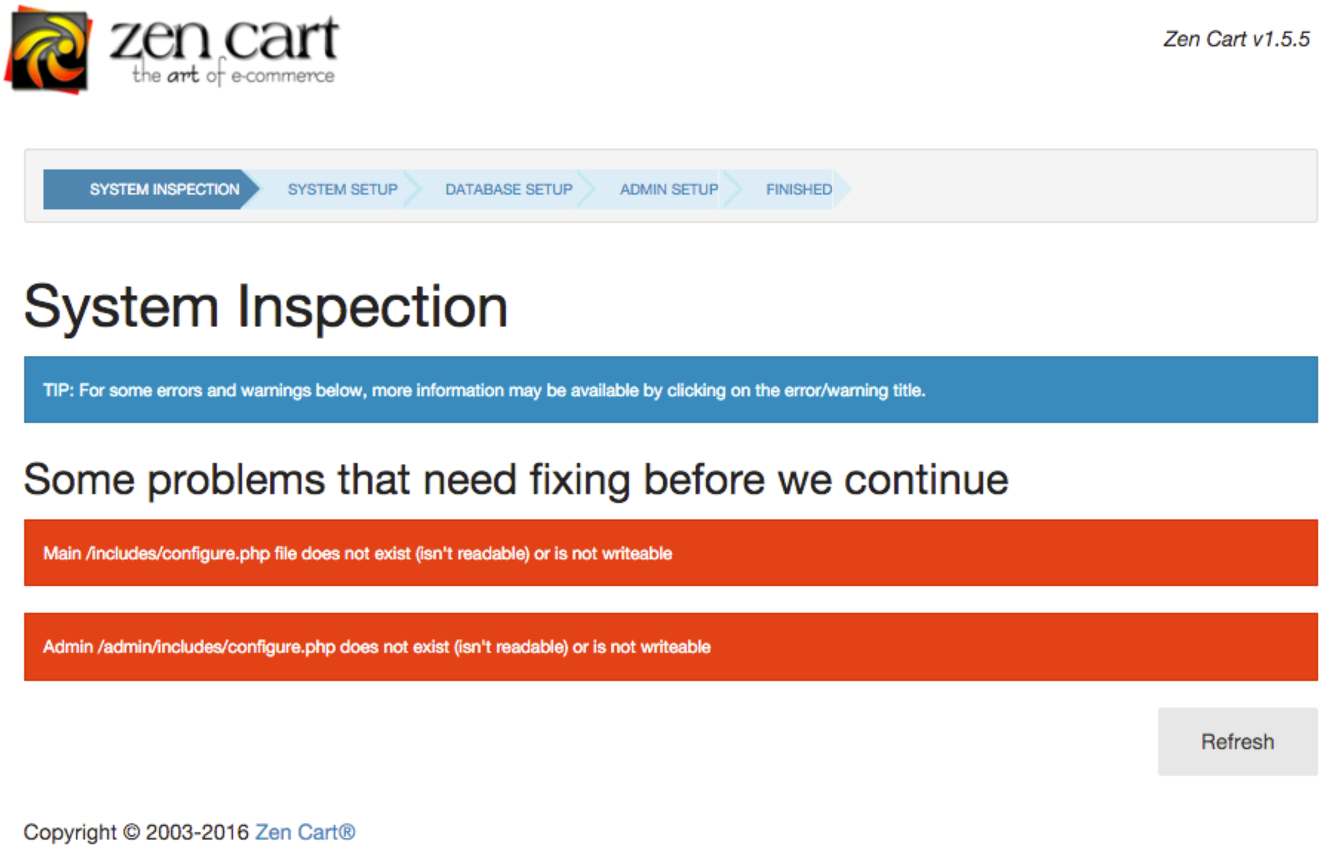
## To begin installation ...

1. The Installation Documentation can be read by clicking here: Documentation
2. Run zc_install/index.php via your browser.
3. The Online FAQ and Tutorials area on the Zen Cart® website will also be of value if you run into difficulties.

### 6.1.2 Step 1 Welcome and System Inspection

The System Inspection page checks that various required web server components exist, and permissions are set correctly for the Zen Cart® application to function correctly. You should review all items on this page and take any necessary actions to correct problems highlighted here, before continuing.

The following screenshot shows the warning about missing configure.php files (which you create manually):



System Inspection

TIP: For some errors and warnings below, more information may be available by clicking on the error/warning title.

Some problems that need fixing before we continue

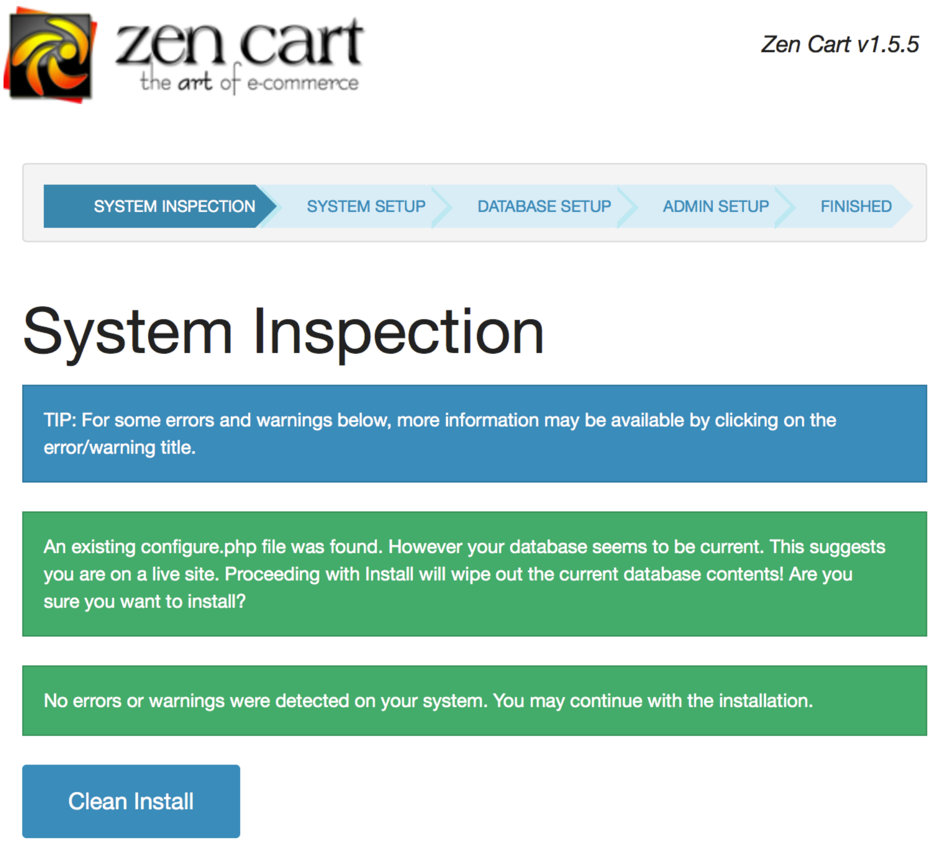
Main /includes/configure.php file does not exist (isn't readable) or is not writeable

Admin /admin/includes/configure.php does not exist (isn't readable) or is not writeable

Copyright © 2003-2016 Zen Cart®

Once you've created those missing files and made them writable, click Refresh and the inspection will run again, showing you any necessary updated suggestions.

When it is possible to proceed, the “Clean Install” button will appear, as shown here:



System Inspection

TIP: For some errors and warnings below, more information may be available by clicking on the error/warning title.

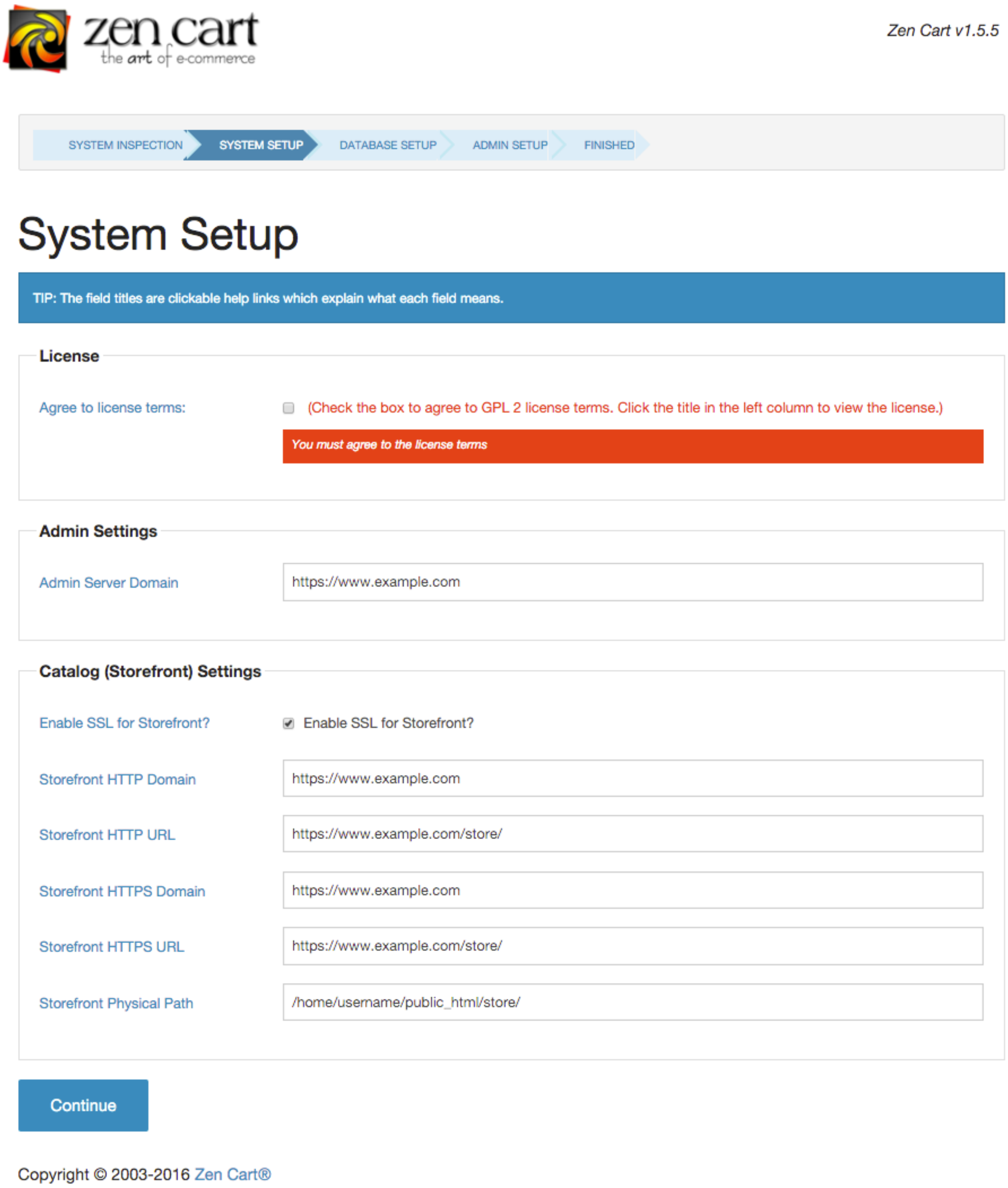
An existing configure.php file was found. However your database seems to be current. This suggests you are on a live site. Proceeding with Install will wipe out the current database contents! Are you sure you want to install?

No errors or warnings were detected on your system. You may continue with the installation.

The image above has an indicator about some pre-existing configure.php files. If this is your first install, you won't see the first green box above.

If you already have a Zen Cart store set up in this database, you may also see an “Upgrade” button here. If you click “Clean Install” here, you will erase any Zen Cart data already in that database.  For instructions on doing upgrades, see the chapter on upgrading, later in this guide.

### 6.1.3 Step 2 System Setup



Zen Cart v1.5.5

System Setup

TIP: The field titles are clickable help links which explain what each field means.

License

Agree to license terms: □ (Check the box to agree to GPL 2 license terms. Click the title in the left column to view the license.)

*You must agree to the license terms*

Admin Settings

| | |
|---|---|
| Admin Server Domain | https://www.example.com |

Catalog (Storefront) Settings

| | |
|---|---|
| Enable SSL for Storefront? | ☑ Enable SSL for Storefront? |
| Storefront HTTP Domain | https://www.example.com |
| Storefront HTTP URL | https://www.example.com/store/ |
| Storefront HTTPS Domain | https://www.example.com |
| Storefront HTTPS URL | https://www.example.com/store/ |
| Storefront Physical Path | /home/username/public_html/store/ |

Copyright © 2003-2016 Zen Cart®

### 6.1.3.1 Agree to license terms

Click the “Agree to license terms” text on the left column to view the GPL2 license terms.
Then check the checkbox to indicate your agreement to the license.

### 6.1.3.2 Admin Server Domain

Enter the URL to your store here. It should have been auto-detected for you. But if your site has SSL capability, you should put the SSL URL here.
Additionally, if your SSL is actually “shared SSL” which uses a URL different from your store's actual domain name, enter that shared-SSL URL here.

### 6.1.3.3 Enable SSL for Storefront?

This determines whether Zen Cart will use HTTPS for the storefront side of your store to automatically encrypt communications on pages which collect sensitive data.
If you don't have an SSL certificate enabled in your hosting service yet, uncheck the box.
You can enable it later manually by following this tutorial: https://docs.zen-cart.com/user/installing/enable_ssl/

*NOTE: To use SSL, Zen Cart® relies completely on you supplying valid/working https:// URLs!*
*If your site doesn't have working SSL yet, then your Zen Cart® store will be broken until you get SSL, or disable it.*

**PA-DSS**
To comply with PA-DSS you **MUST** enable SSL (ie: use HTTPS).
When enabled here, Zen Cart® will automatically activate SSL on storefront pages which need it (ie: checkout, my account, login, etc) as long as the option is enabled here.
Additionally, built-in payment modules which are capable of accepting credit cards directly on your site will NOT function if you do not have SSL capability available and enabled in Zen Cart®.

### 6.1.3.4 Storefront HTTP Domain

This is the URL that will be used to access your store. Again, **auto-detection should have chosen the correct setting** and you should only change it if it is incorrect.

### 6.1.3.5 Storefront HTTP URL

If your store is located in a subdirectory within your site, then add that subdirectory here. Else just use your domain name. **This is normally auto-detected for you and no changes necessary.**

### 6.1.3.6 Storefront HTTPS Domain

This is the domain name for your store, but starts with **https://** … but if you're using “shared-SSL” then you'll need to put the proper shared-SSL domain here. Otherwise same as Storefront HTTP Domain.

### 6.1.3.7 Storefront HTTPS URL

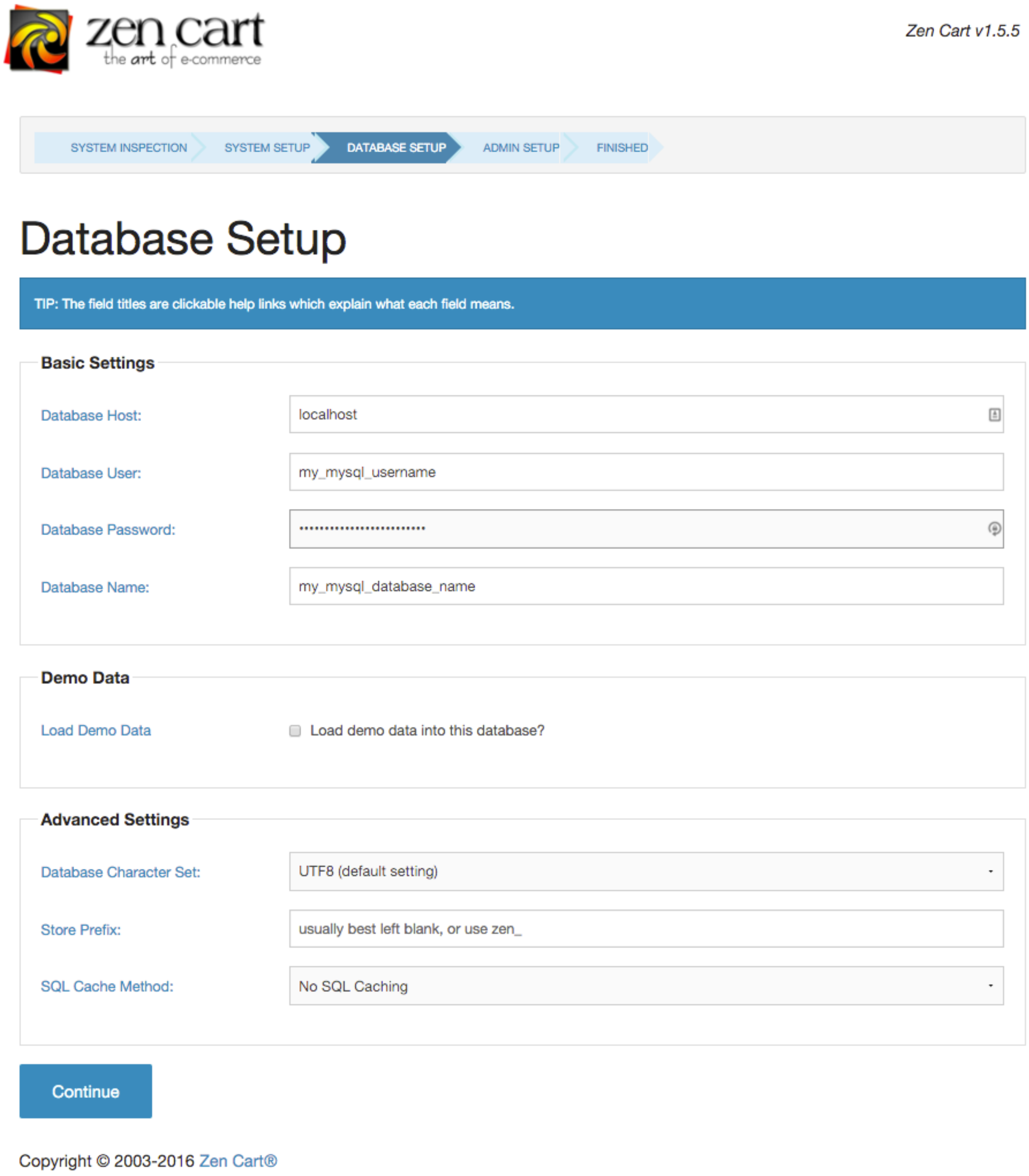
If you're using shared SSL, then sometimes there is a need to add a username or subdirectory to the end of the Shared SSL URL supplied by your hosting company. Enter that here. Otherwise use the same as Storefront HTTP URL.

### 6.1.3.8 Storefront Physical Path

This is the location of the Zen Cart® application files on the hard drive of your server. **Generally the system will auto-detect this**, and you should only change it if the auto-detection has not chosen the correct path.

### 6.1.4 Step 3 Database Setup

Click the title words in the left-column for more detailed information. See section 2 for reference.



# Database Setup

TIP: The field titles are clickable help links which explain what each field means.

**Basic Settings**

| | |
|---|---|
| Database Host: | localhost |
| Database User: | my_mysql_username |
| Database Password: | •••••••••••••••••••••••••• |
| Database Name: | my_mysql_database_name |

**Demo Data**

Load Demo Data ☐ Load demo data into this database?

**Advanced Settings**

| | |
|---|---|
| Database Character Set: | UTF8 (default setting) |
| Store Prefix: | usually best left blank, or use zen_ |
| SQL Cache Method: | No SQL Caching |

Copyright © 2003-2016 Zen Cart®

## 6.1.5 Step 4 Admin Setup

### *6.1.5.1 Admin Superuser Name:*

This is the user name used to initially access your store's administration panel. This user has access to all of the functionality of the Administration panel. You can set up additional users with restricted permissions once the application has been installed.

### *6.1.5.2 Admin Superuser Email*

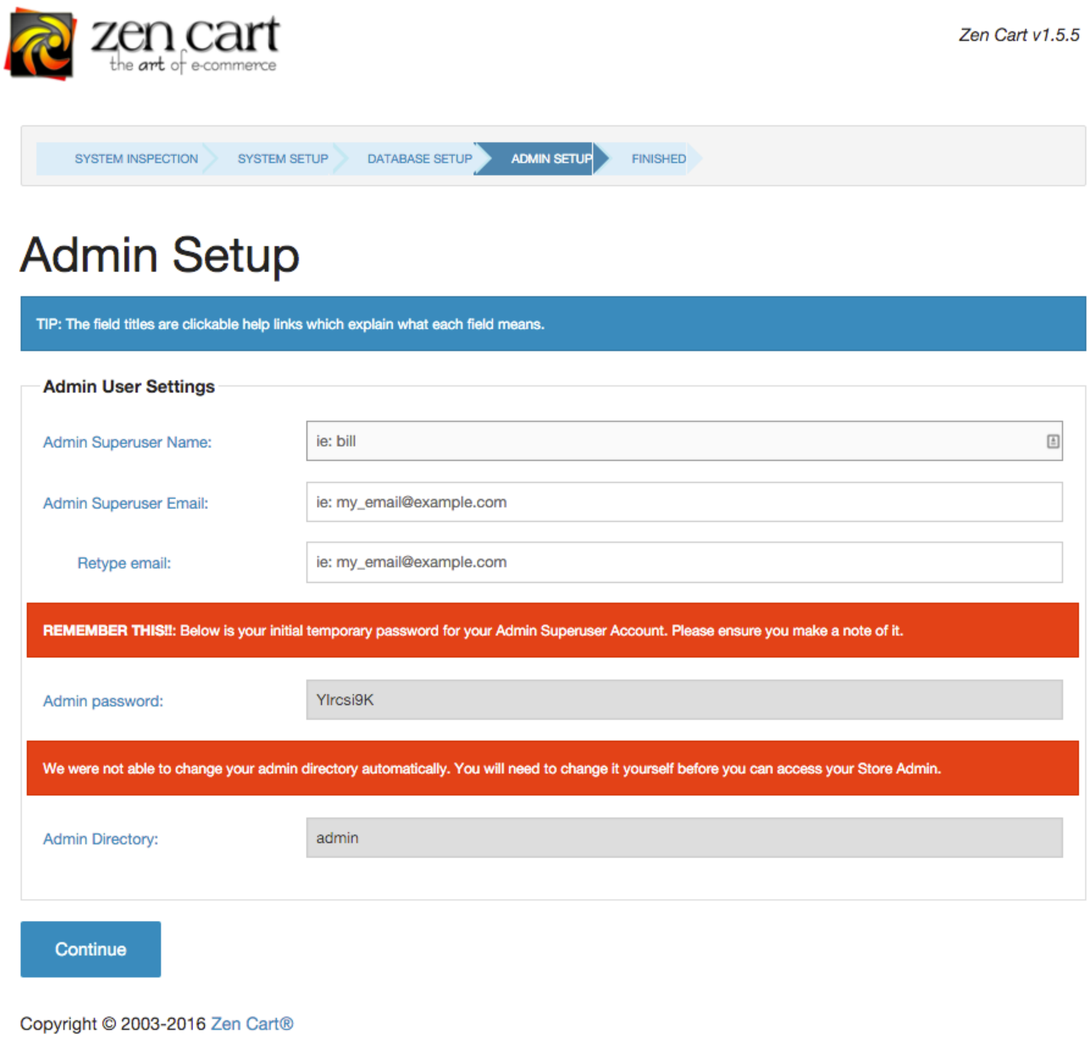
This is the email address of the initial Administrator, and may be used for sending password resets or testing outgoing email newsletters etc.  Type it again to confirm you didn't make any typos.

### *6.1.5.3 Admin Password*

**This is an auto-generated TEMPORARY password, and you will be prompted to set your own password after using this temporary password for your first login to your admin area**.

### *6.1.5.4 Admin Directory*

The installer will attempt to automatically rename your admin folder to something other than "/admin". **It will show you here what that new foldername is. You will use that foldername to login to your admin panel on the next screen.**



# Admin Setup

TIP: The field titles are clickable help links which explain what each field means.

**Admin User Settings**

| | |
|---|---|
| Admin Superuser Name: | ie: bill |
| Admin Superuser Email: | ie: my_email@example.com |
| Retype email: | ie: my_email@example.com |

**REMEMBER THIS!!:** Below is your initial temporary password for your Admin Superuser Account. Please ensure you make a note of it.

| | |
|---|---|
| Admin password: | YIrcsi9K |

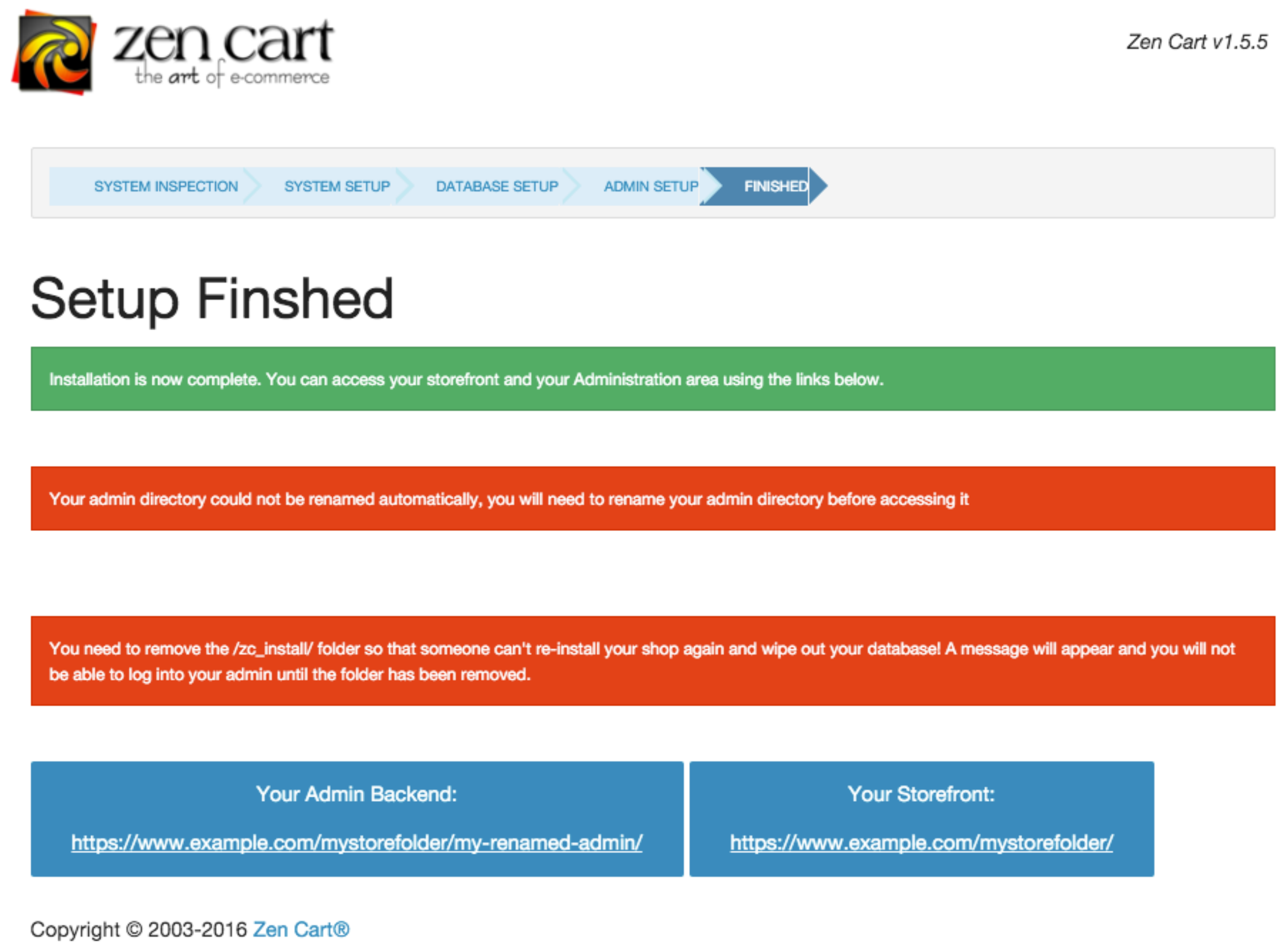
We were not able to change your admin directory automatically. You will need to change it yourself before you can access your Store Admin.

| | |
|---|---|
| Admin Directory: | admin |

Copyright © 2003-2016 Zen Cart®

### 6.1.6 Step 5 Setup Finished

Now that setup is finished, a number of post-installation instructions are presented for you to follow. Further details are enumerated in Section 7 – Post-Installation Activities, later in this guide.



Zen Cart v1.5.5

Setup Finshed

Installation is now complete. You can access your storefront and your Administration area using the links below.

Your admin directory could not be renamed automatically, you will need to rename your admin directory before accessing it

You need to remove the /zc_install/ folder so that someone can't re-install your shop again and wipe out your database! A message will appear and you will not be able to log into your admin until the folder has been removed.

Your Admin Backend:

https://www.example.com/mystorefolder/my-renamed-admin/

Your Storefront:

https://www.example.com/mystorefolder/

Copyright © 2003-2016 Zen Cart®

## 6.2 *Using zc_install to do The Database Upgrade Step of a Site Upgrade*

### 6.2.1 Introduction

Upgrading consists of both manually updating the PHP files on your site, as well as upgrading the database structure to work with the new requirements of the new version. IT IS NOT ENOUGH TO SIMPLY UPGRADE THE DATABASE. YOU MUST ALSO UPGRADE ALL YOUR PHP FILES ACCORDING TO THE INSTRUCTIONS CONTAINED IN EACH INDIVIDUAL RELEASE.

**The document you are reading here does not address doing the PHP file updates**. See the proper upgrade documentation at https://www.zen-cart.com/upgrades to understand and do the full upgrade process. The instructions below ONLY deal with the database-side of the upgrade step.

DATABASE UPGRADE STEP:

To do the database upgrade step, you will use zc_install just as you would for a manual new install:

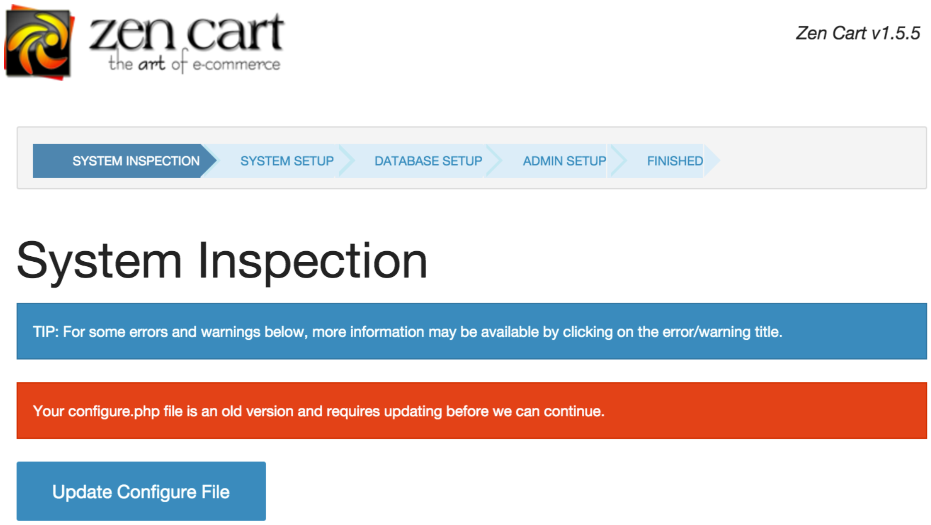
To run the Zen Cart® installation/upgrade wizard, you will need to use your browser to access the webserver where you installed Zen Cart®. The installation wizard is accessed from the folder /zc_install, ie: https://www.MY_DOMAIN.com/zc_install/

The upgrade dialogs are explained on the following pages ...

### 6.2.2 Upgrading configure.php files, if necessary

The first step that zc_install will do is inspect your existing configure.php files, and offer to update their format if they appear to be from an older version.

(The newer files are MUCH leaner, and simpler to read and maintain.)



System Inspection

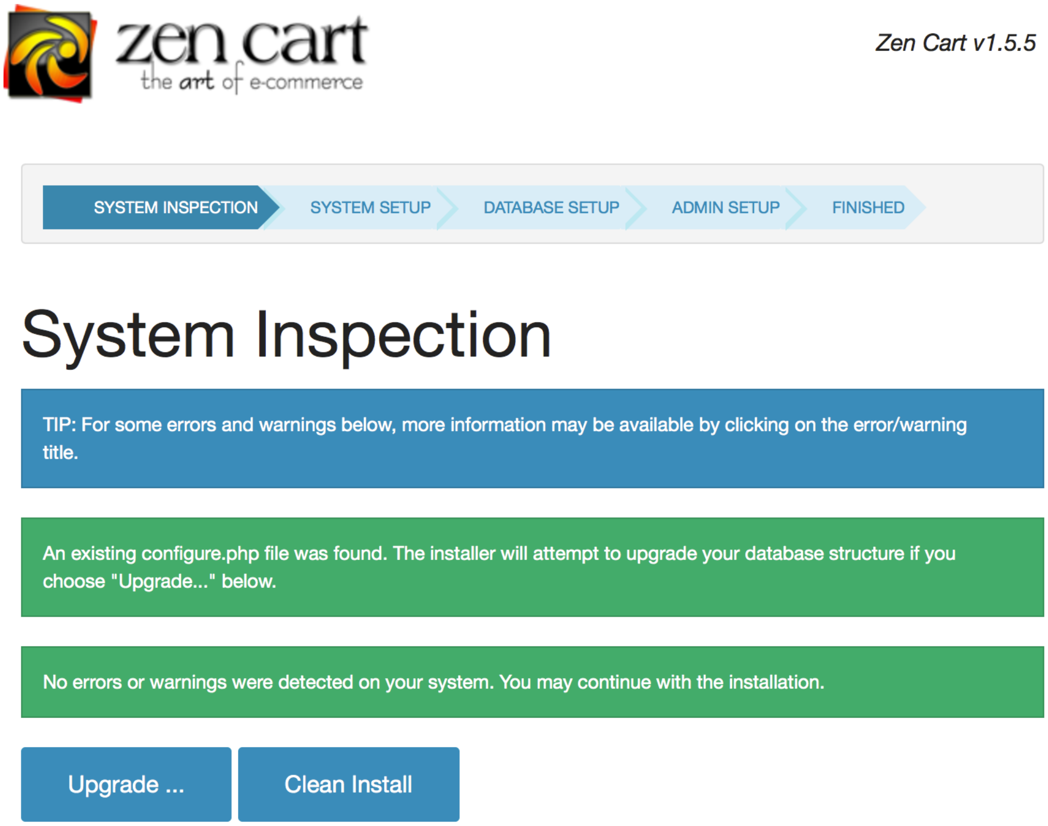
TIP: For some errors and warnings below, more information may be available by clicking on the error/warning title.

Your configure.php file is an old version and requires updating before we can continue.

### 6.2.3 Step 1 Welcome Screen and System Inspection

When you start zc_install in conjunction with a database from an older version, the Inspection will tell you that an upgrade is available, and will give you a button to proceed with upgrading your database.

You may first see a few warnings that need attention, such as ensuring your configure.php files are writable (so that zc_install can auto-upgrade those files to the new format). The "upgrade" button will not appear until critical issues are first resolved, as described on-screen.



System Inspection

TIP: For some errors and warnings below, more information may be available by clicking on the error/warning title.

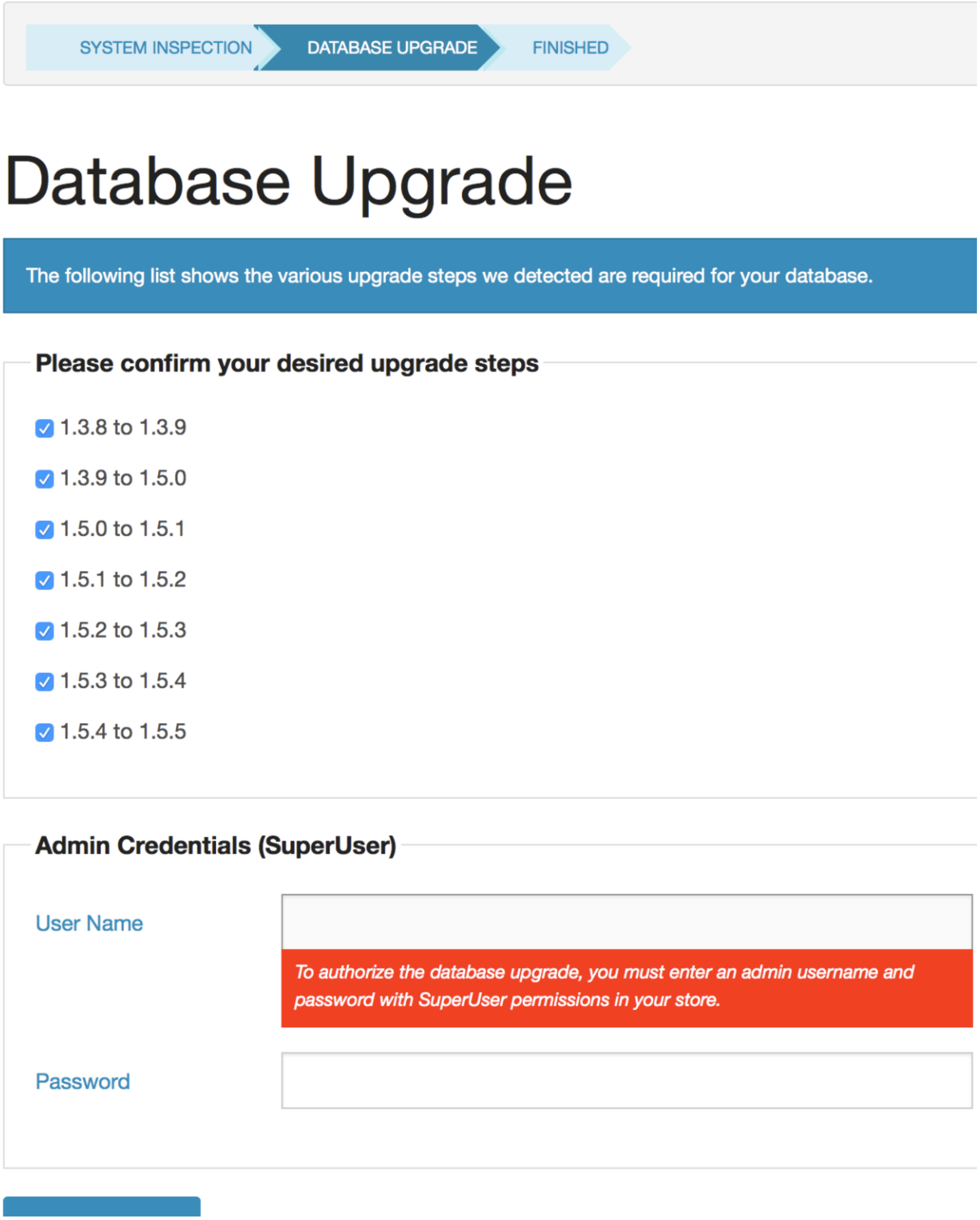
An existing configure.php file was found. The installer will attempt to upgrade your database structure if you choose "Upgrade..." below.

No errors or warnings were detected on your system. You may continue with the installation.

### 6.2.4 Step 2 Version-Upgrade Checkboxes

Now you are presented with a list of version-upgrade steps that zc_install is capable of upgrading for you. The system will pre-inspect your database and pre-check the checkboxes for the version steps which need upgrades performed in your database. In a normal upgrade, you simply need to leave the checkboxes as-is, and scroll to the bottom of the page and fill in your Admin username and password and click Update Database Now to authorize the upgrade.

Checking or unchecking additional boxes is an advanced troubleshooting activity which would only be relevant in the event of a serious problem requiring overriding of normal operation. As for help on the Support Forum if the version detection doesn't automatically work as expected.



# Database Upgrade

The following list shows the various upgrade steps we detected are required for your database.

### Please confirm your desired upgrade steps

☑ 1.3.8 to 1.3.9

☑ 1.3.9 to 1.5.0

☑ 1.5.0 to 1.5.1

☑ 1.5.1 to 1.5.2

☑ 1.5.2 to 1.5.3

☑ 1.5.3 to 1.5.4

☑ 1.5.4 to 1.5.5

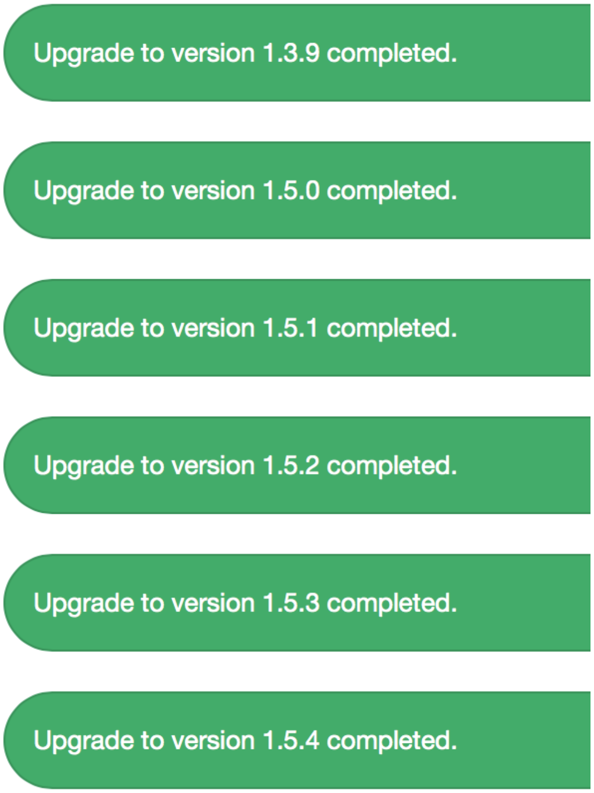
### Admin Credentials (SuperUser)

User Name

*To authorize the database upgrade, you must enter an admin username and password with SuperUser permissions in your store.*

Password

After submitting the form, the upgrade progress will be shown as it goes through each stage, similar to the image here:



Upgrade to version 1.3.9 completed.

Upgrade to version 1.5.0 completed.

Upgrade to version 1.5.1 completed.

Upgrade to version 1.5.2 completed.

Upgrade to version 1.5.3 completed.

Upgrade to version 1.5.4 completed.

If there are no errors, it will automatically progress to a "Setup Finished" screen with some final instructions such as removing the zc_install directory now that you're done.

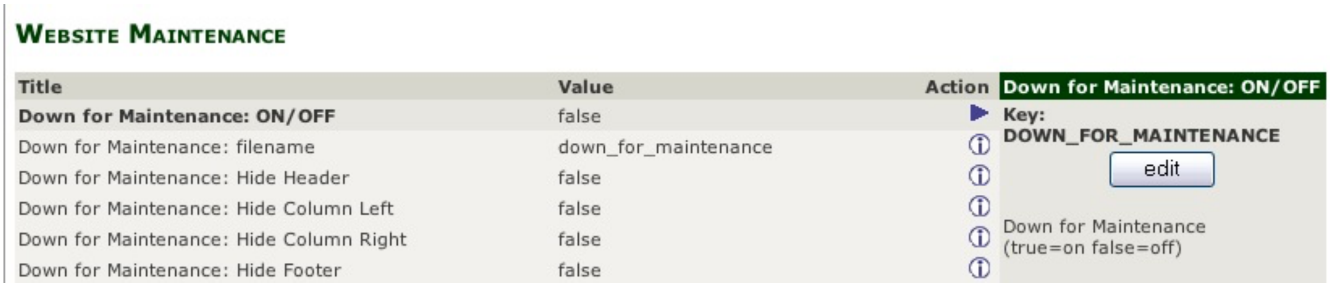
Be sure to turn off maintenance mode, via your Admin, as described in the next section.

AND be sure to upgrade the rest of your site's PHP files, following the upgrade guidance at https://docs.zen-cart.com/user/upgrading/detailed_upgrading/ or https://www.zen-cart.com/entry.php?3-How-do-I-rebuild-my-site-on-the-new-version-instead-of-upgrading

### 6.2.5 Step 3 Database-Upgrade Step Finished

REMEMBER: An upgrade is NOT COMPLETE if you do not ALSO upgrade all your PHP files to work with the new version. This involves reconstructing your customizations in the new versions of the files.

With the Database Upgrade step complete, **if your PHP files have also been updated** and your customizations merged into them, then you're ready to log into your Admin and turn off the Down-For-Maintenance mode via Admin->Configuration->Website Maintenance



**Website Maintenance**

| Title | Value | Action |
|---|---|---|
| **Down for Maintenance: ON/OFF** | false | ▶ |
| Down for Maintenance: filename | down_for_maintenance | ⓘ |
| Down for Maintenance: Hide Header | false | ⓘ |
| Down for Maintenance: Hide Column Left | false | ⓘ |
| Down for Maintenance: Hide Column Right | false | ⓘ |
| Down for Maintenance: Hide Footer | false | ⓘ |

**Down for Maintenance: ON/OFF**

**Key: DOWN_FOR_MAINTENANCE**

Down for Maintenance (true=on false=off)

# 7. Post-Installation Actions

## 7.1 Changing The Admin Directory Name for Security (By-Obscurity)

When Zen Cart® is initially installed, the admin panel is found at INSTALL_DIRECTORY/admin

Since the name "admin" is publicly known, leaving it as "admin" poses some degree of security threat. Therefore, zc_install will normally attempt to rename it for you and tell you the new name during the last step of zc_install. However, if it was unable to do the rename, then before you can access the admin panel you must change the name of that directory to something difficult to guess. If you do not, you will see the warning message mentioned in section 7.5 below.

Changing the admin directory involves simply renaming the admin folder name using your FTP program, and is explained in detail in this tutorial: https://docs.zen-cart.com/user/installing/rename_admin/

## 7.2 Enabling SSL for HTTPS in your Admin

**For PA-DSS compliance, all non-console access (ie: all browser access) to your admin area must be via HTTPS to ensure strong cryptography.** To do this, simply ensure that the HTTP_SERVER URL in your /renamed-admin/includes/configure.php file contains a URL beginning with https://… .

## 7.3 Setting Directory and File Permissions

As mentioned in the **Pre-Installation Actions** section of this guide, a number of directories and files need special permissions in order for Zen Cart® to function correctly. At a minimum these include:

- The directory INSTALL_DIRECTORY/**cache** needs to be writeable.
- The directory INSTALL_DIRECTORY/**logs** needs to be writeable.
- The directory INSTALL_DIRECTORY/**pub** needs to be writeable.
- The directory INSTALL_DIRECTORY/**images** needs to be writeable.
- The directory INSTALL_DIRECTORY/**admin/images/graphs** needs to be writeable.

Also during installation you will have created these files with write permissions:

- INSTALL_DIRECTORY/includes/configure.php and
- INSTALL_DIRECTORY/admin/includes/configure.php

**For security, these 2 files should have their permissions changed so that they are read-only.**

If you need help understanding the concepts of file permissions and some general guidance on making these changes, consult this tutorial article: https://docs.zen-cart.com/user/first_steps/how_do_i_install/

N.B. The Zen Cart® application will attempt to automatically set the permissions of these two files to read-only after installation is complete. However this doesn't work automatically on all servers, and you should ensure that this it is done properly. (Warnings will be prominently displayed if the permissions on these files are incorrect.)

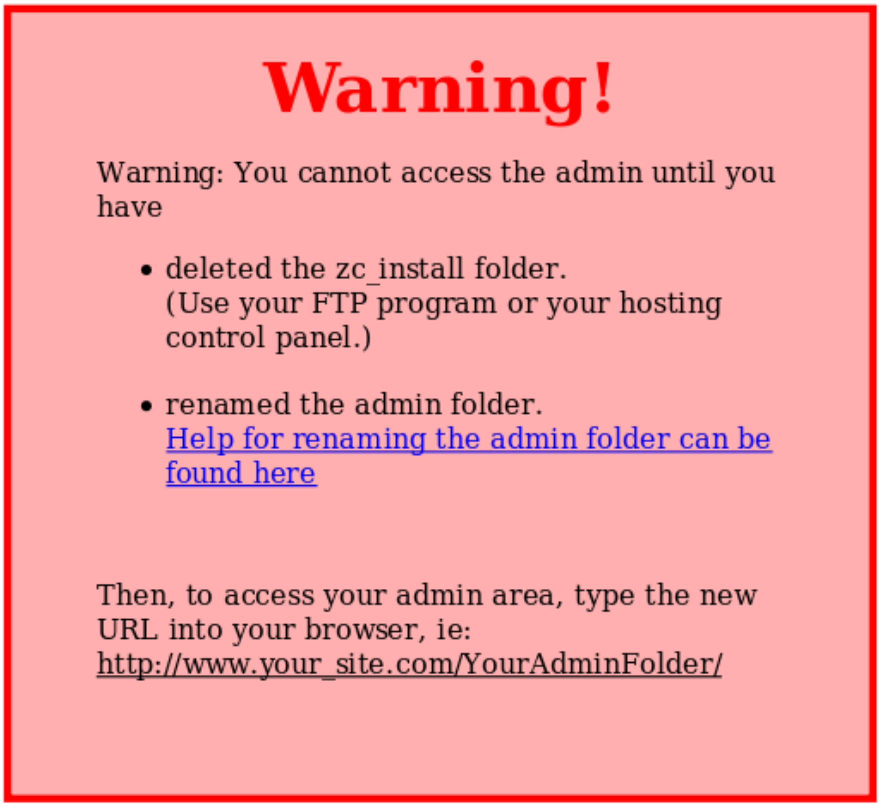
## 7.4 Removing the Installation Directory

Once installation is completed, you must remove the INSTALL_DIRECTORY/**zc_install** folder.

Leaving the zc_install folder on your server poses a security risk, since someone could possibly delete your store database content if they gained unauthorized access to it. Thus, until you remove the zc_install directory, a warning message will be presented if the folder exists when you access your admin console.

## 7.5 Blocked Administration Access

Please note, if you have not changed your admin directory, or have not removed the installation directory then you will not be able to access the Administration console and you will be presented with a screen as below.



Warning!

Warning: You cannot access the admin until you have

- deleted the zc_install folder.
  (Use your FTP program or your hosting control panel.)
- renamed the admin folder.
  Help for renaming the admin folder can be found here

Then, to access your admin area, type the new URL into your browser, ie:
http://www.your_site.com/YourAdminFolder/

Further details on how to change the name of the admin directory can be found at: https://docs.zen-cart.com/user/installing/rename_admin/

## 7.6 Removing Unnecessary Directories

Once installation is completed, you should remove these two folders, to prevent snoopers from gaining information about your site that they have no business knowing, such as your ZC version, and so on:

- INSTALL_DIRECTORY/docs

- INSTALL_DIRECTORY/extras

# 8. Accessing the Administration Panel and Configuring Administrative Users and Passwords

## *8.1 Introduction*

Zen Cart® includes a system for managing multiple admin users and restricting the access of those users to only certain functions of the Administration system.

Initially only one user is created (The user/password you created during installation). This user is assigned a 'Superuser' profile, and has access to all administration functionality. Additional profiles can be created to offer lesser permissions to specific administrative users. This is explained further in the following sections.

**PA-DSS**

Since you will probably have performed the installation without using HTTPS, there is a (small) possibility that a someone might have intercepted your admin username/password.

**If you don't have HTTPS enabled on your site yet, you need to do that NOW (see section 7.2 of this guide), and then change your admin password.**

## *8.2 Administrator Access to Credit Card Numbers*

Zen Cart® doesn't store complete credit card numbers, nor any sensitive/authentication data such as expiry dates or CVV numbers.

All storage of card numbers consists of only the first 4-6 digits and the last 4 digits, in accordance with PCI and merchant agreement rules.

The only place where these partial card numbers are visible is when viewing an order in the administration screens, via Admin->Customers->Orders.

And if payment was handled completely offsite then it's unlikely that any of the partial card number will show at all.

**PCI DSS**

**For PCI compliance, you are not permitted to alter the database to be able to store complete card numbers, nor store them unencrypted in any way unless you use complete cryptographic key infrastructure and undergo your own PCI assessment to authorize those changes. Any alteration of this nature, or any use of 3rd-party modules which provide such features requires your own PCI assessment, else you are not PCI compliant. Discussed more in chapter 15.**

## *8.3 Administrative User Access and PA-DSS requirements*

Before discussing how Admin users are managed, note some fundamental features implemented in this area since Zen Cart® v1.5.0 in order for the application to meet PA-DSS requirements:

Malicious individuals will often try to find accounts with weak or non-existent passwords in order to gain access to an application or system. If passwords are short or simple to guess, it is relatively easy for a malicious individual to find these weak accounts, and compromise an application or system under the guise of a valid user ID.

The PA-DSS requirements are as follows:

- Each admin user must have a unique user name. DO NOT share admin usernames with others. Create separate users for each person accessing your store's admin section.
- Passwords must be at least 7 characters in length

Passwords must consist of a mix of letters and numbers(alphanumeric)

- Passwords and user names are case sensitive
- Passwords must be changed every 90 days
  When changing passwords, users will not be able to choose a password that was used in the prior 4 password changes
- When attempting login, if an incorrect user name/password is entered more than 6 times in a row, access to the administration system will be locked out for 30 minutes
  (For even greater protection, Zen Cart® also implements additional anti-brute-force protection measures beyond this requirement.)
  This lockout can be overridden only by another authorized administrator, after they first confirm the identity of the person requesting the override (ie: talk to the person to whom the account belongs, to be sure they're not a malicious intruder)
- The administration system has a 15 minute inactivity timeout. That is, if no pages are accessed (no clicks which submit data or load a new page) within a 15 minute period, the admin user will be forced to log in again.  Extending this timeout period will invalidate your store's PCI Compliance.

Altering the code to relax these requirements will invalidate your store's PCI Compliance.

Further, the wording of the PA-DSS is: "to maintain PCI DSS compliance, any changes made to authentication configurations would need to be verified as providing authentication methods that are at least as rigorous as PCI DSS requirements."

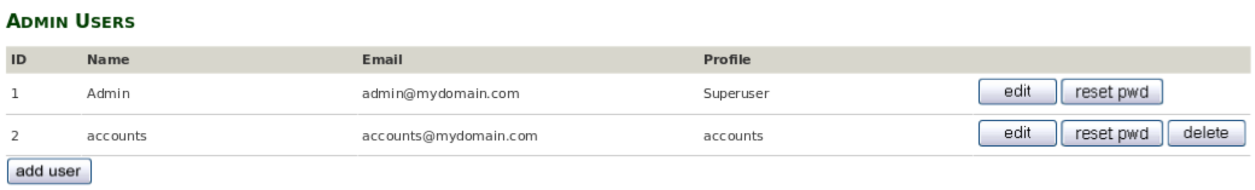
**Disabling PCI password/timeout features:**
NOTE: If you use the switch in Admin->Configuration->My Store to disable the **PA-DSS Admin Session Timeout Enforced**, or the **PA-DSS Strong Password Rules Enforced** settings, YOU ARE MAKING YOUR SITE NO LONGER PCI COMPLIANT. Discussed in chapter 14.4

## 8.4 Users

You can easily manage the users who are allowed access to the administration system using the *Admin Access Management → Admin Users* menu entry.

From this screen you will be able to add, delete and change the details of Admin users.

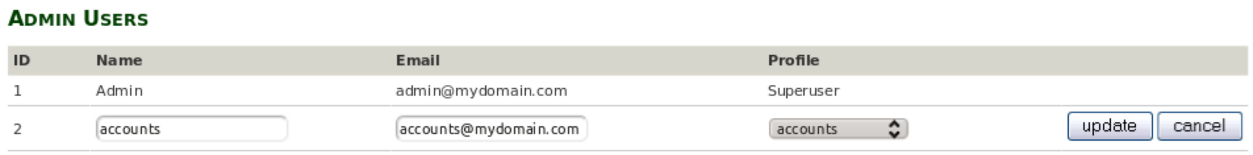
***NOTE: For PCI DSS compliance, every person with administrative access must have their own unique user ID and password, and should never re-use the same ID+password on more than one system/website.***



Admin Users

| ID | Name | Email | Profile | |
|---|---|---|---|---|
| 1 | Admin | admin@mydomain.com | Superuser | edit reset pwd |
| 2 | accounts | accounts@mydomain.com | accounts | edit reset pwd delete |

Clicking the edit button will allow you to change the Admin User name, Admin User email address and the profile



Admin Users

| ID | Name | Email | Profile | |
|---|---|---|---|---|
| 1 | Admin | admin@mydomain.com | Superuser | |
| 2 | accounts | accounts@mydomain.com | accounts | update cancel |

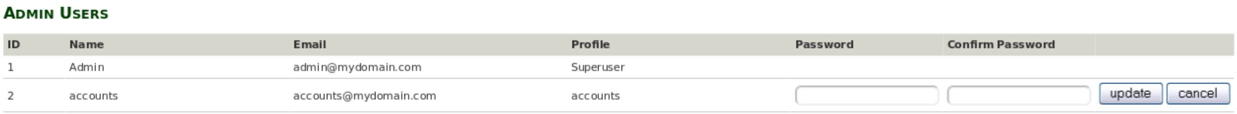
assigned to that user (although you cannot change the profile assigned to the initial Admin User).

Clicking the "reset pwd" button allows you to change the password assigned to the user.

***NOTE: The system only accepts unique usernames, thus no two users can have the same unique userid at any given time.***

## 8.5 Profiles

Profiles describe which functions in the administration system a user can access. Initially only one profile is available, the 'Superuser' profile, which gives access to all the administration system.



ADMIN USERS

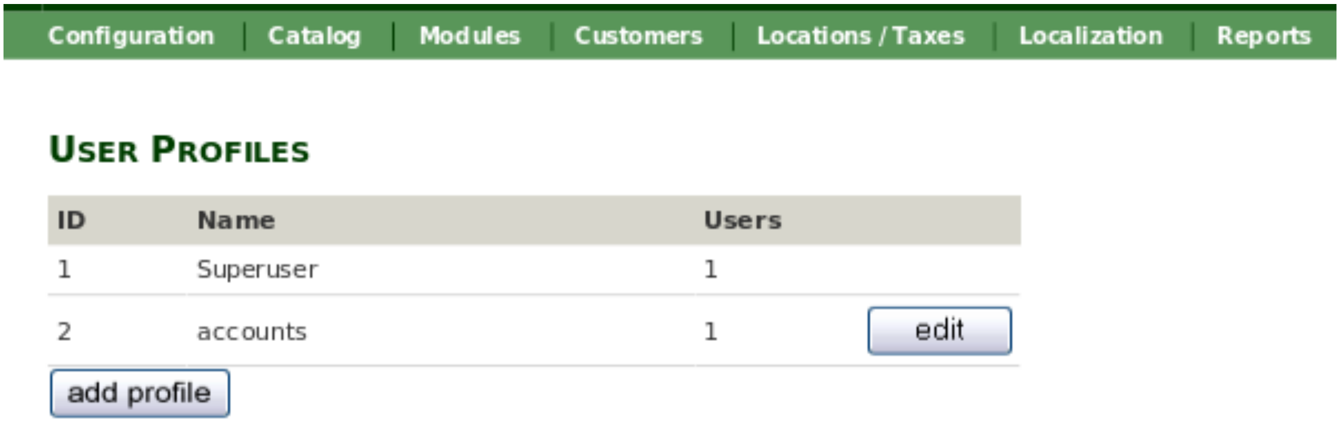
| ID | Name | Email | Profile | Password | Confirm Password |
|---|---|---|---|---|---|
| 1 | Admin | admin@mydomain.com | Superuser | | |
| 2 | accounts | accounts@mydomain.com | accounts | | |

However it is also possible to create profiles, so that different users only have access to a subset of the administration system.

For example, you may have users that only need to run reports, or users whose responsibility it is to add products/categories, and those users should not have access to any other administration system functions. ***NOTE: For PCI-DSS and PA-DSS compliance, only persons requiring access to payment details should be allowed to see them.***

This can be achieved by creating specific profiles, and then assigning those profiles to users.

You can access the Admin Profiles management page using the *Admin Access Management → Admin Profiles* menu entry.

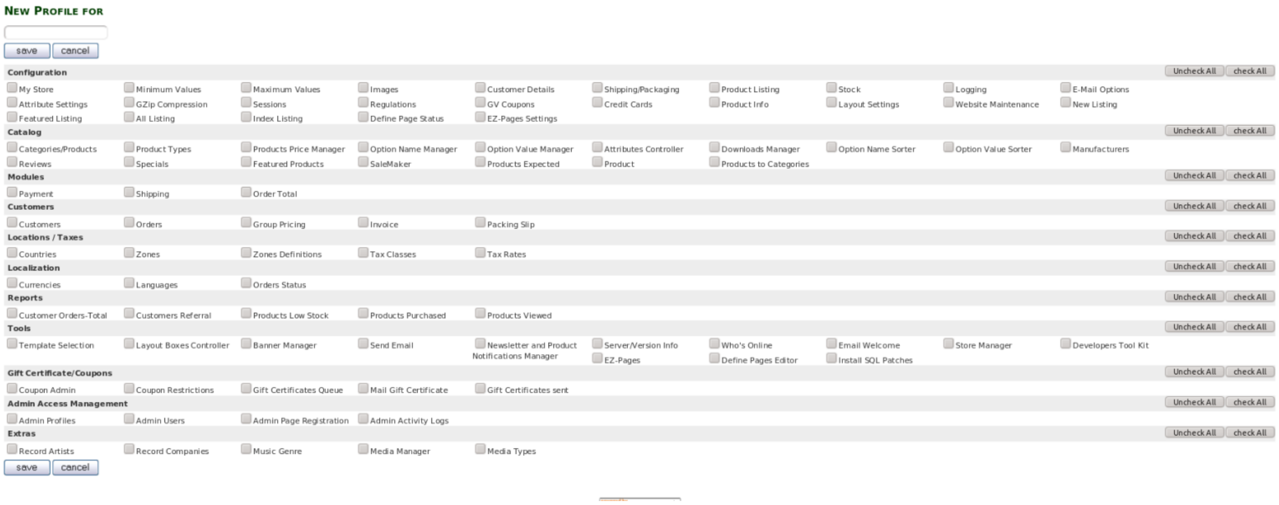


USER PROFILES

| ID | Name | Users |
|---|---|---|
| 1 | Superuser | 1 |
| 2 | accounts | 1 |

To add a new profile, click on the 'add profile' button .

You will then get a screen similar to the following, where you can enter a name for the profile, and select which administration system functions that profile has access to.



**NEW PROFILE FOR**

save cancel

**Configuration** Uncheck All check All

☐ My Store ☐ Minimum Values ☐ Maximum Values ☐ Images ☐ Customer Details ☐ Shipping/Packaging ☐ Product Listing ☐ Stock ☐ Logging ☐ E-Mail Options
☐ Attribute Settings ☐ GZip Compression ☐ Sessions ☐ Regulations ☐ GV Coupons ☐ Credit Cards ☐ Product Info ☐ Layout Settings ☐ Website Maintenance ☐ New Listing
☐ Featured Listing ☐ All Listing ☐ Index Listing ☐ Define Page Status ☐ EZ-Pages Settings

**Catalog** Uncheck All check All

☐ Categories/Products ☐ Product Types ☐ Products Price Manager ☐ Option Name Manager ☐ Option Value Manager ☐ Attributes Controller ☐ Downloads Manager ☐ Option Name Sorter ☐ Option Value Sorter ☐ Manufacturers
☐ Reviews ☐ Specials ☐ Featured Products ☐ SaleMaker ☐ Products Expected ☐ Product ☐ Products to Categories

**Modules** Uncheck All check All

☐ Payment ☐ Shipping ☐ Order Total

**Customers** Uncheck All check All

☐ Customers ☐ Orders ☐ Group Pricing ☐ Invoice ☐ Packing Slip

**Locations / Taxes** Uncheck All check All

☐ Countries ☐ Zones ☐ Zones Definitions ☐ Tax Classes ☐ Tax Rates

**Localization** Uncheck All check All

☐ Currencies ☐ Languages ☐ Orders Status

**Reports** Uncheck All check All

☐ Customer Orders-Total ☐ Customers Referral ☐ Products Low Stock ☐ Products Purchased ☐ Products Viewed

**Tools** Uncheck All check All

☐ Template Selection ☐ Layout Boxes Controller ☐ Banner Manager ☐ Send Email ☐ Newsletter and Product Notifications Manager ☐ Server/Version Info ☐ Who's Online ☐ Email Welcome ☐ Store Manager ☐ Developers Tool Kit
☐ EZ-Pages ☐ Define Pages Editor ☐ Install SQL Patches

**Gift Certificate/Coupons** Uncheck All check All

☐ Coupon Admin ☐ Coupon Restrictions ☐ Gift Certificates Queue ☐ Mail Gift Certificate ☐ Gift Certificates sent

**Admin Access Management** Uncheck All check All

☐ Admin Profiles ☐ Admin Users ☐ Admin Page Registration ☐ Admin Activity Logs

**Extras** Uncheck All check All

☐ Record Artists ☐ Record Companies ☐ Music Genre ☐ Media Manager ☐ Media Types

save cancel

The Edit button will bring up a similar screen, however in this case you will be able to change the current functionality granted to all users associated with that profile.

## *8.6 Admin Activity Logs*

The Admin Activity Log stores important information which might expose malicious activity being conducted by admin users (whether known or unknown) in the backend of your store.

The system logs this data:

- The date and time of the access
- The admin id of the user making the access (user identification)
- The page in the administration system that is being accessed (which infers type of event)
- The parameters related to the page being accessed (which infers identification of affected data and ostensibly the success or failure of the attempted action)
- The IP address (origination) of the admin user performing the event
- Any "suspect" activity that should be reviewed, such as malicious POST data
- Changes made to payment/shipping modules and admin users

There are no built-in settings to alter this functionality. It is installed and enabled by default.

Tampering with this logging functionality or disabling the logs or changing the logging code will result in non-compliance with PCI DSS.

The activity log is held in the database, and over time can become very large. You can manage your activity log via: *Admin Access Management → Admin Activity logs*

**It is important to review these logs regularly, even daily, to monitor for malicious activity and respond accordingly.**

The following sections discuss the review and management of these logs.

### 8.6.1 Daily Log Review – Important Things To Monitor

**PA-DSS**
Please Note: It is a requirement of PA-DSS that you review these logs regularly to detect unauthorized activity and take corrective action to deal with any anomalies discovered therein.

Regular review of these logs will help you avert problems caused by people who have gained unauthorized access to your admin backend, whether that be a hacker, intruder, or even a disgruntled employee.

The “logmessage” column explains what situation occurred. Sometimes this is simply informational, but sometimes it will tell you something more significant, such as “Failed login attempt for baduser123”, or “Account [poor_speller] locked out due to too many invalid signon attempts”, or “Payment module [authorizenet] removed by {gooduser2}”.

In addition, the “flagged” items shown in the Review screen are items which warrant some attention. If a log entry is flagged, that means that some potentially-harmful content has been entered into the admin page which was in use at the time of that log entry. Commonly-flagged items include <script> tags where someone could inject malicious javascript to trigger or create XSS or CSRF risks on your store's admin or storefront.

If you find an entry that's been flagged, you should inspect the data that was submitted (ie: see the “postdata” column) to be sure it was intentional. If it was not intentional or authorized, you should take corrective action to remove the malicious or unwanted content, and also take corrective action to deal with whomever was logged in and submitted the content in the first place. Follow your own internal policies for dealing with such breaches.

Additionally, a severity level is indicated for each entry. These levels have a similar rating as established by the syslog standard as maintained in the computer/technology industry. The levels which apply to your store are:

- WARNING – critical events which should be reviewed daily, as these denote removal of important settings or administrative user privileges, or other serious events.
- NOTICE – important events which should be monitored regularly (at least weekly) to review whether a disgruntled employee or a website intruder has attempted to insert malicious content into your site product descriptions, or install new modules, etc.
- INFO – general list of all activity (every click) done by any administrative user. Useful for post-mortem assessment of a problem if one were to occur.

The following section talks briefly about how the Admin Activity Log viewer screen works.

### 8.6.2 Review or Export Logs

Within this section you can choose to export or review the admin activity logs. An image of the screen is shown on the following page.

First select which data you want to process, according to severity level, using the pulldown menu provided.

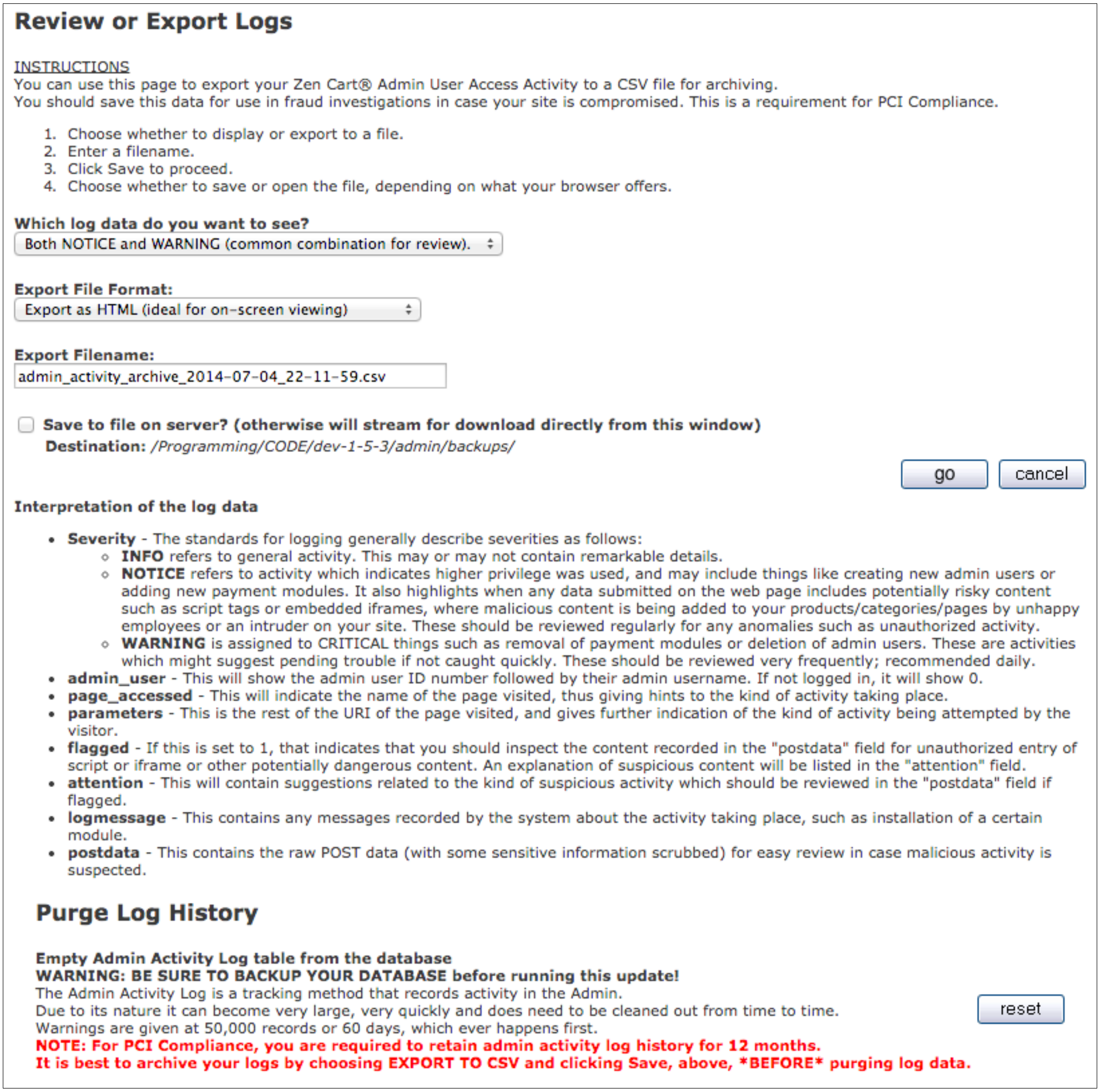
Select the desired Export format:

- To see the data on-screen, simply leave the setting set to “Export as HTML”. NOTE: This may only show the latest 50 entries. For a full detailed report of the entire log, use the CSV option below:
- If you want to export/save the logs then this pulldown must be set to 'Export to CSV'

If you are saving to a CSV file, you can choose to either download the exported file to your local computer immediately or save the file to your webserver and view it via FTP at a later date.
In either case you can also choose the name of the file produced.

To save it to your webserver, tick the check box marked 'Save File on server'.
NOTE: For this to work you must make sure the webserver is configured so that the folder specified by “Destination” is writable by the webserver (for most linux servers this would be for the www-data user).

You can see a sample of the screen in the figure below:



**Review or Export Logs**

INSTRUCTIONS
You can use this page to export your Zen Cart® Admin User Access Activity to a CSV file for archiving.
You should save this data for use in fraud investigations in case your site is compromised. This is a requirement for PCI Compliance.

1. Choose whether to display or export to a file.
2. Enter a filename.
3. Click Save to proceed.
4. Choose whether to save or open the file, depending on what your browser offers.

**Which log data do you want to see?**
Both NOTICE and WARNING (common combination for review).

**Export File Format:**
Export as HTML (ideal for on-screen viewing)

**Export Filename:**
admin_activity_archive_2014-07-04_22-11-59.csv

□ **Save to file on server? (otherwise will stream for download directly from this window)**
**Destination:** */Programming/CODE/dev-1-5-3/admin/backups/*

**Interpretation of the log data**

- **Severity** - The standards for logging generally describe severities as follows:
  - **INFO** refers to general activity. This may or may not contain remarkable details.
  - **NOTICE** refers to activity which indicates higher privilege was used, and may include things like creating new admin users or adding new payment modules. It also highlights when any data submitted on the web page includes potentially risky content such as script tags or embedded iframes, where malicious content is being added to your products/categories/pages by unhappy employees or an intruder on your site. These should be reviewed regularly for any anomalies such as unauthorized activity.
  - **WARNING** is assigned to CRITICAL things such as removal of payment modules or deletion of admin users. These are activities which might suggest pending trouble if not caught quickly. These should be reviewed very frequently; recommended daily.
- **admin_user** - This will show the admin user ID number followed by their admin username. If not logged in, it will show 0.
- **page_accessed** - This will indicate the name of the page visited, thus giving hints to the kind of activity taking place.
- **parameters** - This is the rest of the URI of the page visited, and gives further indication of the kind of activity being attempted by the visitor.
- **flagged** - If this is set to 1, that indicates that you should inspect the content recorded in the "postdata" field for unauthorized entry of script or iframe or other potentially dangerous content. An explanation of suspicious content will be listed in the "attention" field.
- **attention** - This will contain suggestions related to the kind of suspicious activity which should be reviewed in the "postdata" field if flagged.
- **logmessage** - This contains any messages recorded by the system about the activity taking place, such as installation of a certain module.
- **postdata** - This contains the raw POST data (with some sensitive information scrubbed) for easy review in case malicious activity is suspected.

**Purge Log History**

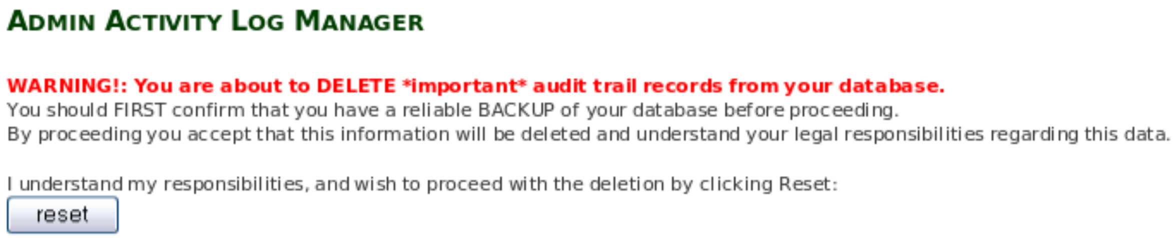
**Empty Admin Activity Log table from the database**
**WARNING: BE SURE TO BACKUP YOUR DATABASE before running this update!**
The Admin Activity Log is a tracking method that records activity in the Admin.
Due to its nature it can become very large, very quickly and does need to be cleaned out from time to time.
Warnings are given at 50,000 records or 60 days, which ever happens first.
**NOTE: For PCI Compliance, you are required to retain admin activity log history for 12 months.**
**It is best to archive your logs by choosing EXPORT TO CSV and clicking Save, above, *BEFORE* purging log data.**

## 8.6.3 Purge Log History action

**PA-DSS**

Please Note: It is a requirement of PA-DSS that these logs are kept for a minimum of 12 months. While we provide methods to safely backup these logs, and to remove them, store managers are ultimately responsible for ensuring that they can reproduce these logs in the event of a PCI audit.

Clicking on the reset button in the "Purge Log History" section of the screen will take you to a new page with the following instructions:


Admin Activity Log Manager

WARNING!: You are about to DELETE *important* audit trail records from your database.
You should FIRST confirm that you have a reliable BACKUP of your database before proceeding.
By proceeding you accept that this information will be deleted and understand your legal responsibilities regarding this data.

I understand my responsibilities, and wish to proceed with the deletion by clicking Reset:

Please ensure that you read, understand, and heed the warning text **before** purging the activity log!

At the very least, save the log history before exporting!

### 8.6.4 PA-DSS Logging – Technical Details

**PA-DSS standards require that we state what information is logged. So to clarify the bulleted list from section 8.5.0 earlier, the following is logged:**
- Individual access to cardholder data  (given current versions of Zen Cart never store such data, there is nothing to log for this particular item)
- Actions taken by any individual with administrative privileges
- Access to application audit trails managed by or within the application
- Initialization of application audit logs
- Invalid logical access attempts (failed logins, shows the username attempted during the failure)
- Use of the payment application's identification and authentication mechanisms
- Creation and deletion of system-level objects within or by the application
- User identification
- Type of event
- Date and time stamp
- Success or failure indication
- Origination of event
- Affected data, system component, or resource

### 8.6.5 Centralized Logging

Logging can be extended by incorporating plugins to allow additional external centralized logging services to be incorporated.

Following is an example plugin, consisting of two PHP files, named with the expectation of using the graylog logging service. Actual graylog API coding logic and security credentials must be added to make it functional.

a) /admin-foldername/includes/auto_loaders/config.admin.graylog.php

```
<?php
  $autoLoadConfig[1][] = array('autoType'=>'class',
                              'loadFile'=>'class.admin.graylog.php',
                              'classPath'=>DIR_WS_CLASSES);
  $autoLoadConfig[40][] = array('autoType'=>'classInstantiate',
                              'className'=>'graylogObserver',
                              'objectName'=>'graylogObserver');
```

b) /admin-foldername/includes/classes/class.admin.graylog.php

```
<?php
```

```
/**
 * @package plugins
 * @copyright Copyright 2003-2013 Zen Cart Development Team
 * @license http://www.zen-cart.com/license/2_0.txt GNU Public License V2.0
 *
 * Designed for v1.5.2
 *
 * initially designed with expectation that this file is placed into /admin/includes/classes/ folder,
 * and corresponding config.admin.graylog.php file into admin/includes/auto_loaders folder
 *
 */

class graylogObserver extends base {

  function __construct() {
    global $zco_notifier;
    $zco_notifier->attach($this, array('NOTIFY_ADMIN_ACTIVITY_LOG_ADD_RECORD'));
  }

  function updateNotifyAdminActivityLogAddRecord(&$class, $eventID, $paramsArray = array())
  {
    // insert relevant calls to graylog API here, passing the data from $paramsArray as arguments
  }
}
```

# 9. Code Customization, Addons, and Plugins

The Zen Cart® community has a vast assortment of available addons/plugins contributed by third-parties, most often other store owners, who have written customized code to extend the capability of Zen Cart® to do additional things beyond the core framework that is Zen Cart® itself. Many have simply put together the customizations they made for their store and shared them back as a significant way of participating and expanding the ever-growing community that has grown up around the Zen Cart® product.

You can find addons to suit almost any special need you might have. And if you can't find something exactly suiting your needs, you can either customize the code yourself or perhaps hire someone to do the customization for you. You are invited to share your customizations back to the community for others who follow along after you to freely enjoy using on their own stores, just as you have benefitted from similar actions by others.

If you don't have the skills to customize programming PHP code yourself and wish to hire someone, there's a vast help-wanted community available as well. You can access some help-wanted resources by following this link: https://www.zen-cart.com/helpwanted (be sure to do your own assessment of the competence of any person you hire to work on your site)

**PA-DSS**

NOTE: The Zen Cart® PA-DSS certification applies only to the official released code provided by Zen Ventures, LLC

**Any customizations you, or a 3rd-party, make to the code, whether by altering admin-provided configuration switches, adding addons/plugins, making code customizations, etc may render the PA-DSS certification void for your site.**

It is up to you to ensure that all addons, plugins, customizations, and changed switch settings, are PCI compliant according to the PCI Security Standards. If you have specific questions about whether a change you've made to your site is PCI-Compliant, contact your PCI Scanning vendor for assistance and guidance and appropriate auditing.

# 10. Engaging 3rd-Party Consultants or Programmers

As mentioned in the previous section, if you don't have the skills to customize programming PHP code yourself and wish to hire someone, there's a vast help-wanted community available as well. You can access some help-wanted resources by following this link: https://www.zen-cart.com/helpwanted (be sure to do your own assessment of the competence of anyone you hire).

**PA-DSS**
Note: It is a requirement of PA-DSS that 3rd-parties are only granted access to specific components truly required to complete the work requested, and that they use that access in a secure manner, and that they destroy all copies of all information obtained once said access is no longer required.

We recommend that you only engage persons who demonstrate an understanding of PCI DSS and will agree in writing that the work they perform for you will be compliant with PCI DSS requirements. To maintain PCI DSS compliance, any coding changes made to authentication or credit card or security configurations needs to be properly verified as being at least as rigorous as PCI DSS requirements.

Some “best practices” you should consider when engaging a consultant are found in following sections.

## *10.1 Webstore “Admin”/Backend access*

If you need to give someone access to your store's admin panel, create a dedicated unique admin user account+pwd for that person (don't let them use a generic username such as their company name), and grant them access to ONLY the features they will need to complete the tasks assigned to them, and remove that account when they are finished. See section 8 for details. If your admin uses 2-Factor Authentication (see section 10.5) then ALL persons having access must use it.

**PCI-DSS requires that anyone accessing the Admin implement and use remote access security features. And should never use the same ID+password on more than one site; always unique.**

## *10.2 FTP Access*

- FTP and SFTP Accounts
  ***(NOTE: PCI-DSS requires the use of Secure FTP, not plain unencrypted FTP)***
  If you engage someone to do work on your website that requires them to have direct access to the files on your webserver, that will most likely require that they have FTP access.
  You should NEVER give them your master FTP password. You should ALWAYS create a new user+password for them using your hosting company's control panel.
  If possible, you should restrict their access to only their IP address, so it can't be abused.
  You should ALWAYS delete their FTP user as soon as their work is completed. It is dangerous to leave unmonitored accounts active in the hands of persons not in your direct supervision and employment. Similar wisdom should be applied to employees as well.

- Secure Access – Use SFTP, not FTP
  EVERYONE (including you!) accessing your webserver should be using SFTP to connect.
  If they use regular unencrypted FTP mode, then your customer data and website security could be compromised. See sections 2 and 4 of this guide for more information about SFTP.
  **PCI-DSS and PA-DSS require the use of Secure FTP, not unencrypted FTP.**

## *10.3 Webhosting Account's Control Panel access*

It is unusual for a 3rd-party to need access to your entire hosting account's control panel. If you must give them access, you must change the password to your hosting account when they are finished.

## *10.4 Secure use of customer database and website files*

If you, or a third party, need to make or use a copy of your store's database, either to prepare a staging/testing area, or to debug a problem, it is very important that the names of everyone who has access to this data are recorded, that they not share the data with anyone else, and that the data is securely deleted when no longer needed. Secure Erasure is best handled by a software tool that will securely obliterate the datafiles on your PC and anyplace where you've stored it. Tools for this can be found by numerous online vendors.

Agreement to handle data in this way should form an integral part of your contract with whomever is granted access to this information.

## *10.5 Two-Factor Authentication*

Two-Factor Authentication is the use of a third-party authorization mechanism to verify your identity when logging in. This is commonly implemented via the need to enter more than just a username and password, specifically not just something you "know", but also something you "have". See this wikipedia article for more explanation: Two_factor_authentication

Zen Cart® allows for the use of (but doesn't directly implement any specific) two-factor authentication as a means to further enhance the security of accessing your system. If you have engaged the use of a third-party two-factor authentication service, you can integrate it with Zen Cart® in one of two ways:

a) Follow the instructions of your two-factor authentication solution for adding the required directives to your /renamed-admin/.htaccess file. This will require the token authentication to take place before being allowed to enter your Zen Cart® admin username and password.

b) Add a custom PHP script to hook into your two-factor authentication solution by defining a constant named ZC_ADMIN_TWO_FACTOR_AUTHENTICATION_SERVICE with a value matching the name of the PHP function which will trigger it. Zen Cart® will first authenticate you using your Zen Cart® admin username and password, and then pass you on to your two-factor authentication solution for subsequent token validation. (It will pass an array containing the ZC admin user ID number, username, and email address, which the two-factor authentication system may optionally use. A boolean TRUE response is expected if login is approved. Anything else will trigger a failure.)

   The custom code for the custom function which triggers your two-factor authentication solution should be placed in the /renamed-admin/includes/functions/extra_functions/ folder. The constant mentioned above should be defined in a separate PHP file located in your /renamed-admin/includes/extra_datafiles/ folder. Any additional classes which your custom functions need to instantiate should be placed in your /renamed-admin/includes/classes/vendors/ folder. This folder may need to be created first.

If you require Two-Factor Authentication for other remote access activities such as VPN or FTP, you will need to configure those respective applications/utilities to use it as per the documentation provided by those utilities, much like you would configure your FTP program to know which server, username, password, etc to use for accessing your webserver.

NOTE: **The PCI 3 and 4 specification requires that all admin logins must use two-factor authentication when the Admin area provides access to any cardholder data.**

# 11. Removing Old Non-PCI-Compliant Data

If your store has used any payment modules that have stored full credit card data or cvv numbers, you must delete all such historical data from your database and your backups.

**PCI Compliance: You must NOT be using any payment module that stores complete credit card numbers or cvv numbers in your database, nor one which emails complete credit card numbers or cvv numbers to the storeowner or anyone else.**

**Zen Cart has NOT stored credit card numbers since v1.3.9a, released in 2010.** However, if you added plugins to add that functionality, or if you had used a prior version, you will need to take steps to clear out any credit card data which you had stored in your store. YOU MUST DO THIS for PCI compliance.

## *11.1 Removing Old Credit Card Data From Database Records*

When viewing an order in your store's Admin->Customers->Orders->Edit screen, if a full card number or cvv number is found, a “Delete” link will show, and clicking will replace that data with blanks.

To perform the cleanup database-wide using phpMyAdmin, you may run the following SQL query:

```
update zen_orders set cc_number = NULL, cc_cvv = NULL where cc_number != '';
optimize table zen_orders;
```

*(You may need to remove or substitute the “zen_” prefix on those two tablenames for the query to work on your site. Check the DB_PREFIX setting in your configure.php file to determine what prefix you need to use. If DB_PREFIX is blank, then remove “zen_” in the examples above.)*

After you've done this, you need to also securely delete the physical data from the server. See below.

## *11.2 Suggested Procedure For Secure Erasure of Old CHD data*

If your store has used any methods of storing full credit card numbers or CVV numbers, the PCI DSS requires that you take specific measures to securely delete all data that could be used to reconstruct historical database records or logs containing credit card/cvv details. The following steps outline an approach one could use for purging that data securely. **These steps require that you have administrator sysadmin access to your server.** Thus it may be necessary to engage your hosting company's server administrator to complete several of these steps.

a) Use the deletion methods mentioned in section 11.1 above to remove the credit card/cvv data from the database table records.

b) Take the store Down For Maintenance

c) Make a complete MySQL backup of your store's database tables. This will be used to restore the data after the deletion. *Test your backup to ensure that it is reliable, and make extra copies for safety*.

d) Using your MySQL console, or hosting control panel tools, delete your store's database and db user.

e) SERVER ADMIN: Shut down MySQL

f) SERVER ADMIN: Use an industry-accepted PCI Compliant secure-deletion tool (such as *sfill* which comes built-in to most standard Linux distributions or can be found in the THC:SecureDeleteToolkit) to delete all sensitive data using an erasure setting strong enough to prevent forensic recovery. Delete all sensitive data including at least:

   i. The physical files on the server which stored your store's MySQL database and MySQL log files

   ii. Free disk space (“slack” space). Remove all traces of the data from the free space (“deleted files”).

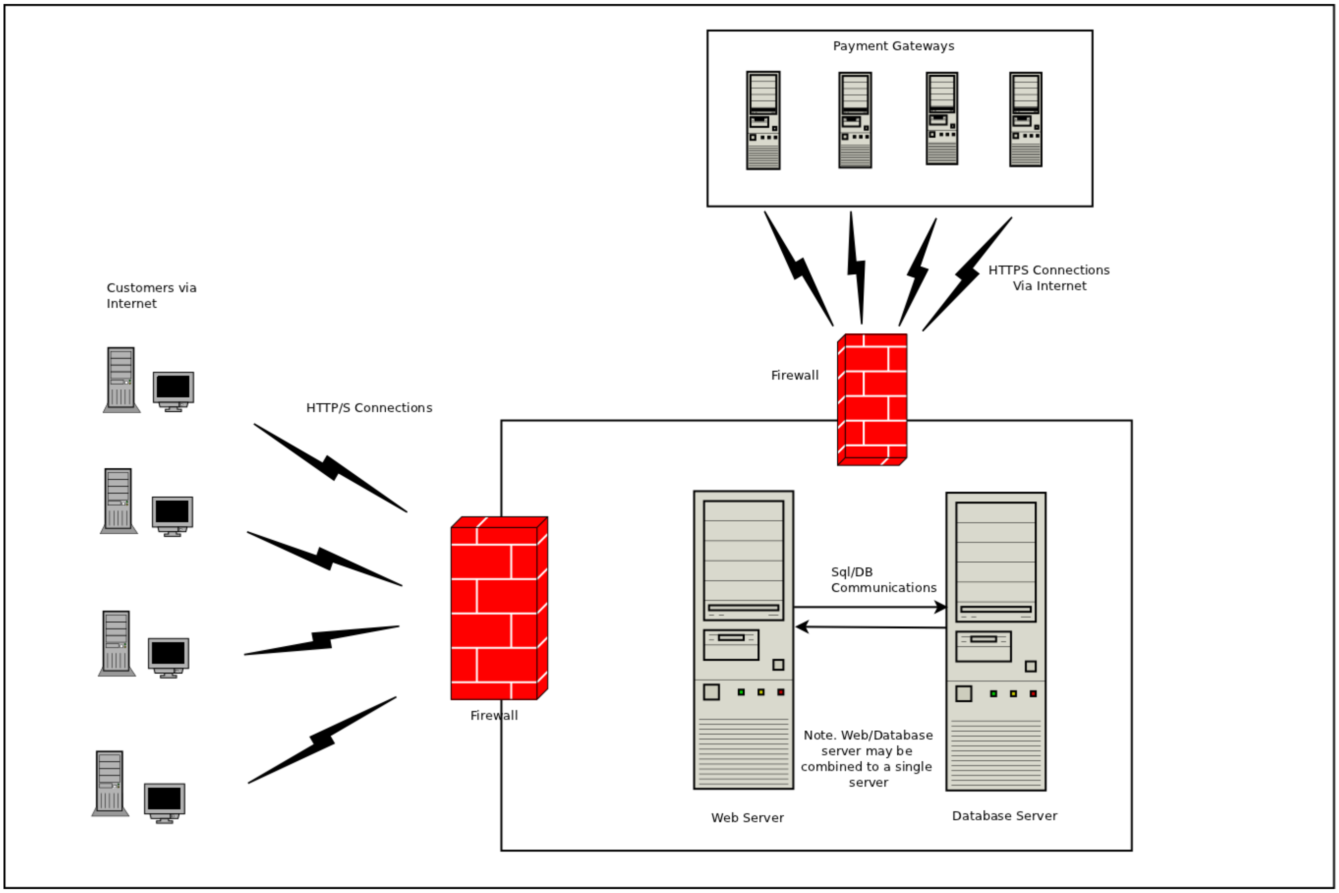
   iii. All backup copies of the database tables stored on any media, including remote or physical server hard disk backup images, backups (exports/dumps) you had stored on your PC or CDs/DVDs or thumb drives.

   iv. All debug log files associated with any payment modules that recorded CC/CVV numbers in flat files.

g) SERVER ADMIN: Start up MySQL

h) Recreate your store's MySQL database, username, password, and import your database from the backup you made in step (c) earlier.  Test your store for normal operation.

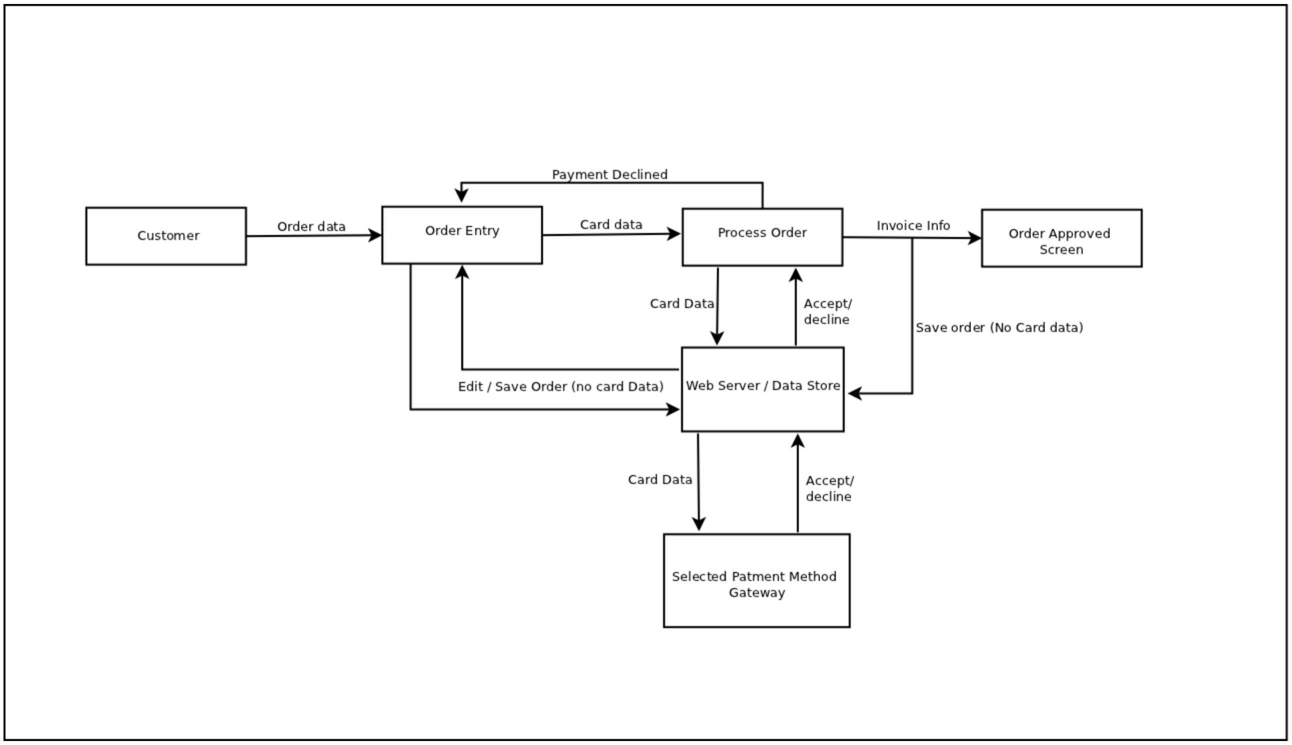
## 12. Network Diagram



Notes:

- In some hosting configurations, the Web Server and Database Server may reside on the same physical server.
- It is recommended that no wireless based systems should be connected to the Web/Database Server environment. Where such wireless equipment is connected then the application user or their hosting provider should:
    - Install perimeter firewalls between any wireless networks and systems that store, process, or transmit cardholder data and that Perimeter firewalls must deny or control all traffic from the wireless environment into the cardholder data environment.
    - Change default encryption keys
    - Change default SNMP community strings
    - Change default passwords for access points
    - Update firmware to support strong encryption for authentication and data transmission
    - Enable strong encryption (WPA2 or similar)
    - Change other default values as applicable
    - Use industry best practices to implement strong encryption for authentication and transmission

# 13. Dataflow Diagram



# 14. Notes about PA-DSS Compliance

## *14.1 Cardholder Data*

a. Storage

Out-of-the-box, Zen Cart® does not store cardholder data, and is crippled from being able to store entire card numbers (PAN) by virtue of database field lengths being too short to store a complete card number.

Thus, Zen Cart® does not display the complete PAN; if any partial PAN information is known (would not be known if payment was hosted/processed externally) only the first 4-6 and last 4 digits are shown, and visible on the Admin individual "order" page and during checkout during payment-confirmation. These settings are not configurable.

To maintain PCI DSS compliance, any coding changes made to handling or storage of cardholder data would need to be verified as providing handling or storage methods that are at least as rigorous as PCI DSS requirements.

b. SMS or Messaging Systems

While Zen Cart® does not give you any complete cardholder data to copy/paste, we remind you that it is unwise and unsafe to pass credit card data via any messaging technologies. Just don't do it.

c. Sessions

Further, Zen Cart® does not store cardholder data in sessions. As a stateless application, Zen Cart® does not retain any of the submitted cardholder information between clicks on the pages.

For example, if a customer uses a module which collects payment directly on a Zen Cart® site, the provided credit card data (ie: the card number and CVV and expiry date) are briefly stored in memory variables. After it finishes processing whatever steps that page executes, whether that's for authorization or anything else, those variables are removed from memory (internal memory pointers are destroyed for the objects and variables which have gone out of scope by virtue of code completion) by the garbage-collection procedures internal to PHP.

## *14.2 Cryptographic Keys and Key Management*

Cryptographic keys are not required because cardholder data is not stored. See explanation above.

Zen Cart® has never used or implemented cryptographic keys as part of the core code out-of-the-box. Thus there are no requirements to manage secure deletion or retirement of historical cryptographic keys.

If you have ever added a plugin/module to add cryptographic keys, it is your responsibility to securely protect and retire and otherwise manage those keys in accordance with PCI DSS requirements, and demonstrate such as part of your own self-assessment or certification process.

## *14.3 Protocols, Services, Dependent Software and Hardware*

The services and protocols and software components used by Zen Cart® are listed below.
**FOR DETAILS of specific version requirements of each, see sections 2.3, 2.4.**
To install and configure any of them, see the relevant documentation provided by their own authors.

- APACHE and PHP
  Zen Cart® is a PHP application which responds to requests directed to PHP via a webserver service running on your server. Alternate webserver engines could be used but are out of scope.
- MySQL
  Zen Cart® follows your configuration instructions to contact and connect to the MySQL service running on

the server you specify in your configuration, using the mysqli API in PHP.
If you choose to host your MySQL database on a separate server, you will need to open port 3306 in your servers' firewalls (or whatever alternate port you might reconfigure MySQL to use via my.cnf configuration file settings) to allow the database communication. Hosting your database on a separate server is not necessary to run Zen Cart, nor to be PCI Compliant.

- HTTP and HTTPS
  HTTP and HTTPS are used to communicate between customer browsers and the webserver, and from the webserver to any 3rd-party services.
- CURL and fsockopen
  Some external communication to 3rd party services is conducted via PHP libcurl (cURL) or the PHP fsockopen() client.
- TLS/SSL
  In the browser, HTTPS is used where needed to protect sensitive data if the storeowner has configured Zen Cart® to use it. In the background TLS is used (version autonegotiated via CURL) to protect sensitive data when transmitted to configured external services, provided the hosting webserver offers it. As such OpenSSL is needed for such external communications.
- REST APIs
  Zen Cart® v2.x does not currently serve a REST API, but does communicate with 3rd-party APIs, whether they be REST or SOAP, etc.
- Hardware
  Zen Cart® has no specific hardware requirements unique to the services it provides. It can run on virtually any server which can run the above required software.

## *14.4 Settings sensitive to PCI compliance*

If you want your site to be PCI Compliant, you must leave the following features set to their defaults:

- Admin->Configuration->My Store->PA-DSS Admin Session Timeout Enforced? – Should be ON (1)

- Admin->Configuration->My Store->PA-DSS Strong Password Rules Enforced? – Should be ON (1)

- Admin->Configuration->Logging->Log Database Queries – Should be off (false)

# 15. Additional Requirements for PA-DSS Compliance

## *15.1 Consequences of altering the system to store cardholder data*

Out-of-the-box, Zen Cart® does not store any cardholder data, and is crippled from being able to store entire card numbers by virtue of database field lengths being too short to store a complete card number.

**That said, if you or anyone with access to your site changes the database either by directly editing it or by installing a plugin which alters it to store cardholder data, you are invalidating the PA-DSS compliance and all such changes would need to be verified as implementing compliant procedures that are at least as rigorous as PCI DSS requirements..**

**You will need to ensure that your alterations comply with the PA DSS specifications, including but not limited to the following:**

a) When collecting magnetic stripe data, card validation codes, PINs, and/or PIN blocks for troubleshooting purposes be sure that you:

- Collect these data only when needed to solve a specific problem
- Store these data only in specific, known locations with limited access
- Collect as little of these data as necessary to solve the specific problem
- Encrypt these data when stored
- Securely delete these data immediately after use

b) If you add the ability to store cardholder data, then it is your responsibility to ensure that exceeding the acceptable retention period must be securely deleted, and it is your responsibility to demonstrate that you have studied and proven that you have securely deleted the data in compliance with PCI and PA-DSS requirements. Your activities surrounding secure deletion extend beyond the scope of this document and must include all subsystems and dependencies which you invoke when adding the ability to store such data

c) You must not alter the software to store cardholder data on Internet-accessible systems
(ie: web servers and database servers must not be on the same server, and the storage server must not be connected to the internet)

d) If you alter the application to allow and/or facilitate the sending of primary account numbers by end-user messaging technologies, you must render the PAN unreadable and/or implement strong cryptography

e) If you add functionality which manages encryption keys, for PCI compliance you must further demonstrate that you:

- restrict access to keys to the fewest number of custodians necessary
- store keys securely in the fewest possible locations and forms
- implement step-by-step procedures to generate, distribute, protect, change, store, and retire/replace encryption keys where customers or resellers/integrators are involved in these key management activities
- provide a sample Key Custodian form for key custodians to acknowledge that they understand and accept their key-custodian responsibilities
- ensure that cryptographic key material and/or cryptograms from prior versions of the application must be rendered irretrievable. Such irretrievability is absolutely necessary for PCI DSS compliance
- provide step-by-step procedures to re-encrypt historic data with new cryptographic methods and/or keys

## *15.2 Default Accounts*

PCI compliance requires that your server's default accounts (including but not limited to: the mysql "root" user, your server's "root" or "Administrator", etc) need to have secure authentication controls (secure complex passwords), even if the account is not used. For each of the "default" accounts, set a secure password. Then ideally those accounts should be disabled or not used (ie: don't login to your server using "root"; instead, disable “root” access and create another user with root privileges and a very secure complex password, and use that user for all server login).

**MySQL**
For example, your MySQL database must not use “root” without a secure password. In practice this default account should not be used at all. For instructions on Linux, see the Appendices later in this document, and for other server configurations, see the MySQL documentation online.

**Zen Cart**
There are no “default” accounts in Zen Cart. You create your own unique Administrator Login when running the zc_install setup script.
However, if you choose to install demo data and do not complete the final stage of zc_install to create an admin user, then there WILL be a default user named “Admin” with a password of “admin”. YOU MUST CHANGE that password for PCI compliance. To do this, login to your Admin panel, using that “Admin” user with password “admin”, and click the “Account” link in the top right corner (next to the Logoff link) and click the “Reset Pwd” button. Fill in the new password with a strong secure password (min 7 characters, with at least one capital and one digit) and click Update to save the new password. If you don't do this, a warning message will appear in your admin console until you do.

**Operating System**
Every operating system is different. For a Linux server, the “root” user exists by default. Whoever manages the server must be sure to set and use a strong password for this account. Or better yet, they really should disable root login altogether and use secure SSH keys and/or two-factor authentication tools. Securing an operating system is a much larger topic than this document. Consult the documentation for your operating system for instructions on establishing strong secure password controls for your specific system.

**Others**
Zen Cart has no other specific system dependencies which require passwords. But if your server makes use of any additional software tools for which a password is created, you must ALWAYS change those default passwords to use strong secure passwords. If you don't do this, then you're leaving open holes that hackers could potentially use as malicious attack vectors to damage your server or steal confidential data.

## *15.3 Strong Authentication Controls*

Reiterating the note in Section 5.1: it is required that you use strong passwords for accounts on dependent subsystems such as MySQL used to store data for the application.

That is: you must create a MySQL user (other than “root” or any other default value) and assign a strong password to that user, and then supply that username and password to Zen Cart to use.
And “root” should never be used – See sections 5.1 and 15.2.

In addition to strong passwords, it is recommended that you invoke two-factor authentication for access to your Zen Cart Admin and to your server's console.

## *15.4 Secure Access*

a) Firewall
In case it is not self-evident, you should always be using a securely configured firewall or personal firewall product on both your webserver and any personal computer with which you access your website.

b) Admin access (one component of “non-console access”)
Additionally, to further reiterate points made in Sections 2.1, 2.4 and 5.1, all non-console access (that is, any

access to your store's data or admin tools other than from the physical terminal screen connected to the webserver at the hosting company … so that means ALL access by you) to the Admin section must be done over HTTPS

c) FTP, SFTP, Remote Access   (another component of  “non-console access”)
Similar to the above point, **all remote access** (FTP, SFTP, all connections to the webserver for altering or accessing the files which run your store) must be done via secure channels such as SFTP or FTP-with-Implicit-TLS.

# 16. Appendices

## *16.1 MySQL Root Password Reset*

As explained previously, for PCI Compliance, you must NOT allow MySQL to use the default (blank) password for the "root" account. To change the "root" password, follow these instructions:

*From the MySQL documentation:*

**C.5.4.1. How to Reset the MySQL Root Password**

If you have never set a `root` password for MySQL, the server does not require a password at all for connecting as `root`. However, this is insecure. For instructions on assigning passwords, see Section 2.10.2, "Securing the Initial MySQL Accounts".
If you know the `root` password, but want to change it, see Section 13.7.1.6, "`SET PASSWORD` Syntax".
If you set a `root` password previously, but have forgotten it, you can set a new password.

**C.5.4.1.3. Resetting the Root Password: Generic Instructions**

The preceding sections provide password-resetting instructions for Windows and Unix systems. Alternatively, on any platform, you can set the new password using the **mysql** client (but this approach is less secure):

1. Stop **mysqld** and restart it with the `--skip-grant-tables` option. This enables anyone to connect without a password and with all privileges. Because this is insecure, you might want to use `--skip-grant-tables` in conjunction with `--skip-networking` to prevent remote clients from connecting.
2. Connect to the **mysqld** server with this command:

```
shell> mysql
```

3. Issue the following statements in the **mysql** client. Replace the password with the password that you want to use.

```
mysql> UPDATE mysql.user SET Password=PASSWORD('MyNewPass')
    ->                   WHERE User='root';
mysql> FLUSH PRIVILEGES;
```

The `FLUSH` statement tells the server to reload the grant tables into memory so that it notices the password change.
You should now be able to connect to the MySQL server as `root` using the new password. Stop the server, then restart it normally (without the `--skip-grant-tables` and `--skip-networking` options).

## *16.2 Password Security in Zen Cart®*

Passwords for both customers and administrators are an important subject. Zen Cart® does not store unencrypted passwords, and as such can never reveal a user's password to that user or to a third party.

Passwords are one-way encrypted by generating a 128-bit salted bcrypt (blowfish) hash where bcrypt is available, or alternatively with a SHA256 hash using a salt of equal length.

Prior versions of Zen Cart® implement a salted MD5 hash in a one-way encryption algorithm. Old passwords are automatically updated to the new encryption method for greater security.

It is advisable to use the latest supported version of PHP for maximum security.

To maintain PCI DSS compliance, any coding changes made to authentication or password code or configurations would need to be verified as providing authentication methods that are at least as rigorous as PCI DSS requirements.

## *16.3 Wireless (WiFi) Networks*

If you are using a wireless network to access your online store, it MUST be configured securely.

- Your wireless router should use industry-standard best-practices for strong encryption and transmission, such as the popular IEEE 802.11.i specification. (Most routers already do this, but you should verify it.)
- The wireless network needs a password.
- DO NOT USE the default password written on a label on the side of the router, nor one listed in the manual as the default or original wireless password.
- The wireless network must use a strong secure complex password. Not something easily guessed, and something at least 7 characters long, with at least one digit and preferably some uppercase letters and

symbols.

- DO NOT USE “WEP” security mode. Use WPA2 with TKIP mode if possible. (AES is less secure.)
- Change the default administrative “user” (often called “admin” or “cusadmin” or “root”) and use a strong complex password, not the same as the wireless password above. Minimum 7 characters, with at least one digit and preferably some uppercase letters and symbols.
- SNMP community features should be disabled. Or the SNMP community names should be changed; don't leave the names set at defaults.
- Secure the router in a place where disgruntled employees or unauthorized persons will not be able to gain access to the physical device. Nobody should be able to access the device's reset button without authorization, else they might be able to compromise your security.

**Any time you “reset” your wifi router (ie: typically done by pressing the little “pinhole” button) you will need to do all the password configuration on it again.**

**Anytime your employees change (anyone with knowledge of or access to the passwords leaves), you will need to do all the password configuration again, using new passwords.**

These are common wireless security settings and practices that should be implemented to ensure the security of your online business. They are not specific to Zen Cart® but nevertheless should be implemented for your own security.

# 17. Implementation Guide Changelog

| Date | Version | Changes from previous version |
|---|---|---|
| 10th Oct. 2010 | 1.4 | Minor grammatical fixes and added section 7 |
| 22nd April 2011 | 1.5 | Added sections regarding admin users/profiles and PA-DSS regulations related to passwords (section 8 et al)<br>Added SFTP guidance and explanation of unzip, FTP/SFTP<br>Added section about engaging 3rd-party services vs FTP etc<br>Added discussion about removing historical PAN/CVV data |
| 11th May 2011 | 1.6 | Add new sections for Network and dataflow diagrams<br>and notation to indicate that credit card gateway modules will not function if HTTPS is not enabled. |
| 17 May 2011 | 1.7 | Added section explaining how to integrate a two-factor authentication system if required in the merchant's environment<br>Added section explaining how to securely erase old CHD<br>Added link to instructions for enabling SSL for entire admin<br>Added explanation that the initial admin password is only temporary and must be changed on first login. |
| 24 August 2011 | 1.7d | Update admin-rename instructions, and for upgrading (5.2)<br>Add section on reviewing admin activity logs (8.5.1). |
| 20 Sept 2011 | 1.7f | PA-DSS Remediation, plus updating trademark symbols |
| 26 Sept 2011 | 1.7g | Added section 2.1, expanded section 4, and clarification to 5.1(2) |
| 9 Nov 2011 | 1.8 | Improvements to consistency. Updated “upgrade” section text. |
| 8 Aug 2013 | 1.9 | Updated for PA-DSS v2 requirements, and to add new /logs/ folder, Stamp ZC v1.5.2. |
| 20 Aug 2013 | 1.9.1 | |
| 22 Oct 2013 | 1.9.2 | Updates to clarify wording relevant to PA-DSS v2.0 requirements |
| 3 Dec 2013 | 1.9.3 | Updates to comply with additional wording required by PCI rules. |
| 20 Feb 2014 | 1.9.4 | Added a copy of the generic MySQL root-password reset instructions. |
| 14 Mar 2014 | 1.9.5 | Updates to comply with future PA-DSS v3.0 requirements |
| 20 Apr 2014 | 1.9.6 | Version update for ZC v1.5.3 |
| 4 July 2014 | 1.9.7 | Misc updates requested during PA-DSS review |
| 4 July 2014 | 1.9.8 | Version update for v1.5.4; updated section on activity logging |
| 13 Aug 2014 | 1.9.8.2 | Reminder to make the server log-export folder writable |
| 14 Nov 2014 | 1.9.8.3 | Added reminder to always download new versions via HTTPS |
| 30 Apr 2015 | 1.9.8.4 | Updated “SSL” references to “SSL for HTTPS” for clarity. |
| 15 Mar 2016 | 1.9.8.5 | Version update for ZC 1.5.5, including updated zc_install screenshots<br>Version update for ZC 1.5.6<br>Version update for ZC 1.5.7 |
| Nov 2018 | 1.9.8.6 | Version update for ZC 1.5.8 |
| June 2020 | 1.9.8.7 | Version update for ZC 2.0.0 |
| Oct 2022 | 1.10.0.1 | |
| April 2024 | 1.10.2 | |