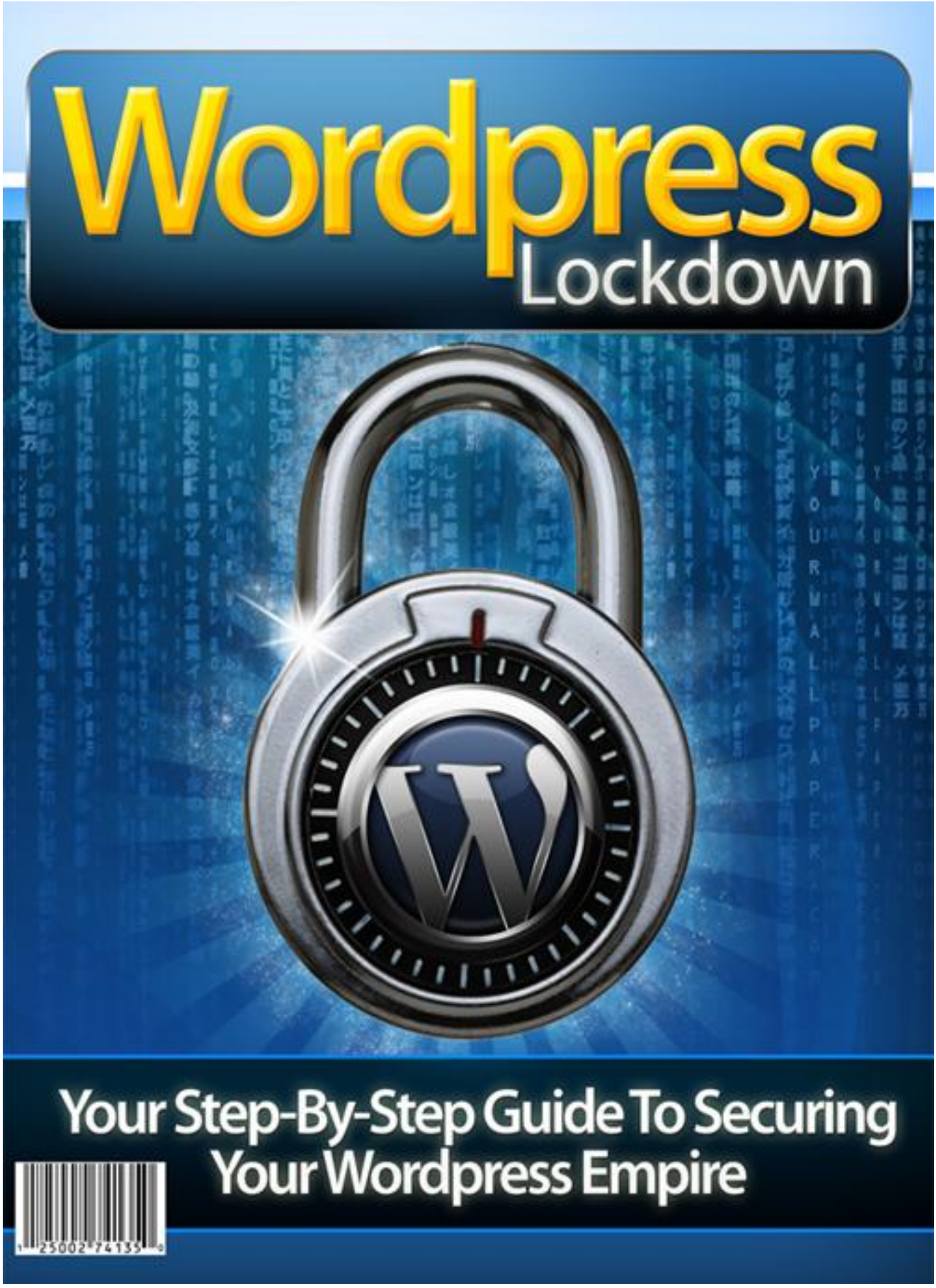
Wordpress Lockdown



Your Step-By-Step Guide To Securing Your Wordpress Empire

Legal Notice

The author and publisher of this Ebook and the accompanying materials have used their best efforts in preparing this Ebook. The author and publisher make no representation or warranties with respect to the accuracy, applicability, fitness, or completeness of the contents of this Ebook.

The information contained in this Ebook is strictly for educational purposes. Therefore, if you wish to apply ideas contained in this Ebook, you are taking full responsibility for your actions.

The author and publisher disclaim any warranties (express or implied), merchantability, or fitness for any particular purpose. The author and publisher shall in no event be held liable to any party for any direct, indirect, punitive, special, incidental or other consequential damages arising directly or indirectly from any use of this material, which is provided "as is", and without warranties.

As always, the advice of a competent legal, tax, accounting, IT or other professional should be sought. The author and publisher do not warrant the performance, effectiveness or applicability of any sites listed or linked to in this Ebook. All links are for information purposes only and are not warranted for content, accuracy or any other implied or explicit purpose.

Contents

Who Should Read This Book?

If you …

- Happen to be a proud owner of a WordPress blog…
- Happen to be an enthusiast who wants to learn more about how WordPress works…
- Happen to be concerned about hackers getting a hold of your blog, and what those hackers will do with it…

…this book was especially written for you.

This book targets Wordpress 3.x specifically, however the general principles and practices are applicable to all versions of Wordpress. Besides, as you will learn from this book, why the heck would you be running an older version of Wordpress anyway?

Packed with critical security information and strategies, this guide provides you with the information you need to secure your blog and keep it secured. It does not contain any hype, sales pitch, or biased affiliation with any product or service. Nor does it contain misleading information or any affiliate links.

It instead, contains a no-bones-about-it approach to securing a WordPress blog with specific tools and some damn good reasons for using them. I decided to publish this information in an effort to answer some hard questions about what WordPress security means, with it requires, what it looks like, and how it functions.

We're now witnessing an increase of abuse with this particular blog platform at a phenomenal level, which is annoyingly frustrating because we know this abuse could be prevented. It is therefore, my sincerest hope that you will read this guide, follow its suggestions, and essentially do your part to improve the quality of blogging that we all seek from the Internet.

As Internet Marketers we spend a lot of time worrying about search engine rankings, link building and traffic. Unfortunately, there is not often time to think about the security aspects of operating in one of the most hostile business environments on the planet. Often times, our minds only turn to these topics when we have actually been hacked and our precious income earning sites have been hijacked or destroyed. Worst of all, compromised websites are

quickly noticed by search engines and are either de-indexed or have their ranking severely penalised - don't let this happen to you!

This book is for those who already know blogging basics, but want to make their blog(s) a secure and spam free Internet phenomenon.

It is true that this guide does not cover every conceivable aspect of Wordpress security or provide copious amounts of detail on each topic covered. Nor does it cover everything there is to know about cyber security in general - in order to cover this subject in depth one would need many millions of pages indeed!

This guide does, however, pick out those security principles and practices related to the Wordpress platform that provide the best bang for the buck, or the best security and piece of mind for your blogging empire for the least application of effort and time. The techniques discussed in this guide are tried and true, used by countless professionals around the planet - and used by myself as well - to improve the security of their Wordpress sites.

Thank you for reading and may your Wordpress empire be more secure and resilient!

How to Use This Book

To signify different types of information, I have used different styles and typographic conventions throughout the entire book.

Code Samples

Code samples are printed on a highlighted background to make them stand out from surrounding text. Example:

```
<?php
if ( defined('WPCOM_API_KEY') )
      $wpcom_api_key = constant('WPCOM_API_KEY');
```

Important Notes

Important notes are also printed on a highlighted background with outlines around them.

This box means the note is important or useful information is given to the reader

Mini Points

To emphasize the importance of certain topics or points too small to warrant their own box or color, I've inserted several icons throughout this guide. Here's a brief table, which outlines what each of these icons mean.

	When you see this icon, it means that the strategy right next to it is a guaranteed winner - a strategy that wins you a guaranteed search engine position and higher web traffic because of appropriate security measures.

	This icon is meant as a warning. Don't do what's written next to this time bomb or you'll be sorry… in a matter of… well, time!
	This picture indicates a definite killer. Heed the warning accompanying its text if you want to stay alive - well, your blog anyway.
	This icon marks important information.
	This picture describes software you can download from the Internet, and use to implement some of the strategies outlined within.
	Here's a pointer toward information on the World Wide Web.
	This marker points to a last minute thought - one that I didn't want to let get away before publishing this guide! Includes tips, ideas, and small notes.

Introduction

Like you, thousands of people depend on WordPress as a blogging platform. WordPress is the most popular blogging platform on the internet because for the most part, it's easy to use, convenient, and robust, making it suitable for all websites of any size. However, the growing number of WordPress blogs gives hackers an almost unlimited playground for wreaking havoc.

Once a WordPress blog is hacked, it takes a huge amount of work to clean up the mess, which can include site redirects, corrupted databases, and worst of all, financial loss. In this book, I will introduce some critical protection measures designed to defend your blog against these types of destructive activities and more.

You'll discover:

- Why you need to secure your blog.
- How to secure your own computer and how that contributes to your blog's security.
- What makes an appropriate web host.
- The secure way to install a WordPress blog.
- How to securely update a live blog.
- What information you need to keep private.
- How to secure your MySQL database.
- Common sense security measures people take for granted.
- How to back up your security efforts.
- Ways to prevent your blog from comment spam.
- What to do when you've been hacked.
- Resources for additional help.

Let's take a closer look at why this tremendously popular and robust blogging platform needs a little extra security boost on your part.

The Need for WordPress Security

More than 25 million[1] people use WordPress as their blogging platform because it really is one of the best platforms available. But as suggested earlier, a wide user-base doesn't make it less vulnerable to attacks. A large user-base actually makes it a premium target for attacks since the result of hacking attempts affects a significant portion of the online community. This is the same principle why the Windows operating system has more viruses written for it than the Mac - because there are more Windows boxes out there and the potential to successfully compromise more systems is greater.

The good news is that WordPress developers are quick to repair bugs and keep your blog up-to-date with improved coding. As soon as vulnerabilities are discovered, the WordPress community releases relevant patches to prevent hackers from exploiting those vulnerabilities. And this is why it's so important to keep your WordPress software updated.

Running the latest version minimizes risks to successful hacking, however, it isn't fool-proof. There are additional things you must do to minimize risks – things that I describe in detail throughout this book.

The Risk of Being at Risk

When a blog is hacked, it becomes the victim of all sorts of horrifying events. It could be deleted, replaced with inappropriate materials, or redirected to an identity theft or 'phishing' site. These are, of course, just a few examples. There's really no limit to what hacking could do, and to be honest, there are too many examples to list here. The most important thing you need to do at this point, is understand the ultimate risk of being hacked: ruining your business through a loss of reputation.

The typical blogger toils away in hopes of selling products and/or building a thriving, interactive community, understanding that these things won't happen unless there's a history and reputation of trust between the blog administrator and the blog user or customer. Once that history and reputation is compromised, sales falter and communities dissipate. A compromised history and reputation also destroys any chance of establishing trust with new users.

[1] http://en.wordpress.com/stats/

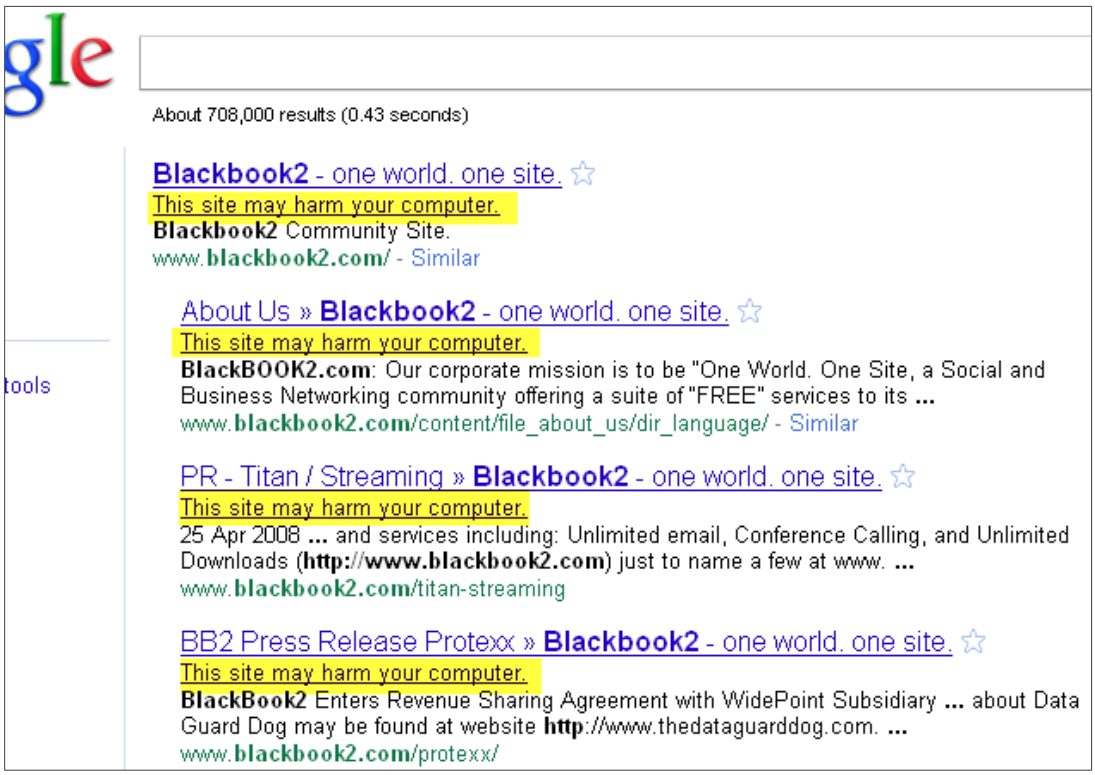
The bottom line is, your blog is a target for hackers no matter who you are or what your blog is about. Hackers don't choose their targets based on the content of a blog or who runs it - they choose them based on the resources they offer. They want to steal the use of your web platform, which means that no-one is exempt from their deeds.

How Web Browsers and Google Communicate Your Trustworthiness (Or Rather, Lack of)

When a blog is hacked, web browsers equipped with a malicious site detection system don't hesitate to spread the word. Visitors who've never heard of your blog before, but who may happen to discover it through a search engine, social networking profile, or forum link, for example, are warned to avoid your blog with a notice similar to the illustration below:

Firefox blocking a site with malicious content.

Web browsers aren't the only devices that can scare people away from your blog. Google has its own malicious site detection system too, and should your blog demonstrate hacked behavior, its search engine displays a similar warning to the following:



Having your blog associated with these types of warnings damages your reputation and prevents any chance of becoming a successful blogger.

What Makes Your Blog Attractive to Hackers

First things first, you must understand that the most common type of hacking isn't always personal. Only in extreme cases, like the case of Wikileak's Anonymous fan base, is hacking the result of a personal vendetta. In general, hackers will attack *any* blog simply because they exist as opportunities to wreak havoc and/or earn money for themselves.

Think about the common thief as an example. Thieves rarely rob stores because they refused them excellent customer service. Thieves robs stores because they house cash and products. In other words, thieves rob stores because of the opportunities they provide.

It's the same with blogs (vulnerable blogs, that is). Hackers attack vulnerable blogs because of the opportunities they provide. The vulnerable blog gives hackers:

- **Free storage space.** With free storage space, hackers can host malicious scripts and distribute them under a trusted site. Hackers aren't like you and me. You and I set up blogs with our own credit cards. Hackers can't do that because credit card information

will identify them and hold them responsible for the damage they do. Instead, hackers do their dirty work through other blogs because it's safer for them!

- **Free redirecting tools.** To ensure a blog is properly indexed in search engines, bloggers employ an .htaccess and robot.txt file. The robot.txt file tells engines, "Hey there, please list my blog and all its cool stuff in your index." while the .htaccess file tells search engines "Oh, and here's how it should be indexed." When search engines encounter these files, they follow their instructions, which could tell engines to index an entire site, ignore specific parts of it, or load certain pages instead of others. Now just imagine what could happen when a hacker gets a hold of these instructions. Instead of following a blogger's instructions, search engines will follow a hacker's instructions.

- **Free access to advertising money.** A lot of blogs use AdSense and other similar programs as their monetization method. When those blogs are hacked, attackers can change specific parts of those programs' code, and literally redirect a blogger's earnings to their own accounts.

- **A free domain name.** Through DNS hijacking**,** hackers can change your DNS record (domain name information) to point to a blog that imitates yours. When visitors reach the imitation blog, they assume it's legit, and use it as a trusting user. All information entered into an imitation blog is passed on to the hacker, which could include user names, passwords, bank details, and more. Preventing this sort of attack is beyond the scope of this guide, but it is worthwhile to note here so you understand the sophistication of some hackers.

- **A free platform for spamming.** Blogs that allow commenting are especially vulnerable to comment spam: valueless comments that promote other websites. Often, this type of spamming is robotic, and keeping up with it can be a huge pain in the neck – so much so, some blog owners remove the opportunity to comment because they just can't control the spam.

As you can see, blog security is a very serious deal. Let's look at the role your own computer plays in WordPress security. It's the very first step you should take in securing your blog since bloggers can unknowingly contribute to their own vulnerability. Have you ever thought about the security of your home computer affecting the security of your blog sitting out there in cyber-space?

Securing Your PC

Anti-Virus Protection

Key-loggers are programs that record the strokes you make with your keyboard.

Created and distributed by unscrupulous individuals, some key-loggers upload your activity to a remote, identity theft database.

Hackers can plant a virus on your computer, like a key-logger of some sort, and steal your blog's information (username, password, email addresses, etc.) for their own use when you connect to your blog. This is what makes securing your own computer against viruses extremely important. There's probably nothing worse than knowing your own computer contributed to your blog's hack attack, so don't ignore this critical step.

Fortunately, there are tons of anti-virus programs available, and if you cannot afford to buy one, you absolutely must use a free antivirus program for minimum protection. Once you're able to afford it, upgrade for maximum security immediately. The difference between paid and free anti-virus protection can be significant.

Comcast Internet subscribers may access a free, commercial version of the Norton™ Security Suite here: http://security.comcast.net/get-protected/index.aspx

Free anti-virus protection is better than nothing! Just pony up the doe when you can because viruses are no joke, and they're getting harder to remove every day. If you can, try to find an anti-virus package that includes a Internet Security Suite of tools such as a firewall. This will increase the robustness of your system significantly.

Here's a screenshot of Avast (paid version) in action, blocking an attack.

Kind of hard to resist buying anti-viral software when you can see the protection they provide. Consider yourself warned.

I personally use a product by G-DATA that I personally believe is the best on the market. You you can learn more about here if you are interested: http://www.gdata-software.com/home-security/internetsecurity-2011/. Whatever you choose to use, free or paid, just make sure you go and get something NOW.

Some of the more popular commercial anti-virus programs are Kasparsky (http://www.kaspersky.com), McAfee (http://www.mcafee.com), and Norton (http://www.symantec.com). Avast (http://www.avast.com) and AVG (http://free.avg.com) are popular freeware programs.

Automatic Updates

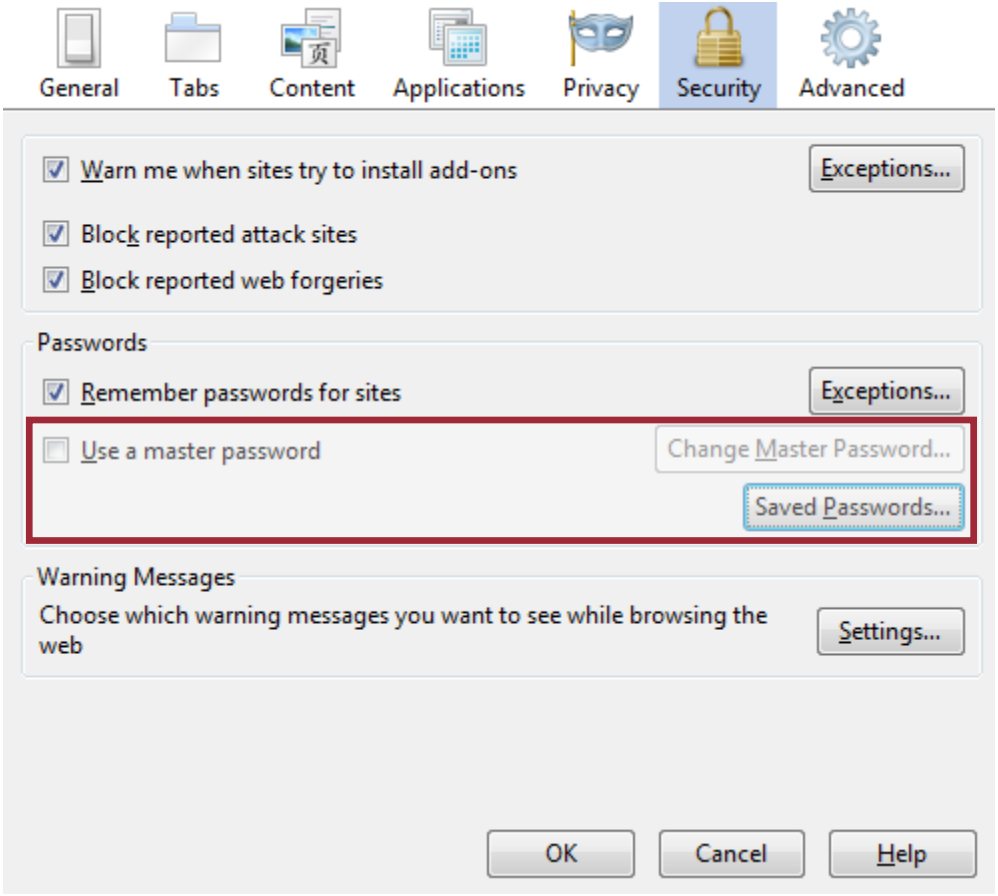
Just like with your blog, you must ensure that you are always running the latest versions of all of the software you use on your computer. Software vendors are continually plugging holes in their products that hackers have used to exploit systems, and you need to make sure you take advantage of these updates. If you fail to run the latest version of your software, you might as well open a large door to your computer for hackers to enter and steal your livelihood.

Most software have an option to automatically check for new updates - make sure this is tuned on for all of your software. This includes your underlying operating system as this is one of the largest attack surfaces available.

Password Management

With your computer now relatively secure, you now need to secure your behavior. That is, you need to develop a password system that is more difficult to exploit, and use a different password for every website that requires one. That's right - from now on you will NEVER use the same password twice! One of the easiest ways to give hackers access to your password(s) is to use the same one all over the web. The minute hackers gain access to the password, they can log in to all of your accounts and destroy everything you've built.

It's admittedly difficult to remember a lot of different passwords, especially if you log into hundreds of different websites. But it's a non-optional requirement. Make things easier on yourself by keeping a secured password database on your computer, or by using a password manager, such as the one offered in FireFox.



A password manager helps you generate strong passwords, and it keeps them secure with encryption. You need only remember one master password with a password manager. This

master password automatically retrieves other passwords from a protected database with high grade encryption.

A **brute-force attack** is a technique used to break an encryption or authentication system by trying all possibilities. (Source: www.cuhk.edu.hk/itsc/security/isglosry/index.html)

The best passwords contain capital letters, lowercase numbers, and symbols. They're not your pet's name, friend's or spouse's name, a dictionary word, movie title, or anything that can be easily guessed. Hackers not only keep a list of frequently used passwords, they use brute-force attacks to discover new passwords. And thanks to the laziness of the general populace, passwords are, quite frankly, discovered with minimal effort.

Examples of poor, weak passwords: mypass, letmein, passwords, myname, 123456, 111111 etc.

Examples of strong, hard to guess passwords: U#$oi9fd4, 1f^&%84fd, FD5@4)6w_e1, etc.

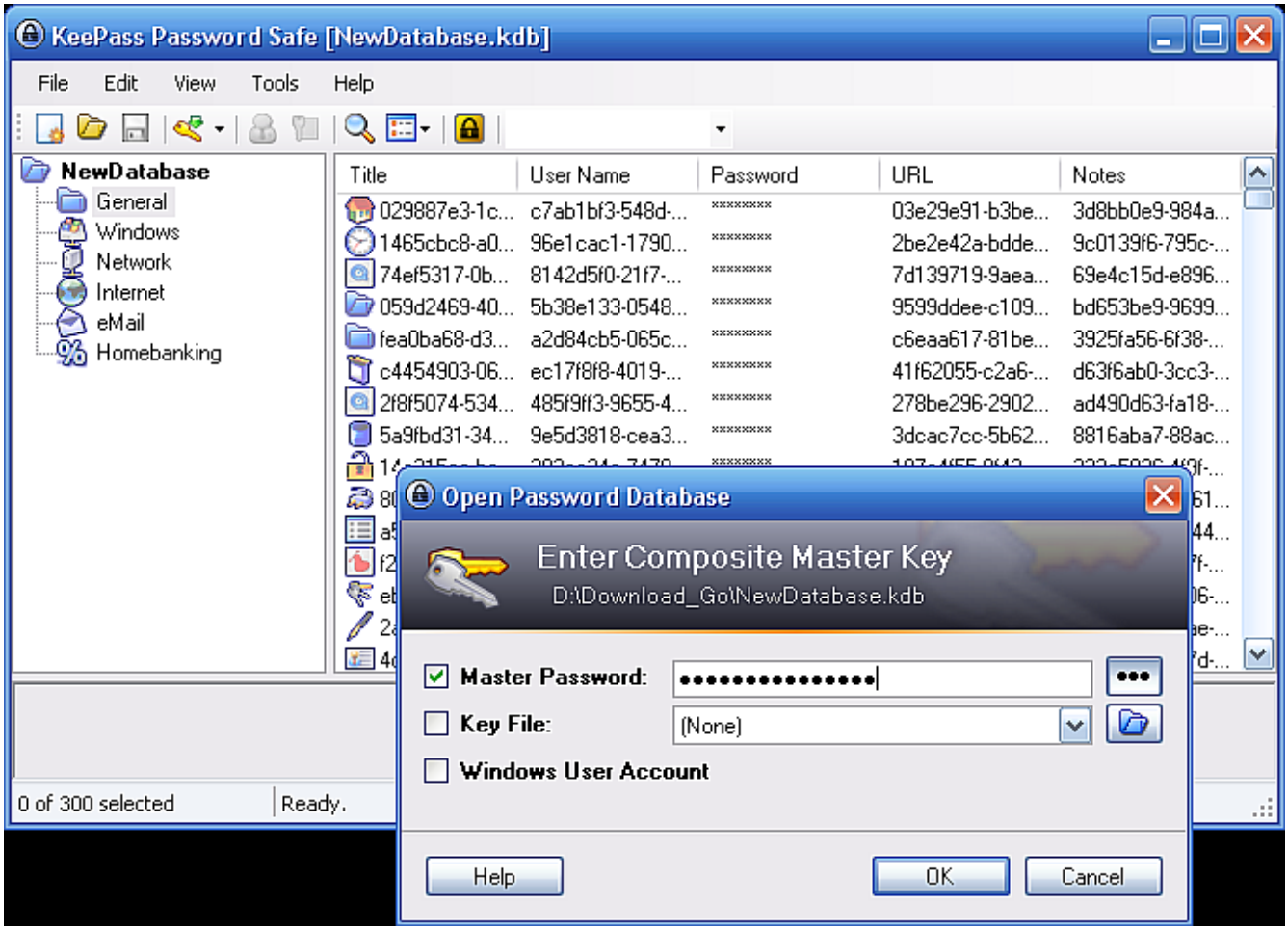
If your favourite browser doesn't come with a password manager, you can download one that will work with it. Here are a few suggestions:

KeePass

Although KeePass is open source, it's not to be underestimated. It is the only manager I use and I highly recommend it. It's open source so industry experts can properly verify implementation of its encryption algorithm – a feat not possible with commercial password managers.

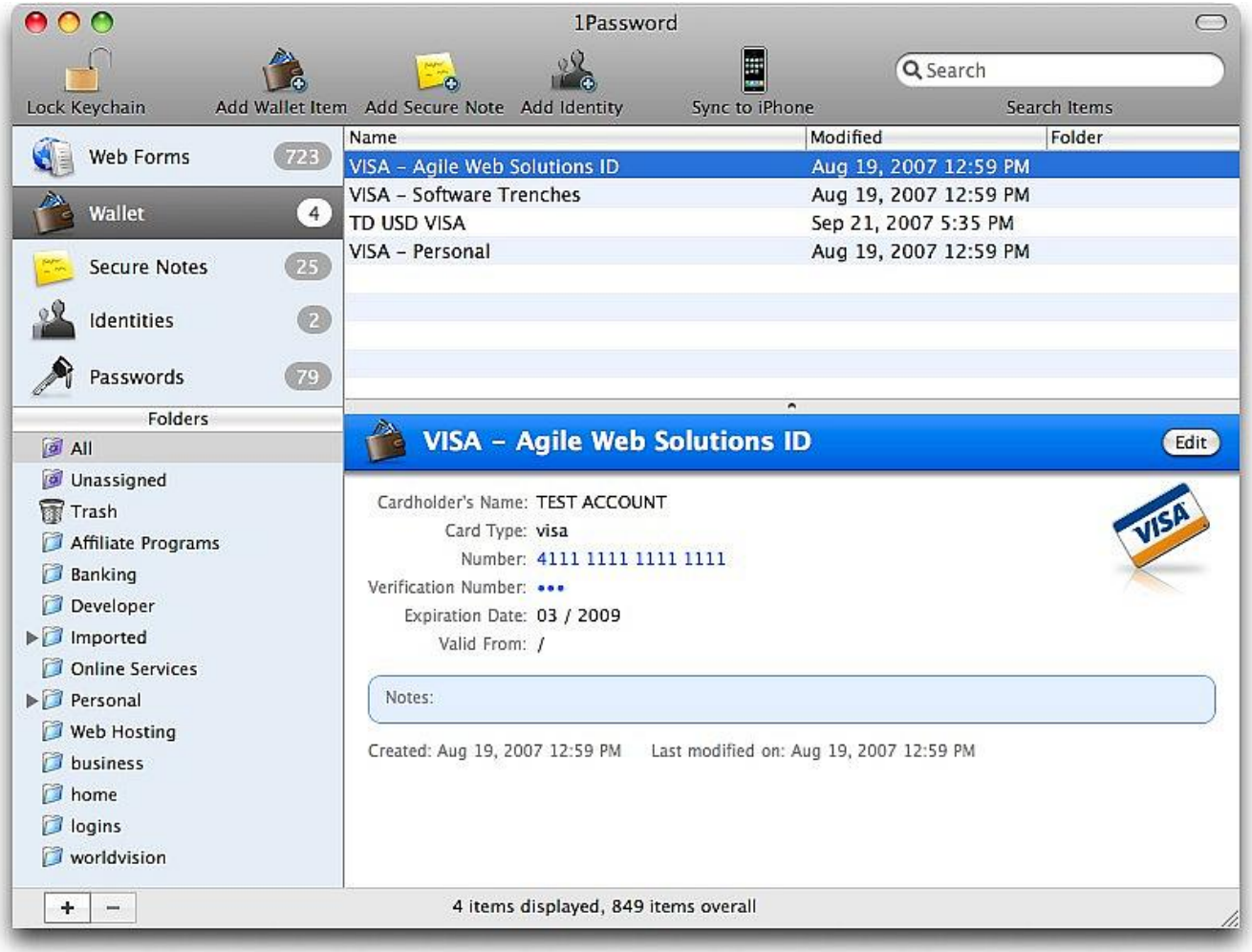
encryption algorithm - a mathematical formula used to encrypt or decrypt a string of text. (Source: www.cafesoft.com/support/security-glossary.html)

KeePass is available for all operating systems, including Windows, Mac, *and* Linux. It's also available for portable computers, iPhones, PocketPCs, and Android phones. Get KeePass here: http://www.keepass.info.



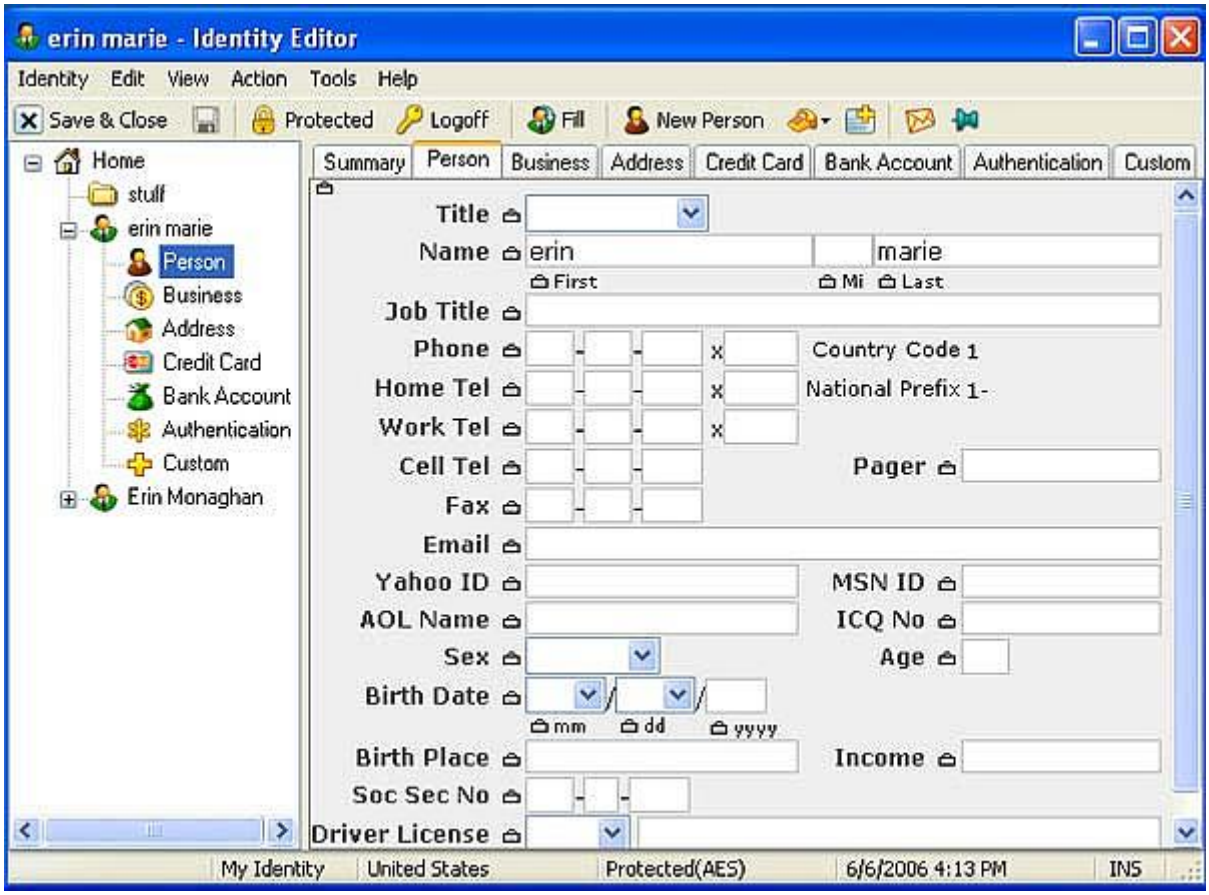
1Password

1Password is a commercial password manager for Mac, Windows, Palm and iPhone. It integrates well with web browsers, and it includes a form filler to speed up site registrations. But it does more than just store passwords. It also stores notes, software licenses, credit cards, and attachments. A single user license is all you need to install 1Password onto al your PCs and mobile devices. Get 1Password here: http://agilewebsolutions.com/products/1Password.



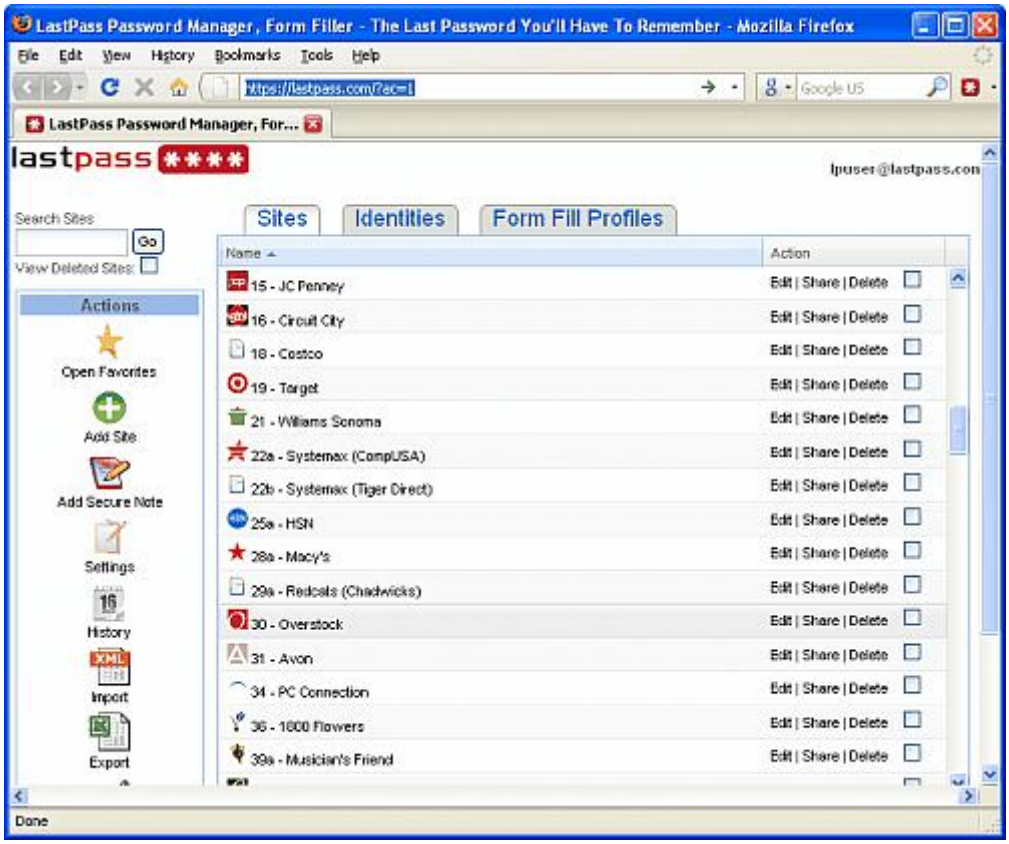
RoboForm

Like 1Password, RoboForm offers a web form filler that completely automates entering and form filling. But unlike LastPass (described below), it has a graphical user interface. RoboForm works with Internet Explorer, Firefox, Google Chrome, Safari, and other browsers. It also works on the most popular mobile devices: iPhone, Andriod, BlackBerry, Windows Mobile, Symbian, and the Palm. Portable users can take advantage of this program as well. Get RoboForm here: http://www.roboform.com.



LastPass

Although LastPass is available for all popular personal computer and mobile platforms, it won't work without a browser because it stores passwords in "the cloud." (That's a fancy way of saying, "on the Internet"). Passwords stored online give you immediate access from any device that has a web browser. And though the idea of storing passwords online doesn't sound very secure, LastPass actually encrypts your passwords on *your computer* before they're uploaded to its server. That means not even LastPass.com can see your passwords. Get LastPass here: https://lastpass.com.


LastPass Password Manager, Form Filler - The Last Password You'll Have To Remember - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://lastpass.com/?ac=1
LastPass Password Manager, For...
lastpass ****
lpuser@lastpass.com
Search Sites
Go
View Deleted Sites:
Actions
Open Favorites
Add Site
Add Secure Note
Settings
History
Import
Export
Sites
Identities
Form Fill Profiles
Name
Action
15 - JC Penney
16 - Circuit City
18 - Costco
19 - Target
21 - Williams Sonoma
22a - Systemax (CompUSA)
22b - Systemax (Tiger Direct)
25a - HSN
28a - Macy's
29a - Redcats (Chadwicks)
30 - Overstock
31 - Avon
34 - PC Connection
36 - 1800 Flowers
39a - Musician's Friend
Edit | Share | Delete
Done

A Secure Web Host

Without a secure web host, your efforts to secure your own computer will be for naught. So here, we want to talk about what makes a hosting company a secure one since so many companies differ in what they offer.

Costs

As a conscientious consumer, you're naturally going to want to get the best deal for your money. That's why you might be inclined to purchase hosting at the lowest price you can find. If there's one thing we want to make clear, it's that cheap hosting is the last consideration you want on your priority list.

Cheap hosting is cheap for a reason. And that reason is that they don't offer all the protections available. (They often can't afford to). They also cram you into overcrowded shared-servers which increases your attack surface. What you want to prioritize instead, is security, and if that costs a little more than what you wanted to pay, go ahead and make the investment. Your blog is worth it.

PHP and MySQL

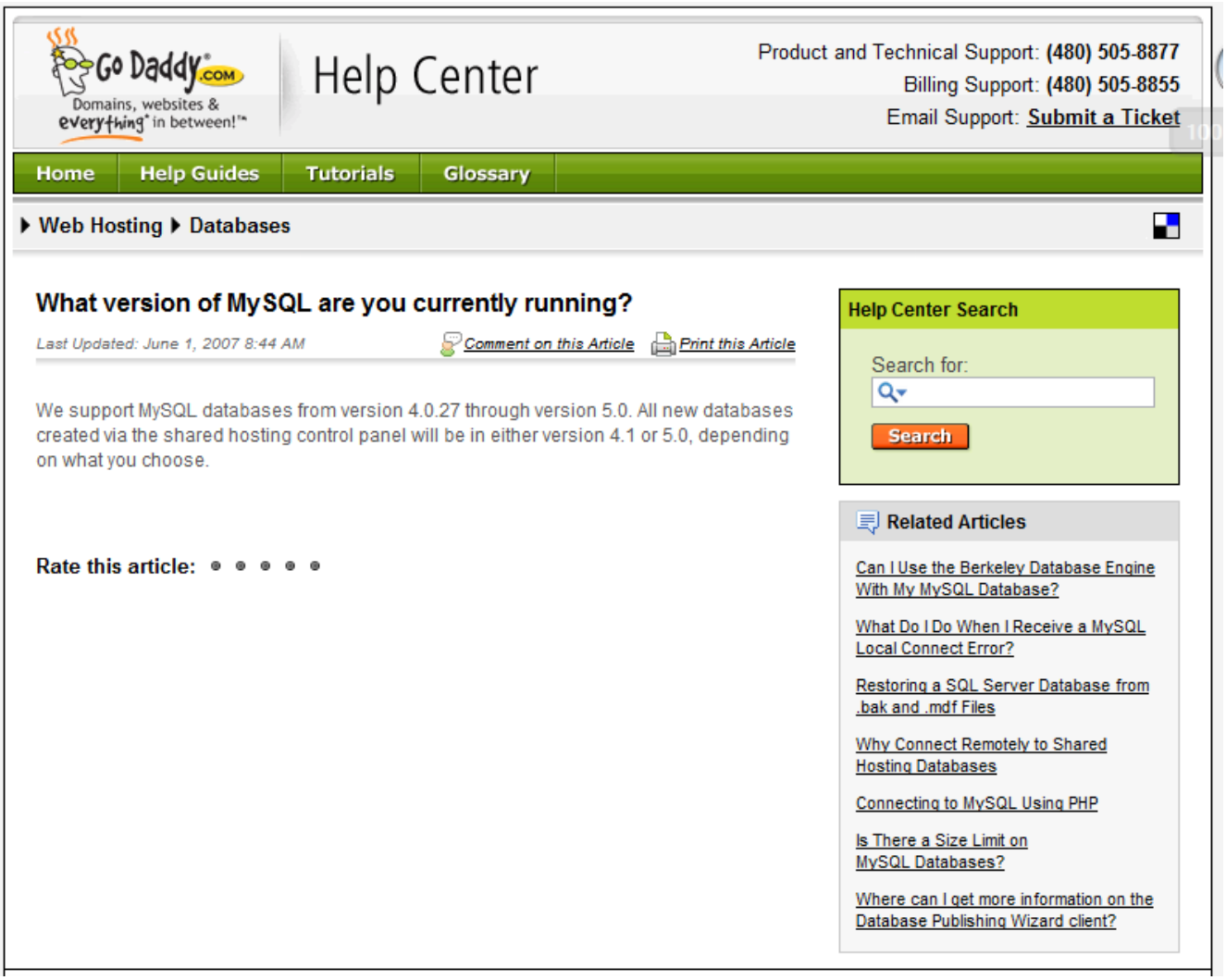
Never purchase hosting that doesn't run a recent version of PHP and/or MySQL!

Secure hosts regularly update their operating systems and apply necessary patches. When a cheap web host runs an outdated version of PHP and/or MySQL, all the blogs it serves becomes vulnerable to bugs which hackers use to exploit.

Never purchase hosting that doesn't run a recent version of PHP and/or MySQL. If you are in doubt, snoop around a service's FAQ or simply shoot the company an email and ask. Then verify what you're told with the latest releases at the official PHP and MySQL websites.

Official PHP website: http://www.php.net

Official MySQL website: http://www.mysql.com



Example GoDaddy FAQ addressing its MySQL versions.

Be aware that a secure host may not immediately run a new release. New releases of both PHP and MySQL may contain bugs that need fixing, and responsible hosts will hold off making these releases available for your own protection. Web hosts are usually reluctant to update unless there's a good security reason for doing so.

That doesn't mean a huge gap between versions is acceptable, however. On the one hand, you don't want to host with a provider that's running five year old software. On the other hand, you don't want to host with a provider that updates its software every day since the mere process of updating can cause downtime and other problems mentioned above without proper testing.

Secure File Transfer Protocol

After securing your computer against vulnerabilities (viruses, weak passwords, etc.) and hosting with a modern provider, your next step is to secure the connection between your computer and your blog. This step is easily accomplished by using current FTP (File Transfer Protocol) software.

File Transfer Protocol (or FTP) software is software that allows users to copy files between their local system and any system they can reach on the network.

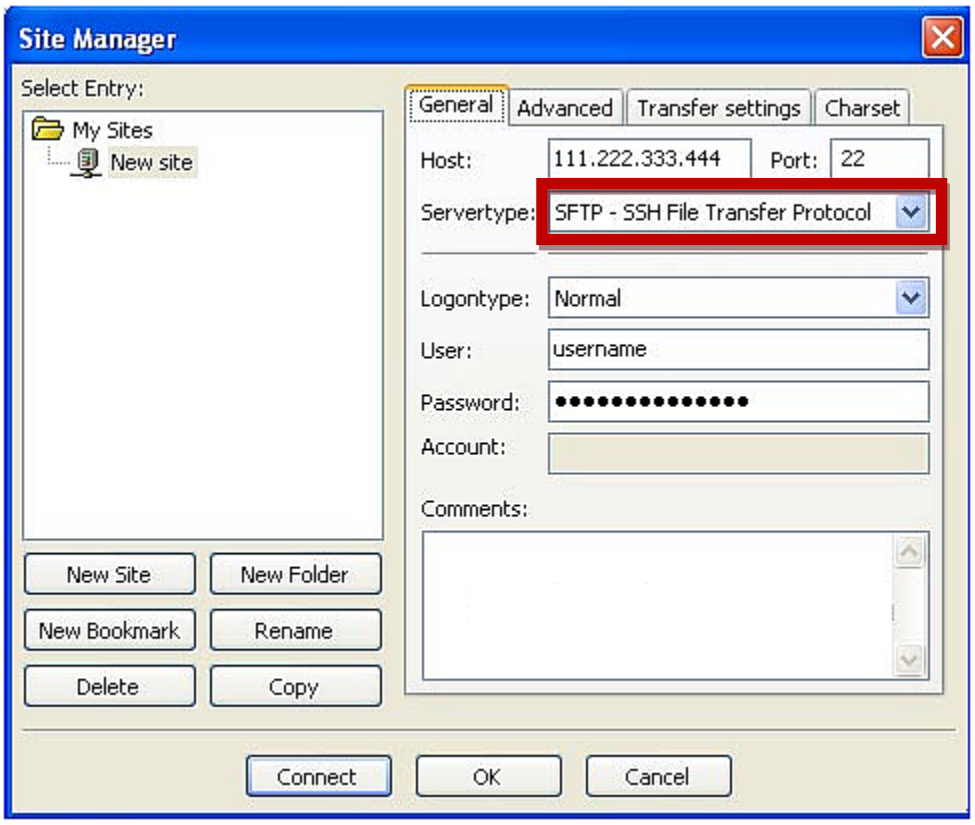
In the past, ordinary file transfer protocol software was adequate because hacking just wasn't as wide-scale as it is now. Unfortunately, a lot of people assume the same software is appropriate for today's use, despite the fact that the software they're using is probably over 20 years old and doesn't provide today's necessary protections.

Earlier FTP software, which was never designed to provide secure connections in the first place, doesn't encrypt connections. It, in fact, sends a user's username and password to a remote computer in clear, readable text. If communications are monitored by a Trojan or intercepted anywhere between sent and received signals, usernames and passwords can easily be retrieved and used by a hacker to access a site.

It is, therefore, imperative that you use up-to-date, secure FTP software when connecting to your site and transferring files.

A secure alternative to bare FTP is FTPS, FTPS-SSL or Secure FTP. Secure FTP uses the same technology your bank uses to protect you online. But this relationship takes two to work. In other words, you not only need to use it, your host also needs to support it. So just like you'll verify a host's current PHP and MySQL usage, do the same with its file transfer protocol. Make sure every host you're interested in using provides a secure connection, and then get your hands on software that will use this connection.

I recommend and FileZilla for all of your SFTP needs.



Example FTP software that offers a secure connection option.

Different FTP programs may offer this option in different locations or as different options, though the procedure of securing a connection is pretty much the same.

1. Open the FTP program
2. Open Options
3. Provide your FTP server details
4. In Server Type: select SFTP, SSH, or FTP over explicit TLS/SSL according to what your host supports.
5. Press **OK** to save

A successful, secure connection will depend on the type of connection offered by your host, and connections **must be compatible**, else a connection can't be made.

For more, in-depth information about SFTP, SSH, or FTP over explicit TLS/SSL connections, see http://en.wikipedia.org/wiki/SFTP.

Secure Your WordPress Installation

WordPress comes with an installer, which is very easy to use. Problem is, many hosts offer WordPress installation with a one-click option, which doesn't give you opportunities to tweak your blog the way it needs to *install securely*. Here, we explain how to install your blog the secure way, rather than the more vulnerable, easy, one-click way.

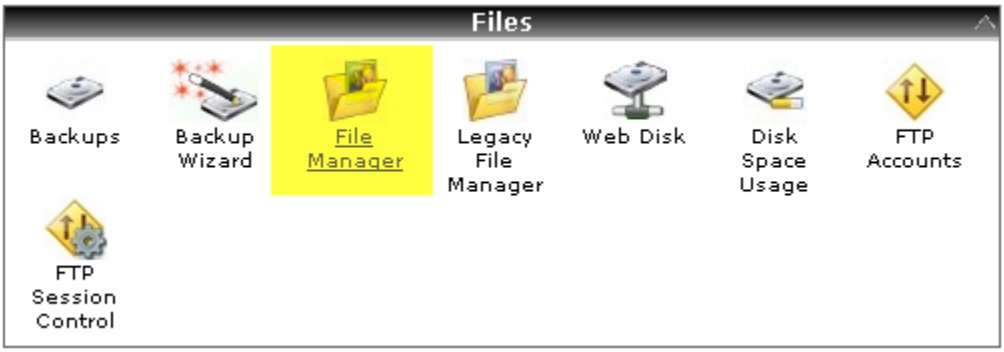
WordPress Software

The latest version of WordPress is available at http://wordpress.org/download/ or http://wordpress.org/latest.zip.

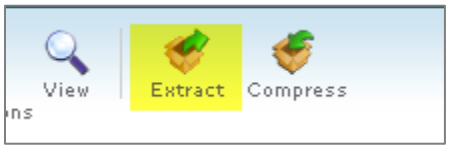
Hop on over to WordPress.org and download the latest version. Extract the archive to a folder on your own hard drive and then either upload it to a folder on your site via FTP software or via your host's site manager, such as cPanel.

cPanel is a Unix based web hosting control panel that provides a graphical interface and automation tools designed to simplify the process of hosting a web site.

With cPanel, you can open File Manager as illustrated below, and run WordPress's archive without having to extract on your own hard drive. This is preferable as you will have only transferred one file to your host (the Wordpress .zip file) and not the hundreds of files after extracting on your own machine first. This will save you HEAPS of time in waiting for the transfer to complete.



After clicking the File Manager icon and uploading the file, extract it into the desired subfolder of your web server with the **Extract** command.



Next, visit your WordPress blog URL. If you extracted the blog's files to your root folder, your blog URL will be something like www.website-name.com. If you extracted the blog's files to a subfolder of your root folder, your blog URL will be something like www.website-name.com/folder-name. The resulting page you'll see will ask you for database connection details.

Remember, examples of good, strong, hard to guess passwords are U#$oi9fd4, 1f^&%84fd, FD5@4)6w_e1, etc.

Give your database a name, a username, and indicate the name of the database's host (typically localhost). You will also have to create a database on your wen host that uses the same details.

When generating a password, use the same strategy outlined above in the section entitled, "Password Management."

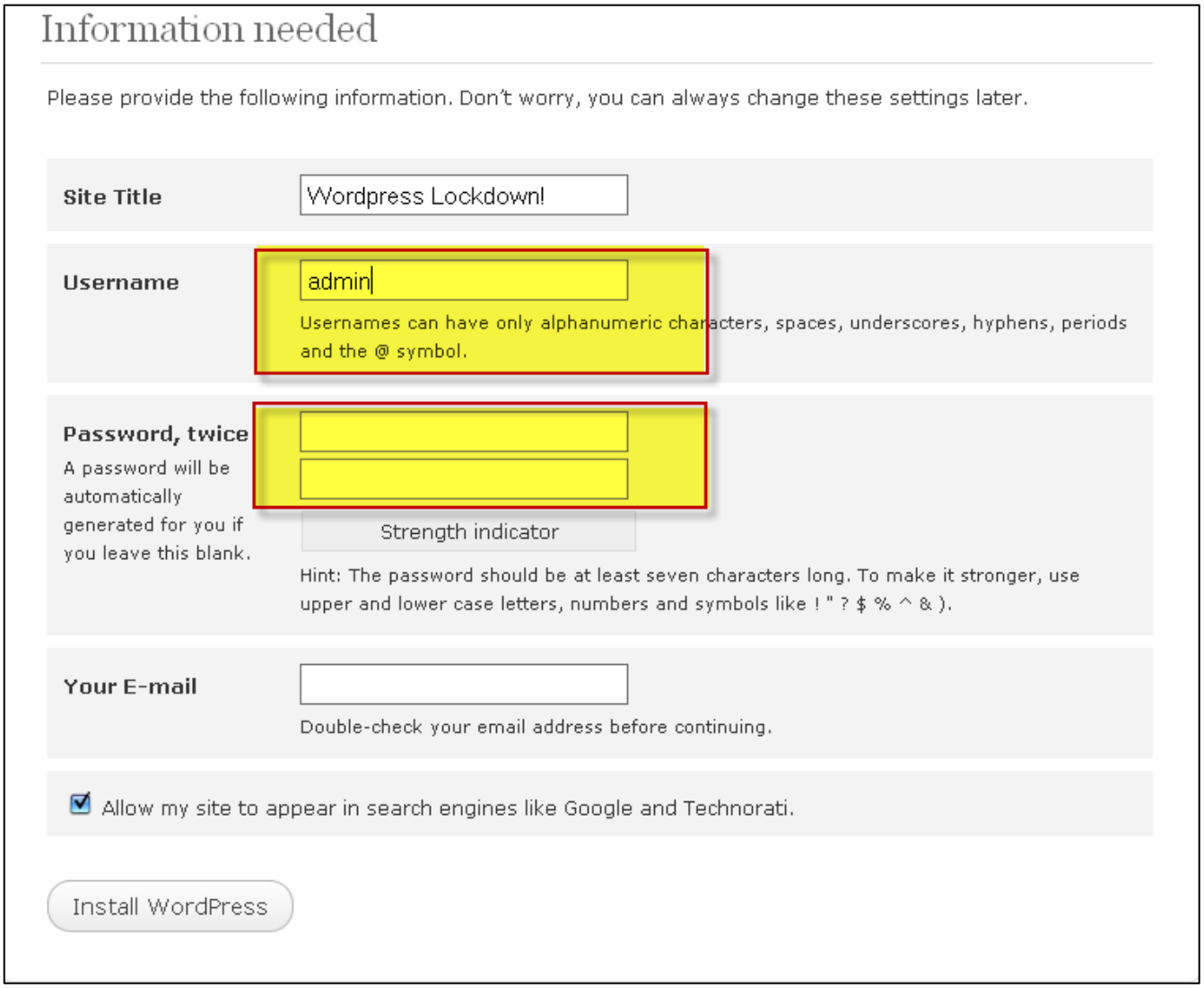
Change the Default Table Prefix

Many hackers assume that you will use **wp_** as your table prefix. So change your blog's table prefix to fool them and add another level of security. It's not hard to change the table prefix during installation. Just replace the **wp_** with a random word and don't forget the "_" at end – it's for your convenience.

Change the table Prefix to prevent easy hacking.

Default Admin

Since newer WordPress versions let you use a username other than the default "admin," go ahead and enter a unique username. The default "admin" is a good hacker target because it's so common and because a lot people simply won't change it.



Already have a blog installed with "Admin" as the head user? No sweat. If you want to change the admin of an already installed WordPress blog, create a new user with administrative privileges and delete the old one. Easy!

Securing the "wpconfig.php" File

Via .htaccess

The wpconfig.php file saves your WordPress configuration, database password, and secret keys. **It should, therefore, never be accessible via a web browser.** Prevent accidental exposure by editing your .htaccess file.

The .htaccess file (hypertext access file) is the default name of a directory-level configuration file that allows and/or restricts access to specified files and directories.

You'll find your .htaccess in the root directory of your WordPress folder. Open it with a plain text editor, and add these lines ***at the very end of the file***.

```
<files wp-config.php>

order allow, deny
```

Those two simple lines inserted into any .htaccess file will block access to a blog's wp-config.php file.

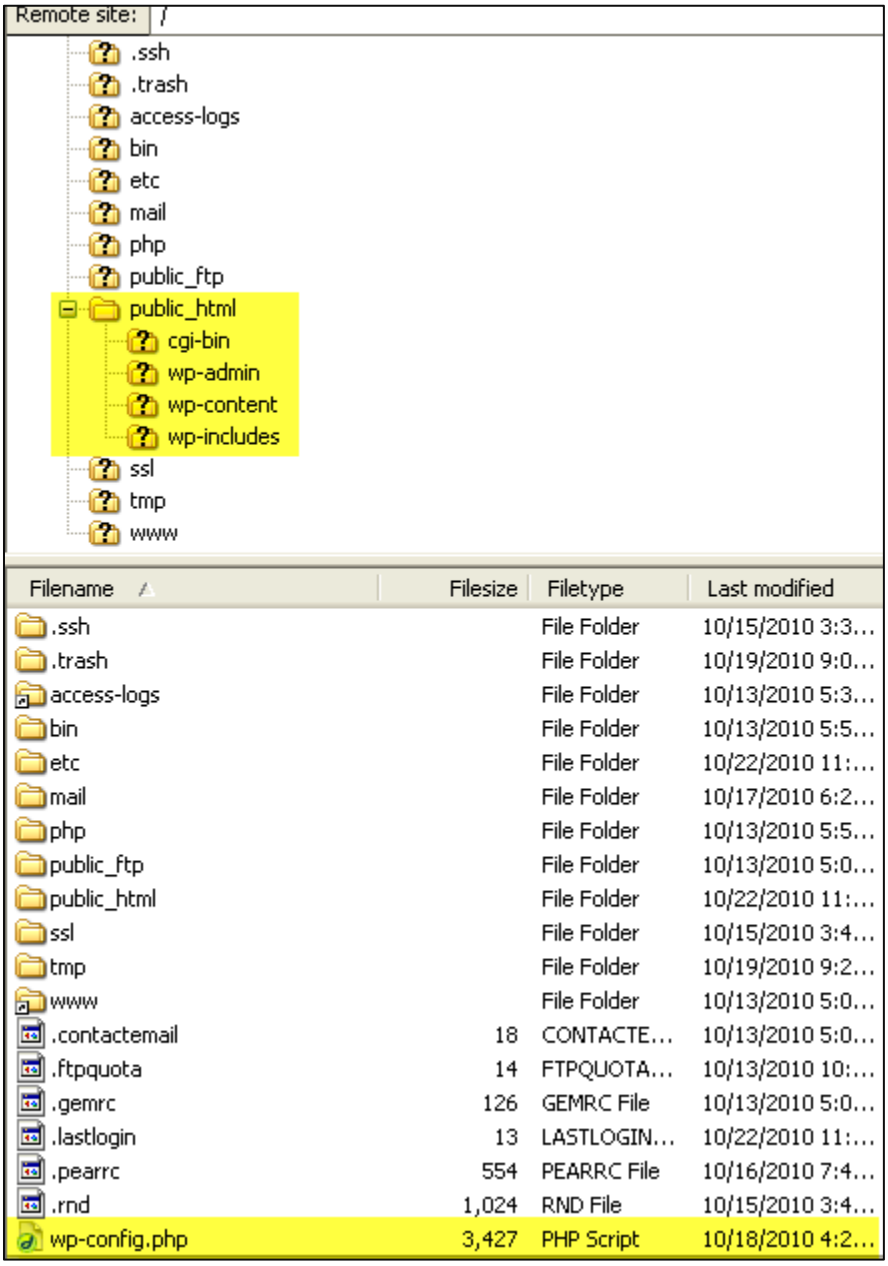
Via Directory Movement

Another way to secure your blog is to move its wp-config.php file ***out*** of your root folder. Browsers cannot access files above the web server root, however, server side scripts can. (WordPress is a server side script.) Therefore, moving the wp-config.php out of the root folder can greatly improve security.

A server side script is a program that is processed on the server, before the information ever reaches the viewer's computer.

This strategy isn't necessary if you follow the strategy above (although it can provide increased protection), however you should note that this strategy won't work if you installed your blog into a subdirectory. You can only move the file ***one level up*** from where WordPress is installed.

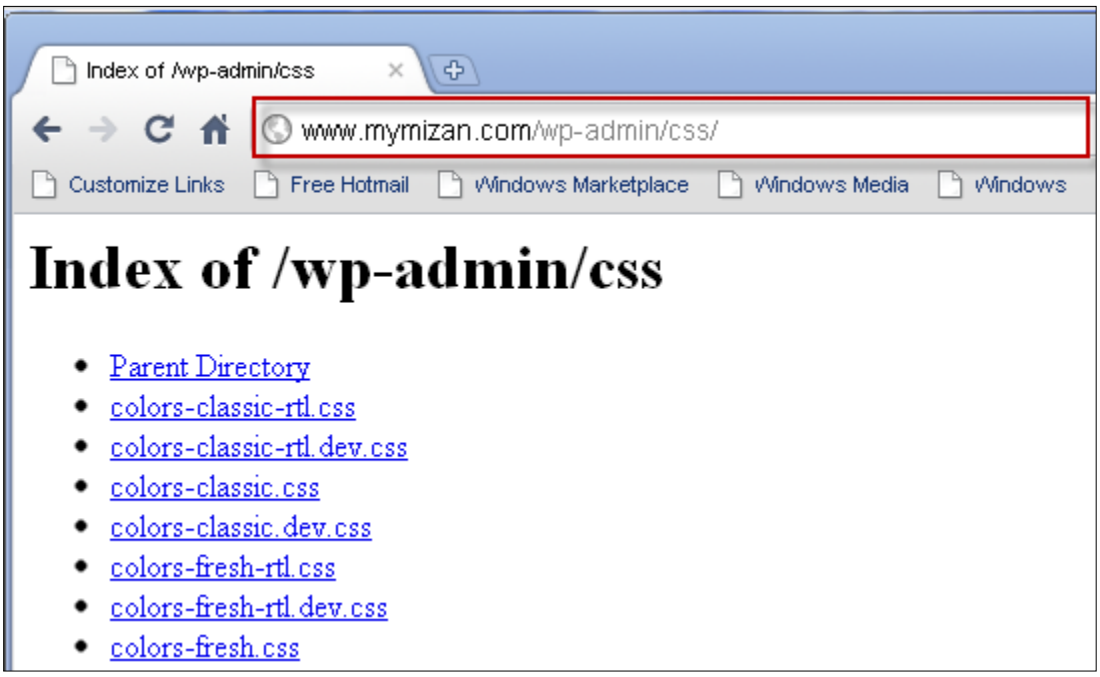
So if you installed WordPress in the root directory of your site, generating a www.site-name.com type of blog URL, moving the file one lever up will take it out of web root, which is impossible for a browser to access. However, if you installed your blog in a subdirectory, generating a www.site-name.com/blog/blog-contents type of blog URL, one level up from that installation is the ***blog*** directory, which is ***not*** out of the web server root (and consequently, vulnerable to attack). Blogs installed into a subdirectory can be secured with the .htaccess strategy described above.



My WordPress files are in public_html but, wp-config.php is at / (home directory FTP root) which is one directory level up and outside of the web root.

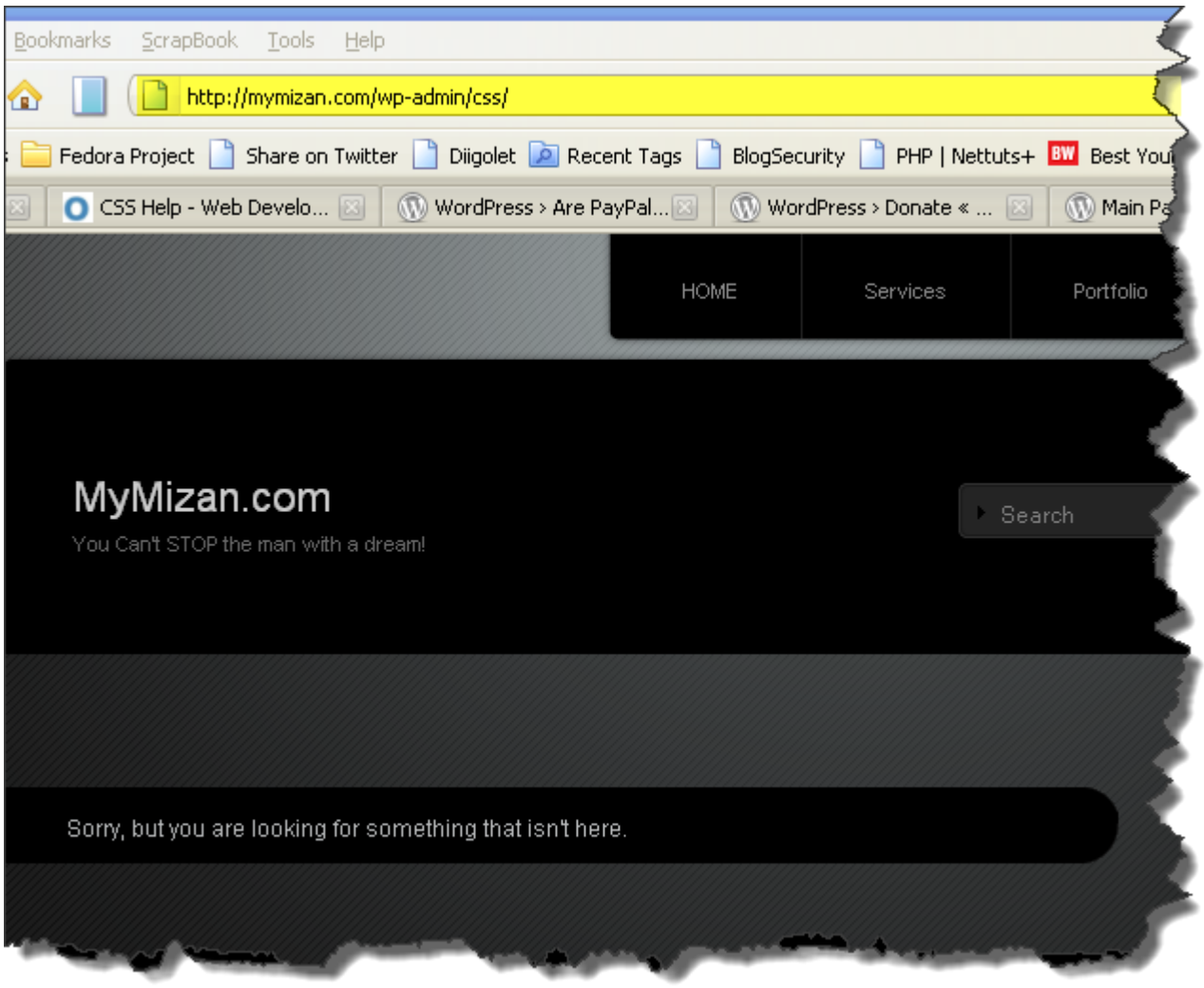
Prevent Directory Browsing

In most server environments, directory browsing is enabled by default, which creates a huge security risk. In the example below, files in **wp-admin/css/** are browsable. Hackers can access the files in the CSS folder by just typing in the folder's URL "www.mymizan.com/wp-admin/css/."



Using .htaccess, we can prevent directory browsing by adding the following line to the .htaccess file.

```
Options -Indexes
```



Now, when someone wants to browse the www.mymizan.com/wp-admin/css/ folder, they're greeted with a 404 error page. Nice.

File Permissions

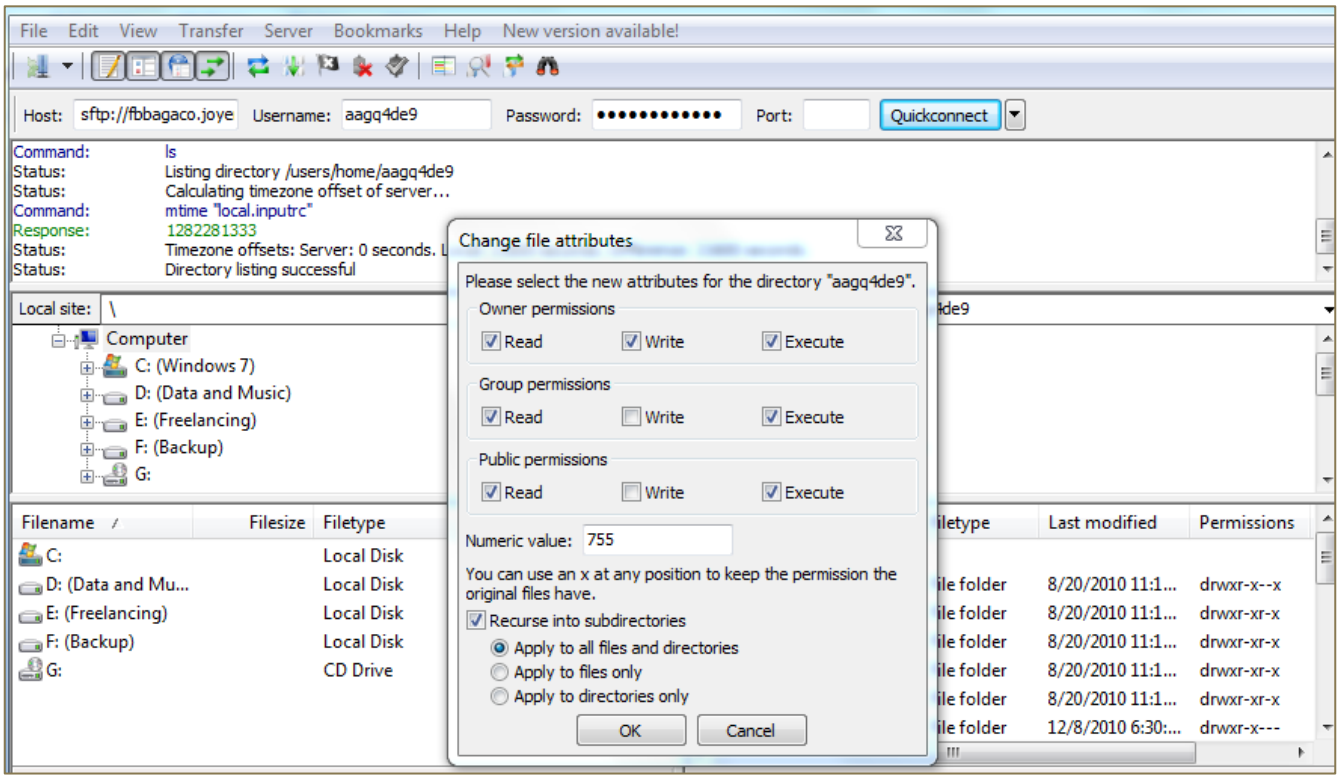
Like most blogs, yours may be hosted on a server that uses shared hosting. With shared hosting, a ***lot*** of bloggers host their blogs on the same physical machine, making it all the more important to ensure others cannot read your files. Adjusting file permissions is but another way to keep your files from prying eyes.

With your FTP or shell software, do the following:

***chmod** 750 wp-config.php*

Chmod is a UNIX command that changes the mode of a file. There are three modes a file can have turned on or off: read ability, write ability, and execute ability.

All other files should be chmoded **644** and all directories should be chmoded **755**, as illustrated below.



Updating WordPress

We couldn't over-stress the importance of updating WordPress if we wanted to. Besides adding new features, updates fix security bugs too. Never postpone an update. Install it right away when you see a prompt to do so on your blog's dashboard. Updates help prevent hacking! So...

Requirements

Before updating, make sure your web host provides WordPress's minimum requirements[2]. These requirements are nothing spectacular, and chances are, if you've successfully installed WordPress, your host already provides what's needed.

Backups

Back up your blog before proceeding. I describe how to securely back up your blog later on in the book, so if you like, you can jump to it now, or read through this section first before proceeding. In any case, never update your blog without backing it up beforehand. You'll find out why soon.

Disabling Plug-ins

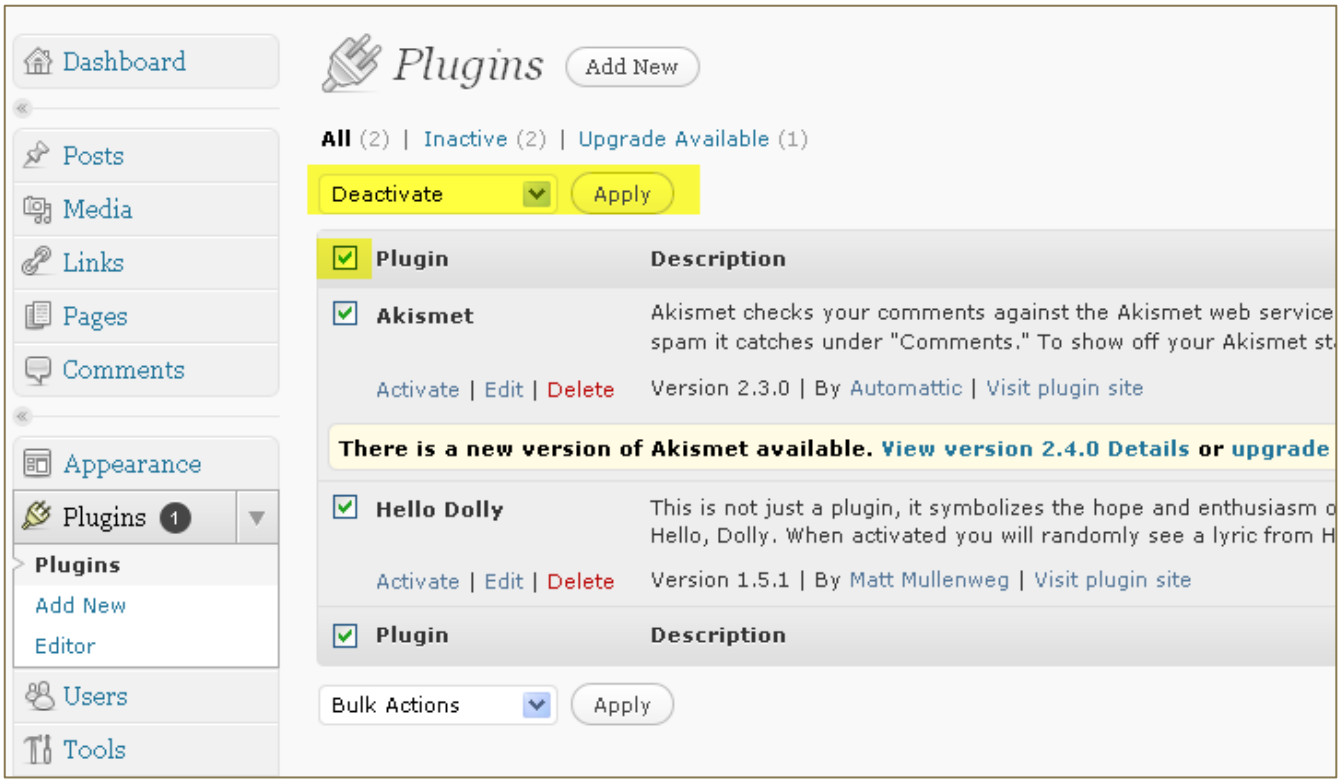
After updating to a major version, some plug-ins may not work. Therefore, it is wise to not only back up your blog and all of its accompanying plug-ins (in case you have to restore it to its original state), it's important to disable plug-ins before you upgrade as well. If you don't, you may encounter compatibility issues and errors after updating your copy of WordPress.

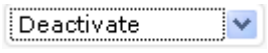
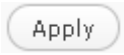
Once your blog is updated, you can reactivate your plugins later. Only in cases of severe incompatibility and resulting errors will you need to disable a plugin (and keep it disabled) until its developer releases a version that's compatible with the latest WordPress upgrade.

Here's how to safely disable all your plugins at once:

---

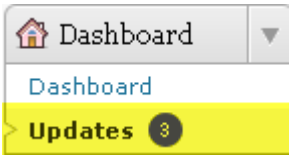
[2] http://wordpress.org/about/requirements/



- Login to WordPress.
- Click the **Plugins** tab.
- Click the top arrow beside **Plugin** to select all.
- Select  from drop down menu and click  .

Automatic Update

Automatically updating your blog is the easiest way to keep your WordPress version current. Recent versions support this feature, but if you have an earlier version, I show you how to update your blog manually next. For those with a recent version of WordPress:

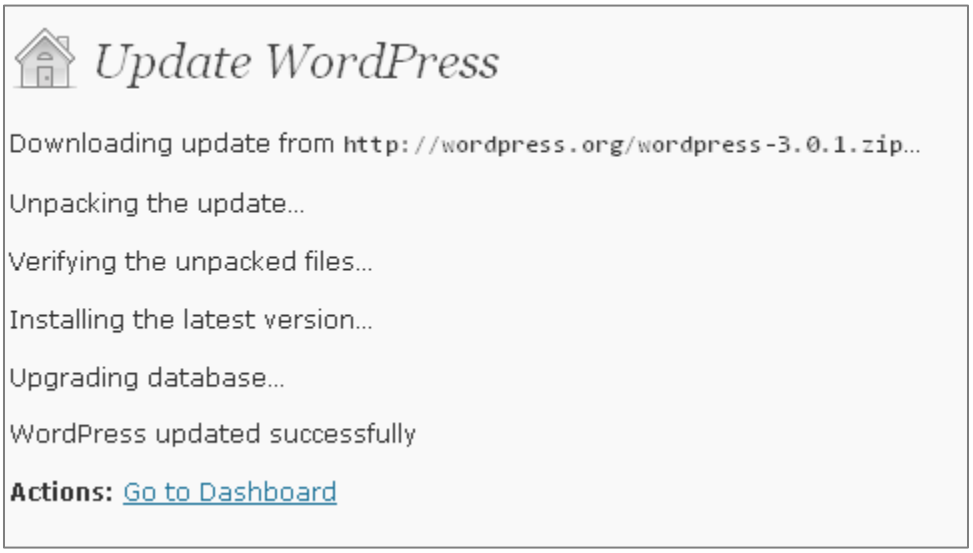
- Click the  option under the Dashboard.

- Click the  button to update.

- Wait until the update is finished. Do ***not*** navigate away from the page during the

update!

You'll see the following:



Upon a successful update, WordPress will display an **Updated successfully** message. If, however, your automatic update failed, delete the **.maintainance** file in the blog's root folder to bring WordPress out of maintenance mode, and then proceed with the manual update instructions below.

Manual Update

When automatic updating is not an option, you can always manually update your blog this way:

1. Download the latest WordPress files from http://WordPress.org/download.
2. Extract them to a folder on your own hard drive.
3. Login to your web host with FTP.
4. Delete the old **wp-admin** and **wp-includes** folders.
5. Do ***not*** delete the **wp-contents** folder! This folder stores your blog's contents (themes, plugins, etc.).

   If you customized your themes, do ***not*** overwrite their files, otherwise you'll lose all your edits.
6. Now upload and overwrite the extracted files to your web host. You should overwrite every file in your old WordPress installation with the locally downloaded copy.

7. Be careful while overwriting what's inside the **wp-contents** folder. Do not overwrite the directory; instead overwrite every file in the directory separately. Remember that if you have made any customizations to your theme you may have to skip updating some files or re-do the customizations to the new files once overwritten.
8. Visit the **wp-admin** page of your blog and log in.

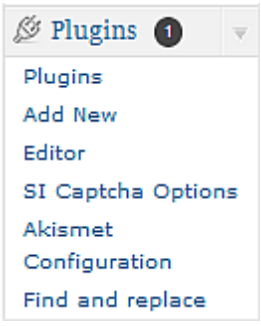
WordPress might display database upgrade confirmation, which will happen automatically when required. Should you see this, follow the link WordPress displays and then follow the instructions to make your database compatible with your new upgrade.

Enabling Plugins

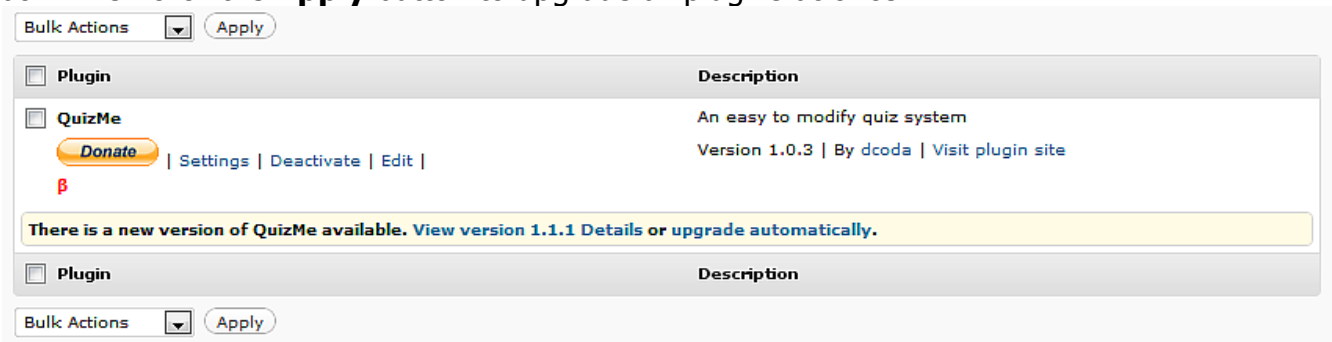
After you've upgraded your blog, you can re-enable the plugins you disabled earlier. A quick way to enable installed plugins is by bulk. However, you might want to enable one plugin at a time just to make sure each one is compatible with your updated blog. A lot of problems with WordPress are often the cause of an incompatible plugin! All it takes is one incorrect line of code to render a blog useless! Finding the culprit causing mayhem can be next to impossible if you enable all of them at one time.

Updating Plug-ins

One of the great things about the WordPress community is that its plugin developers are quick to make their add-ons compatible with each new release of WordPress. A few days after upgrading your entire blog, you might find a few numbers of plug-in updates available on the left panel of the dashboard.



When you see available plugin updates, click the tab to access the list of available plugins. Click the checkbox next to **Plugin** and select **Upgrade** from the **Bulk Actions** drop down box. Then click the **Apply** button to upgrade all plugins at once.

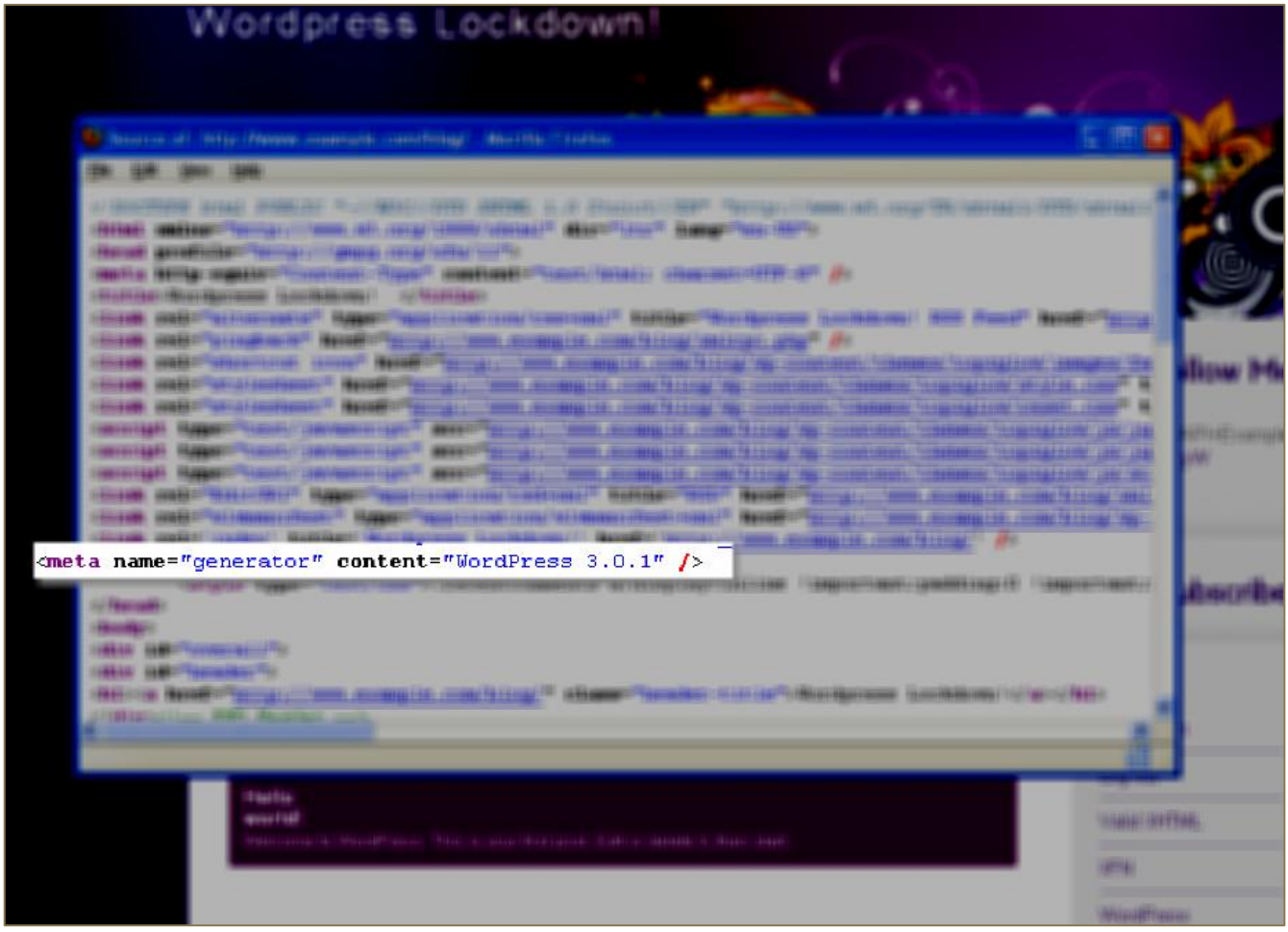


Just like with *enabling* plugins, you might want to *upgrade* one plugin at a time just to make sure each upgrade is compatible with your blog.

Hiding Private Information

Hide WordPress Version

If you read the source code of any WordPress blog, you'll find a meta tag that displays WordPress's version number installed. Delete that tag for security reasons. A hacker will have a harder time wreaking havoc if s/he doesn't know which one of his tools works with your version of WordPress.



Older versions of WordPress are full of bugs – and these are bugs hackers use to their advantage. It won't stop experienced hackers, but it will discourage hackers new to the hacking game along with script-kiddies.

To remove your blog's version number, add the following code to the **function.php** section of your theme file. WordPress won't show any version number from that point on.

```
remove_action('wp_head', 'wp_generator');
```

Hide Login Information

When a user fails to log in, WordPress shows what went wrong with the log in process by displaying what part of the log-in was incorrect. Knowing what information is missing not only reveals what information isn't missing (hence, which data they got correct), it additionally helps hackers discover what's needed for a successful break-in.

Let's say a person logs in with a correct username, for example, but the wrong password. WordPress will inform that person that the password was incorrect. If this person was a hacker, he would be delighted to know that he got half the puzzle solved, and that all he needed to successfully log in was the correct password.

If you hide this type of log in information, hackers will have no idea what they need to successfully log in and will need to attempt a full brute-force of your log-in credentials. If you have been taking my advice and are using a strong password that is not used anywhere else, this will become a formidable task for them and they will move on.

Shown when an entered password is wrong. This tells a hacker that he entered a correct username, but wrong password. A persistent hacker will keep trying to log in with different passwords.

This information is displayed when an entered user name is incorrect. Here, a hacker will know that the user name is not correct and continue to try more.

Don't let hackers know if their efforts are right, half right, or not right at all by adding the following code to your themes **function.php** file.

```
add_filter('login_errors',create_function('$a', "return null;"));
```

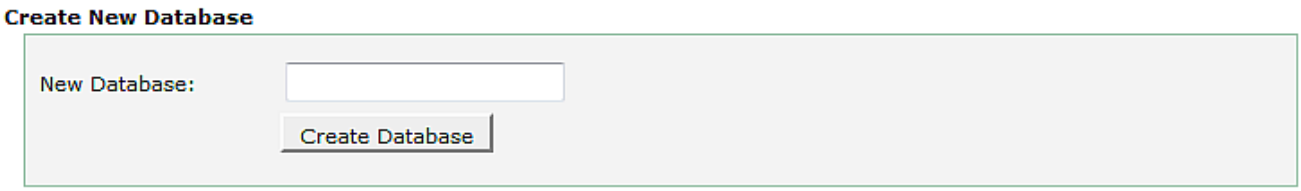
Now nothing is shown in the box - leaving the hacker confused.

Database Security

Secure the MySQL Root Account

A MySQL Root Account is the **first** user account created. It's created during a database's installation.

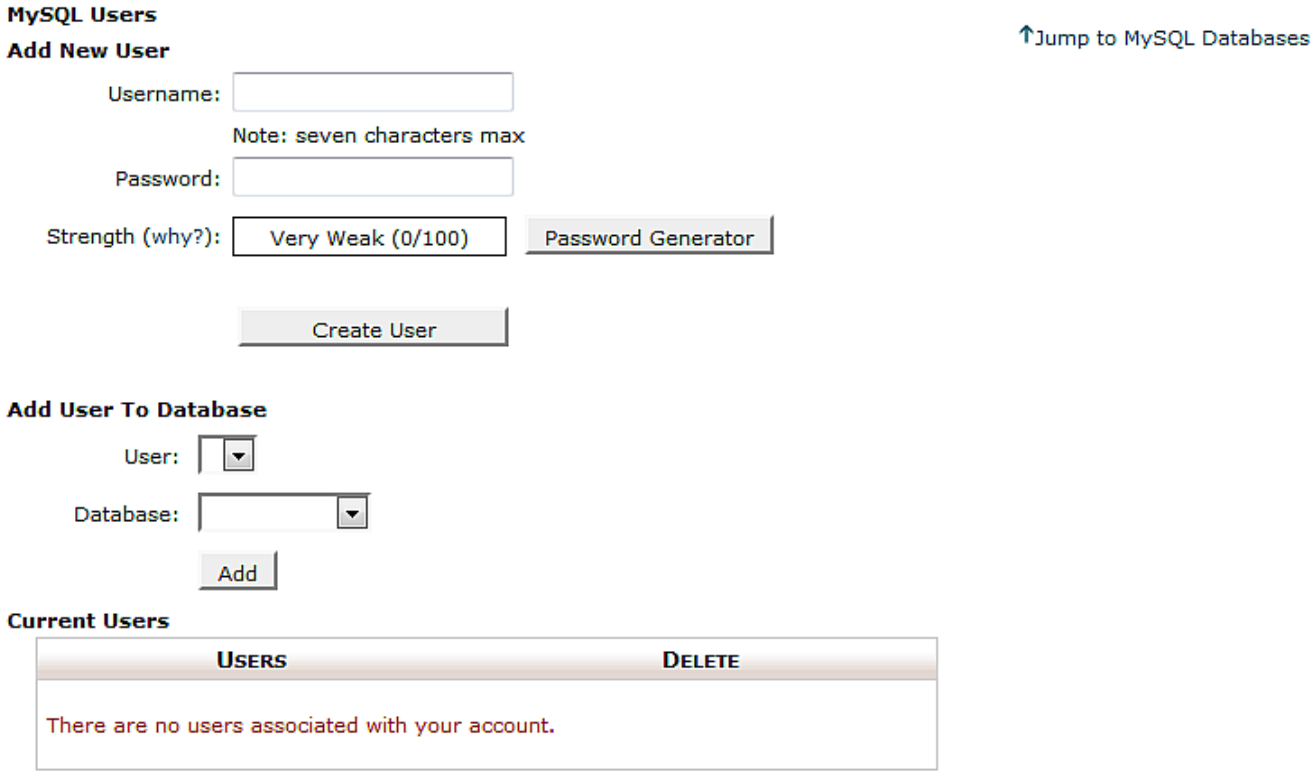
Using MySQL's **root** account to host your database is convenient, but should someone hack that account, that person can easily access all other databases associated with that account since (1) the root account is a superuser account – it has all privileges and can do anything, and (2) the root account doesn't require a password. Once a hacker connects to a MySQL root account, he's granted all privileges to wreak havoc.



You must, therefore, use the root account to create new databases only. No account, except the root account, should *ever* have the ability to create new databases. You mustn't use the same password for every subsequent, new database account either. Always, always, always use different passwords for each. That will reduce the risk of a security breach.

Separate Users

Just as you'll want to use a different password for each database, you'll want to use a different username as well. That way, if one account is compromised, other databases on the same system will remain secure as long as they're accessed via different username and password combinations.



Restrict Permissions

Be sure to give your WordPress database accounts only the necessary permissions they need to operate. **Never** give a database account permission to delete anything. In most cases (if not all), the only permissions databases outside of the root account need, are basic **usage permissions:**

☐ ALL PRIVILEGES	
☑ SELECT	☐ CREATE
☐ INSERT	☐ ALTER
☐ UPDATE	☐ DROP
☑ DELETE	☐ LOCK TABLES
☑ INDEX	☑ REFERENCES
☐ CREATE TEMPORARY TABLES	

Additional Security Tools

Secret Keys

A secret key is a password that contains extraneous elements, making it harder to replicate via password-type hacking tools. Consider for example, the password, "test." That password is simple and easy to guess. A password like "f464-jy68%64!8_6ui@4o}86!uj4$5bf4[hfs," however, is almost impossible to guess. It would take years to guess a password like that, but that's what secret keys do. They make a blog"s sessions must more secure.

Defining unique secret keys in your blog's **wp-config.php** file will make your blog more safe. And you can define them even after you've already installed a blog. The definition won't cause any problems to a running site, but you will, however, have to login again. (No biggie.)

If you've installed your blog with the WordPress installer, your secret key may already be defined. In this case, you don't need to do anything else here.

First, generate your secret phrase from WordPress.org at *https://api.WordPress.org/secret-key/1.1/salt/*.

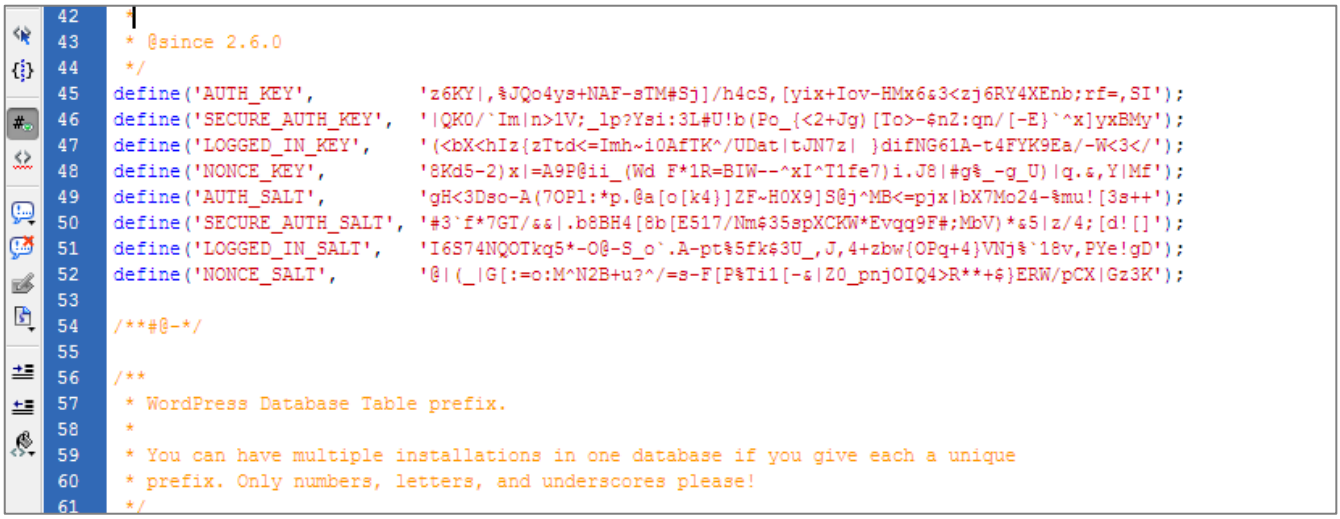
Second, make a backup copy of your blog's **wp-config.php** file.

Third, insert your secret key phrase into the config file as illustrated below, but be extremely careful. If you make a mistake, your blog won't display!

```
43   * @since 2.6.0
44   */
45  define('AUTH_KEY',         'put your unique phrase here');
46  define('SECURE_AUTH_KEY',  'put your unique phrase here');
47  define('LOGGED_IN_KEY',    'put your unique phrase here');
48  define('NONCE_KEY',        'put your unique phrase here');
49  define('AUTH_SALT',        'put your unique phrase here');
50  define('SECURE_AUTH_SALT', 'put your unique phrase here');
51  define('LOGGED_IN_SALT',   'put your unique phrase here');
52  define('NONCE_SALT',       'put your unique phrase here');
53
54  /**#@-*/
55
56  /**
57   * WordPress Database Table prefix.
```

Replace each instance of "put your unique phrase here" with each secret key phrase.

After changing the strings, your file should look something like the illustration below.



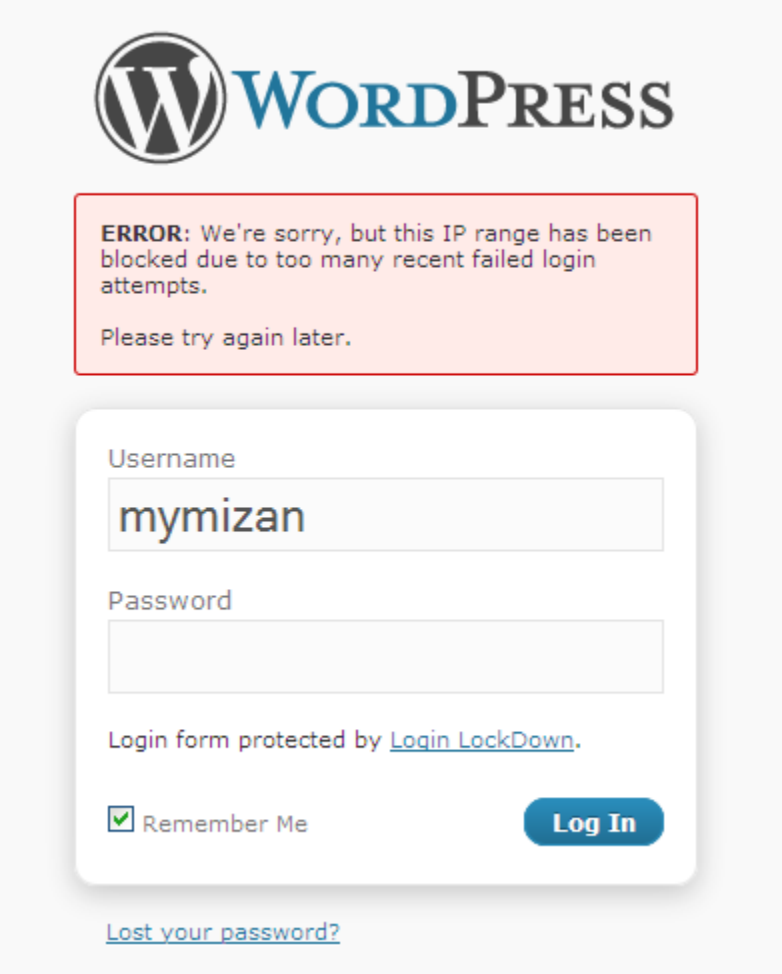
```
 *
 * @since 2.6.0
 */
define('AUTH_KEY',         'z6KY|,%JQo4ys+NAF-sTM#Sj]/h4cS,[yix+Iov-HMx6&3<zj6RY4XEnb;rf=,SI');
define('SECURE_AUTH_KEY',  '|QK0/`Im|n>1V;_lp?Ysi:3L#U!b(Po_{<2+Jg)[To>-&nZ:qn/[-E}`^x]yxBMy');
define('LOGGED_IN_KEY',    '(<bX<hIz{zItd<=Imh~i0AfTK^/UDat|tJN7z| }difNG61A-t4FYK9Ea/-W<3</');
define('NONCE_KEY',        '8Kd5-2)x|=A9P@ii_(Wd F*1R=BIW--^xI^T1fe7)i.J8|#g%_-g_U)|q.&,Y|Mf');
define('AUTH_SALT',        'gH<3Dso-A(7OPl:*p.@a[o[k4}]ZF~H0X9]S@j^MB<=pjx|bX7Mo24-%mu![3s++');
define('SECURE_AUTH_SALT', '#3`f*7GT/&&|.b8BH4[8b[E517/Nm$35spXCKW*Evqq9F#;MbV)*&5|z/4;[d![]');
define('LOGGED_IN_SALT',   'I6S74NQOTkq5*-O@-S_o`.A-pt%5fk&3U_,J,4+zbw{OPq+4}VNj%`18v,PYe!gD');
define('NONCE_SALT',       '@|(_|G[:=o:M^N2B+u?^/=s-F[P%Ti1[-&|ZO_pnjOIQ4>R**+$}ERW/pCX|Gz3K');

/**#@-*/

/**
 * WordPress Database Table prefix.
 *
 * You can have multiple installations in one database if you give each a unique
 * prefix. Only numbers, letters, and underscores please!
 */
```

You can get unique keys automatically generated for you at the wordpress site here: http://api.wordpress.org/secret-key/1.1/. This will save you tons of time and is highly recommended.

Login LockDown

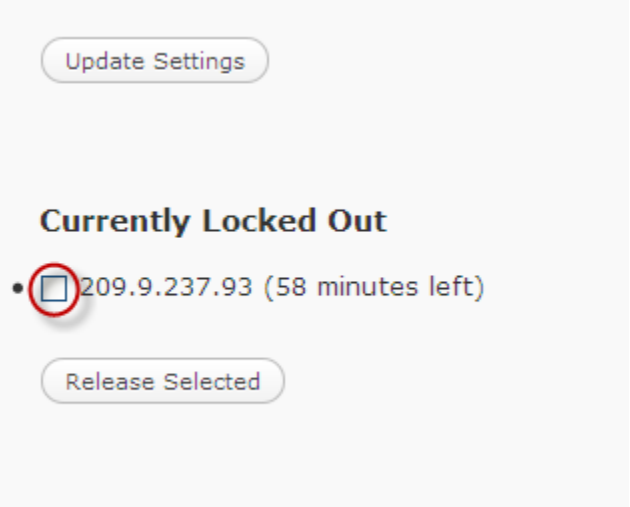
The **Login LockDown** plugin records the IP address and timestamp of every failed login attempt. If more than a certain number of attempts are made within a short period of time, the login function is disabled for all requests from the computer making the failed login attempts. This helps prevent brute force password discovery (aggressive password guessing).

Currently, the plugin defaults to a 1 hour lock down after 3 failed login attempts made within 5 minutes. But this can be changed through the plugin's options panel if desired. I find that the defaults are fine.



If a hacker enters the wrong password five times, he'll be blocked. Login LockDown protects your blog from brute force password attacks.

Should you need to release a lock down, you can do that from the plugin's options panel as well.



Select the desired IP number and then click Release Selected to remove the IP from the ban list.

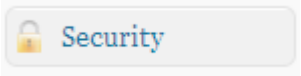
Install the Login LockDown plugin from the WordPress plugins repository, or from http://www.bad-neighborhood.com/login-lockdown.html.

WP Security Scan

Install the WP Security Scan plugin from the WordPress plugins repository, or from http://semperfiwebdesign.com/plugins/wp-security-scan/.

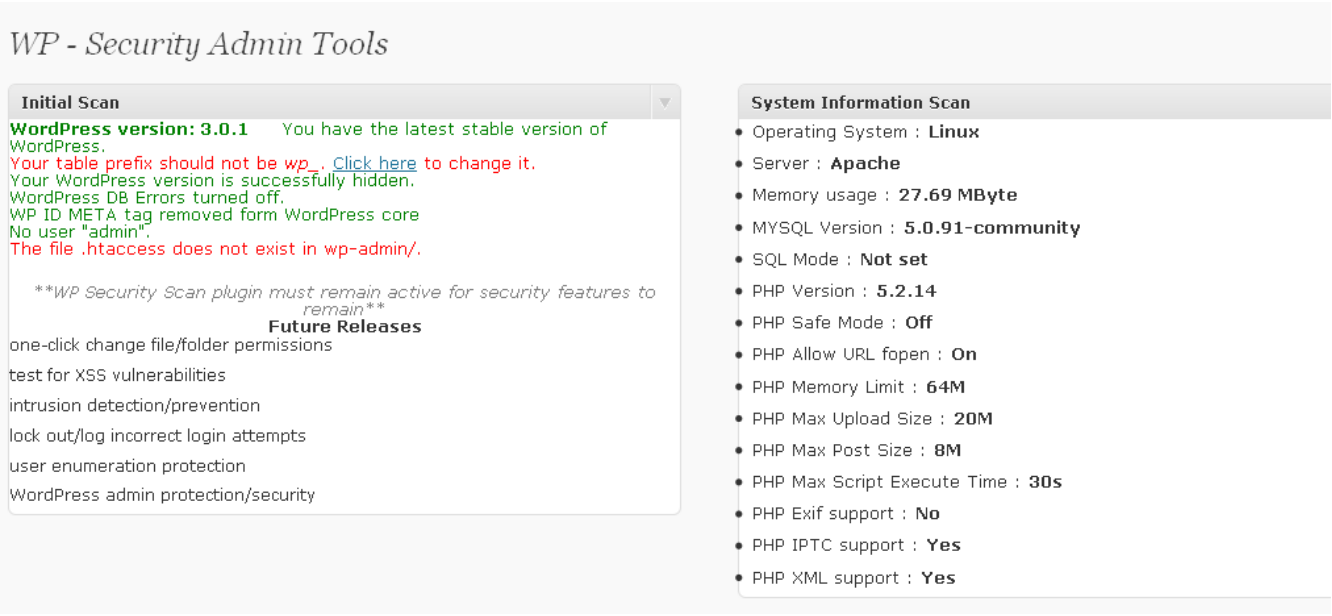
The WP Security Scan plugin automatically checks a blog's current security measures, and suggests improvements.

After installing it, you'll see a new **Security** menu added to the panel.

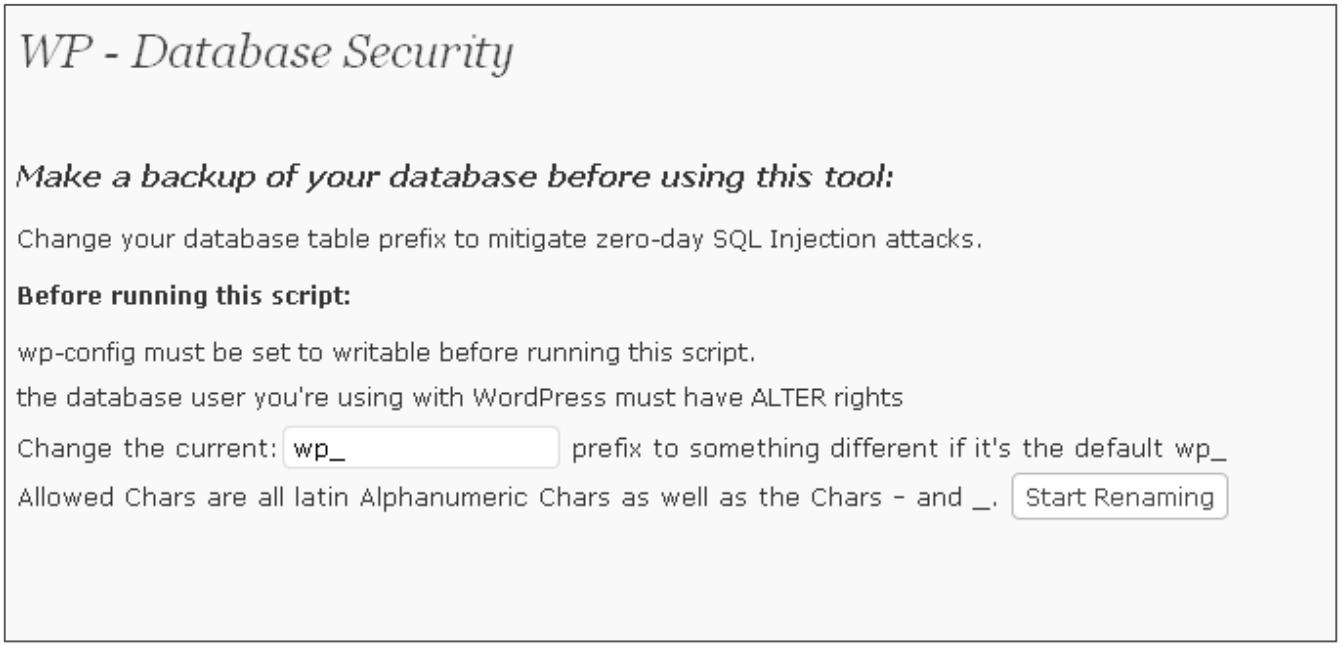
Run the plugin by clicking on the 

button. You'll see the scan's results as illustrated below.

If you followed my instructions above, your blog should pass these scans!

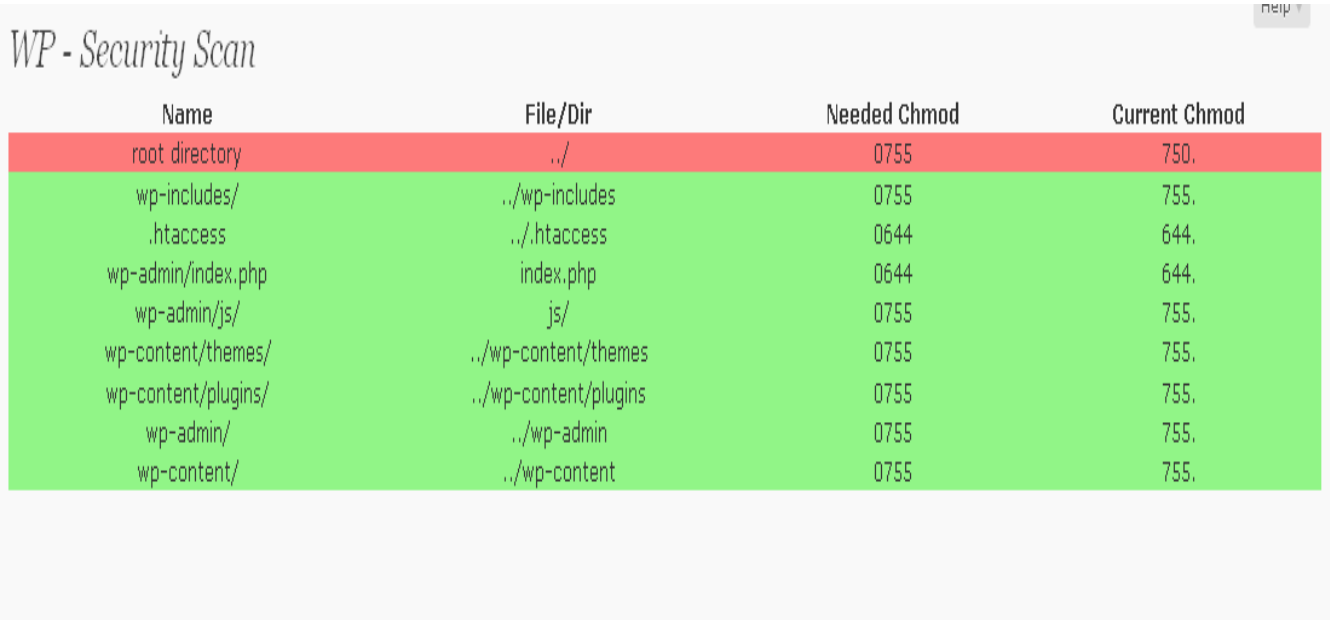


WP Security Scan lets you secure your blog's database table prefix with a different name as a way to thwart hackers. However, you have to be extremely careful when editing this prefix. One mistake will corrupt the entire database, so back it up before proceeding (see page 37 for secure back up instructions).



Before clicking on Start Renaming, check that you have given a different prefix. Then, click it once.

This scanner additionally displays file system permissions. Use this information to verify your permissions are set correctly as per below, via FTP or ssh.



WP - Security Scan

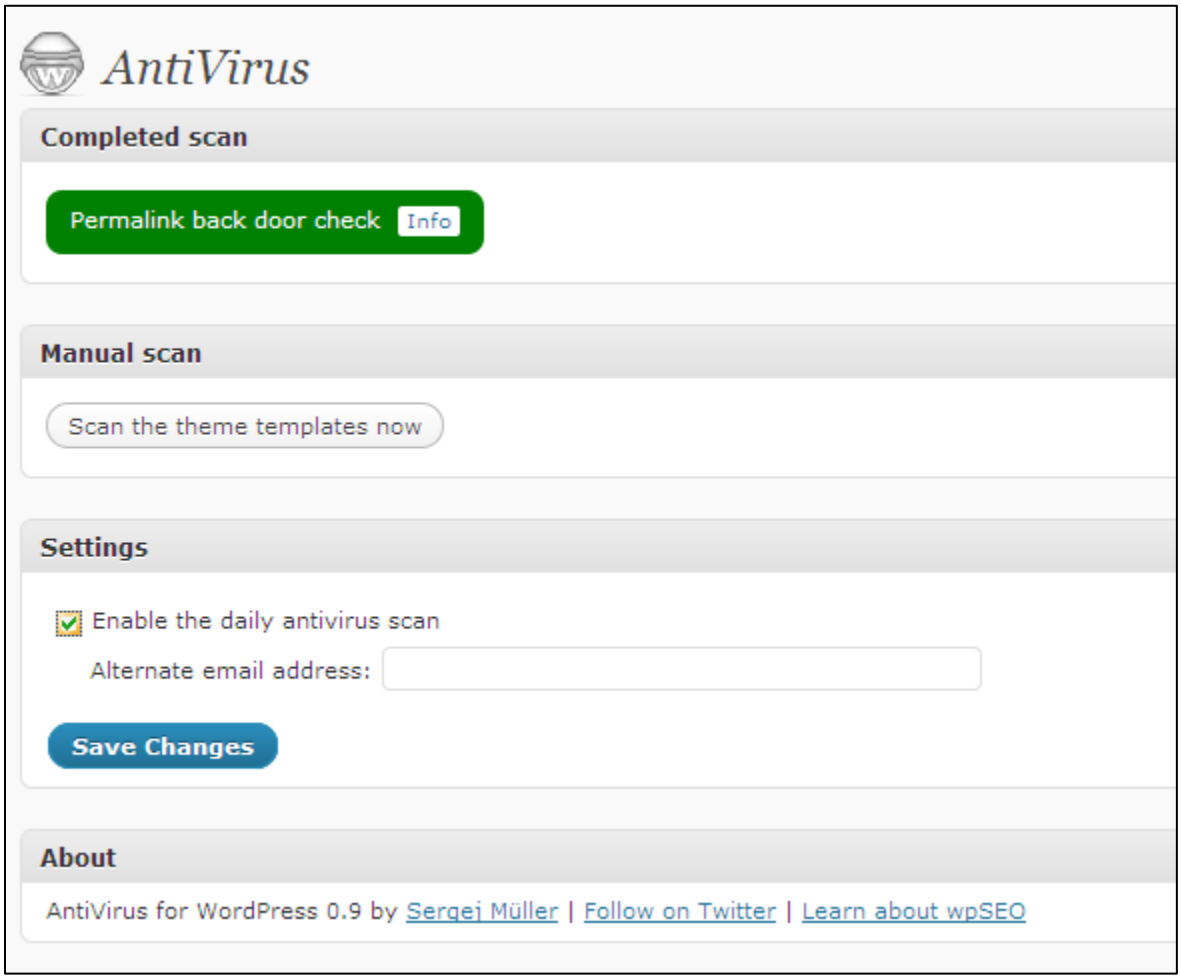
Name	File/Dir	Needed Chmod	Current Chmod
root directory	../	0755	750.
wp-includes/	../wp-includes	0755	755.
.htaccess	../.htaccess	0644	644.
wp-admin/index.php	index.php	0644	644.
wp-admin/js/	js/	0755	755.
wp-content/themes/	../wp-content/themes	0755	755.
wp-content/plugins/	../wp-content/plugins	0755	755.
wp-admin/	../wp-admin	0755	755.
wp-content/	../wp-content	0755	755.

AntiVirus For WordPress

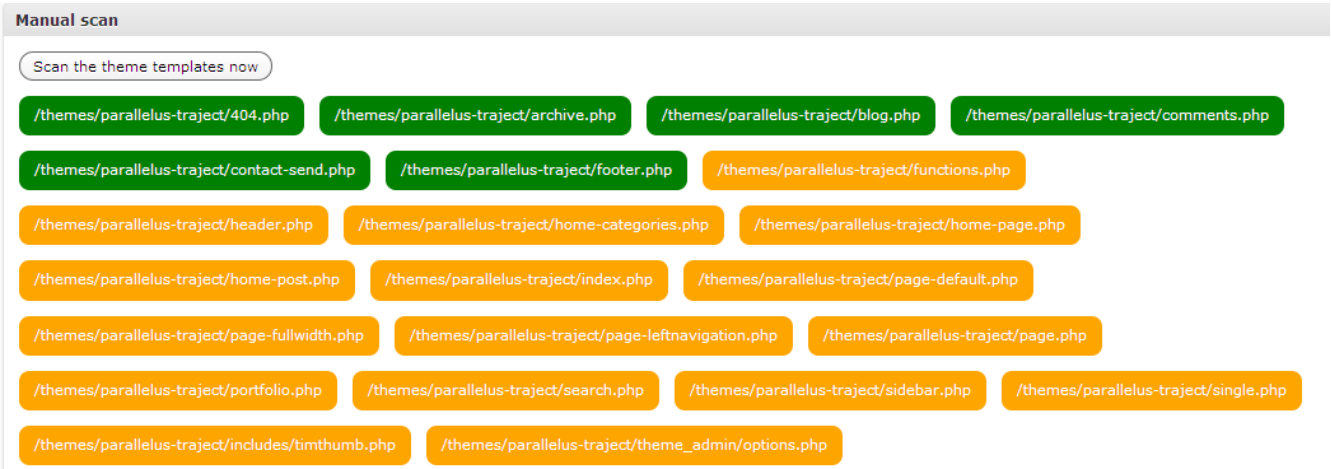
Install the Login LockDown plugin from the WordPress plugins repository, or from http://wpantivirus.com/.

The AntiVirus for WordPress plugin protects your blog against exploits and spam injections with automatic daily scans, or manual scans, and searches for malicious content.

It is easy to install and use and is quite intuitive. I recommend it for all blogs.



Automatic daily scan setup.



Antivirus for WordPress Manual Scanning

reCaptcha

A captcha is a challenge that checks whether a response is computer generated or human entered. It's made in such a way that only a human can decode it, and you've probably run into these things while filling out forms online. Using a captcha with your comment form makes it unreachable to spamming bots because they can't correctly interpret and respond to the captcha challenge.

Install the reCaptcha plugin from the WordPress plugins repository, or from http://wordpress.org/extend/plugins/wp-recaptcha .

The reCaptcha plugin is one of the best captcha plugins available. Usage requires a special key code, however registration isn't necessary. Generate your key code from *https://www.google.com/recaptcha/admin/create* and use it to install reCaptcha. After a successful install, your blog will post a challenge to each commenter prior to accepting comments.

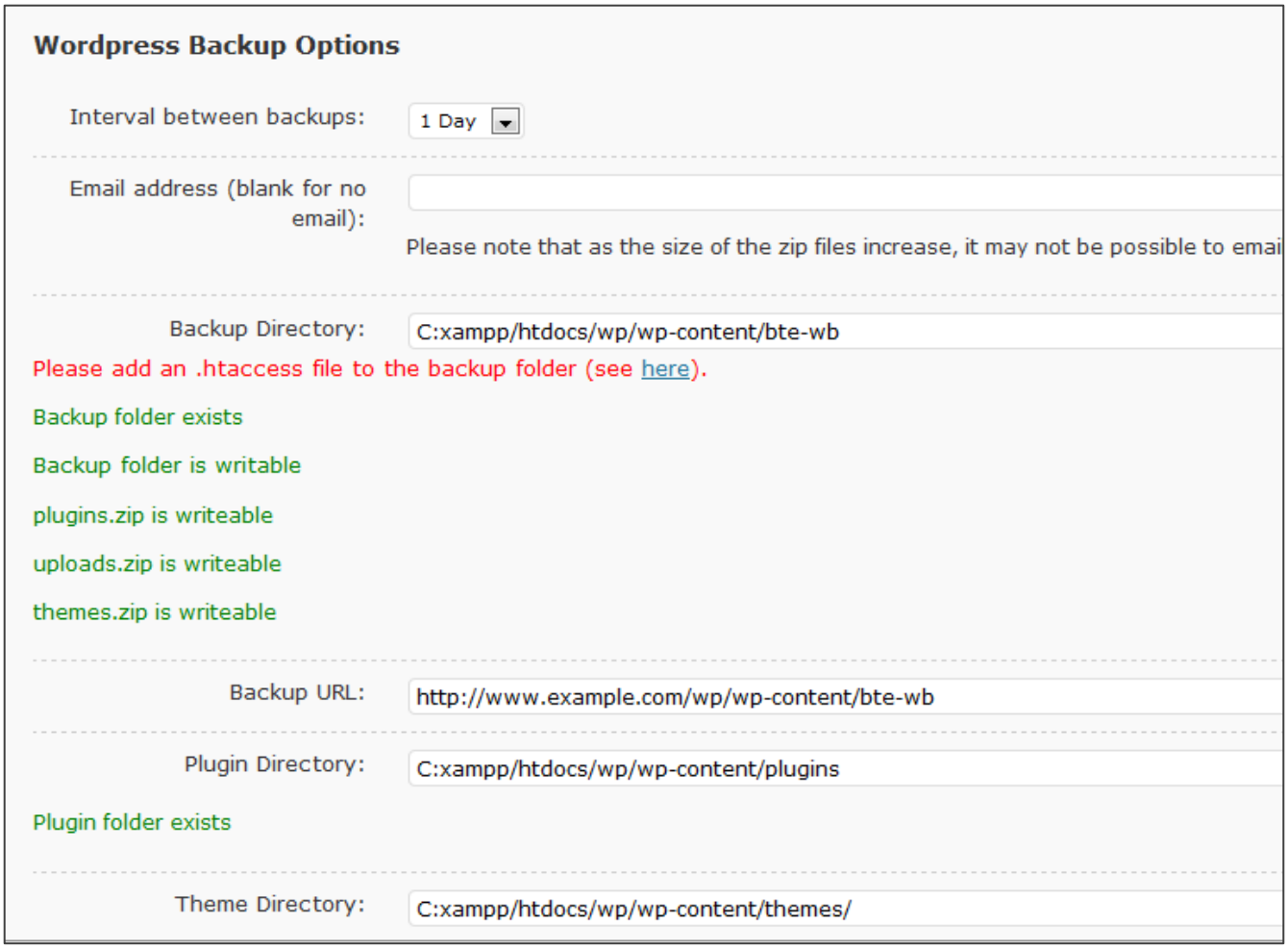
Backing Up Your Blog - A MUST!

Despite knowing regular backups are an important part of the blogging process, a lot of bloggers don't archive their sites because it takes time and dedication. The larger the blog, the more tedious the process. That is, unless you're savvy enough to use a script that automatically backs up your blog at regular intervals. This section introduces you to some of the most popular scripts designed to keep all your configuration files, and most importantly, your security measures up-to-date and in one place.

WordPress Backup

WordPress Backup is an automatic backup plugin that regularly archives your upload, plugin and themes directory. If you prefer, it will email you these files as long as the archived size is within a predefined limit.

The screenshot below illustrates the options you have in the plugin's settings submenu.



Sometimes the WordPress Backup plugin can't detect your theme directory path and plug-in path. That's not a problem. Enter them manually and you're all set to go.

Install the WordPress Backup plugin from the WordPress plugins repository, or from http://www.blogtrafficexchange.com/wordpress-backup/.

It's important to note that the WordPress Backup plugin doesn't back up your blog's database, which stores your settings and posts. You'll need a separate plug-in called WP-DBManager to backup your database. Download and install WP-DBManager from http://lesterchan.net/portfolio/programming/php/.

VaultPress

Vaultpress is a WordPress backup ***service*** offered from the creator of WordPress. Backups made through this service are saved on WordPress.com's infrastructure, but it isn't open to everyone. The service is in beta, and you'll need an invitation from VaultPress to use it at the time of writting this book. To get an invitation, apply at *http://vaultpress.com/signup/* and be prepared to pay for it within a few days. This service costs about $15/month.

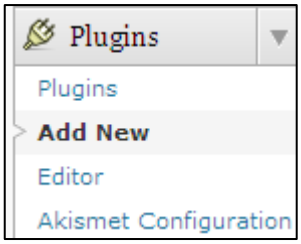
Backupify.com

Backupify is a zero-attendance backup plugin that archives WordPress files on a daily or weekly basis. It performs the whole process automatically and then emails you once the archiving is complete. Note that unlike with the WordPress Backup plugin, this plugin doesn't email you the archives. It simply sends a message when the archiving is complete.

The actual archives (limited to two gigabytes total) are stored at the Backupify server for free. More storage space is available at an additional cost, however two gigabytes is adequate for most bloggers.

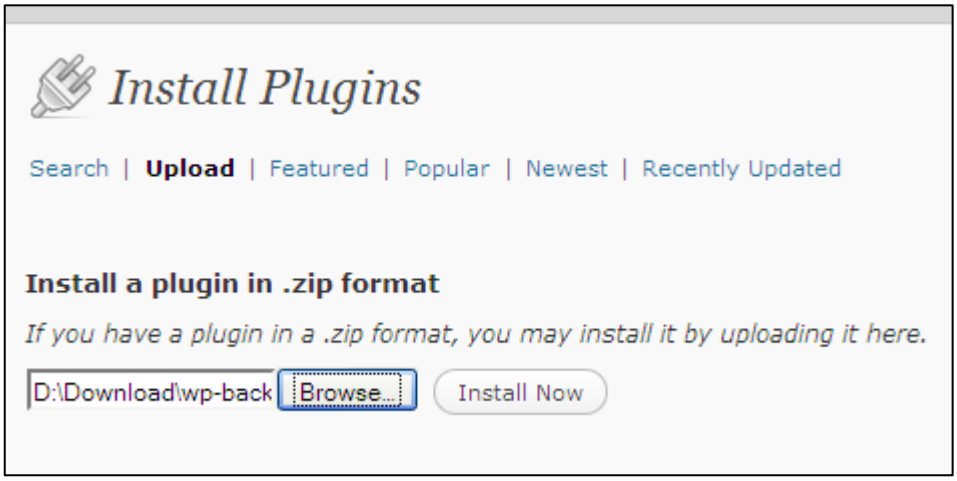
To configure Backupify for WordPress:

1. Create a backupify account at *www.backupify.com*.
2. Download the wp-backupify plugin from *https://secure.backupify.com/wp-backupify-1.0.6.zip*.
3. Login to your blog.
4. Click the plugins subpanel and then click **Add New**.



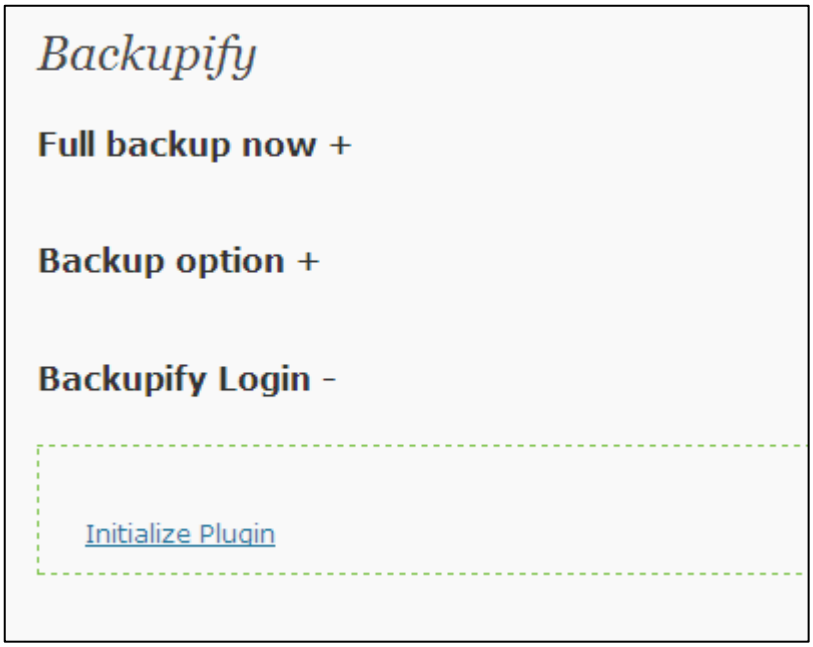
5. Select **Upload** from the **Install Plugins** menu.

6. Select the file to upload and click **Install Now**.

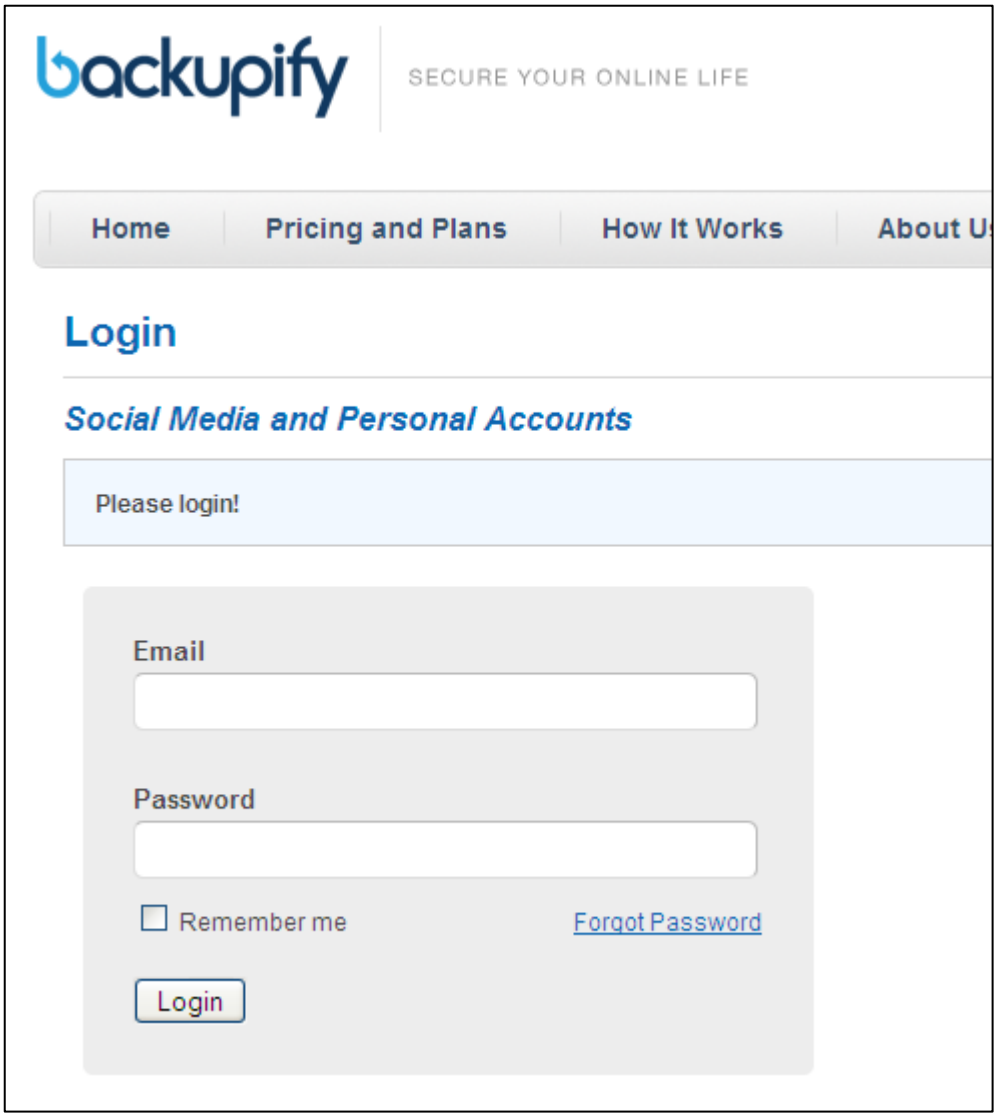


7. Start the plugin by clicking on **Activate**. The Backupify subpanel is added automatically. Click it to access its options.

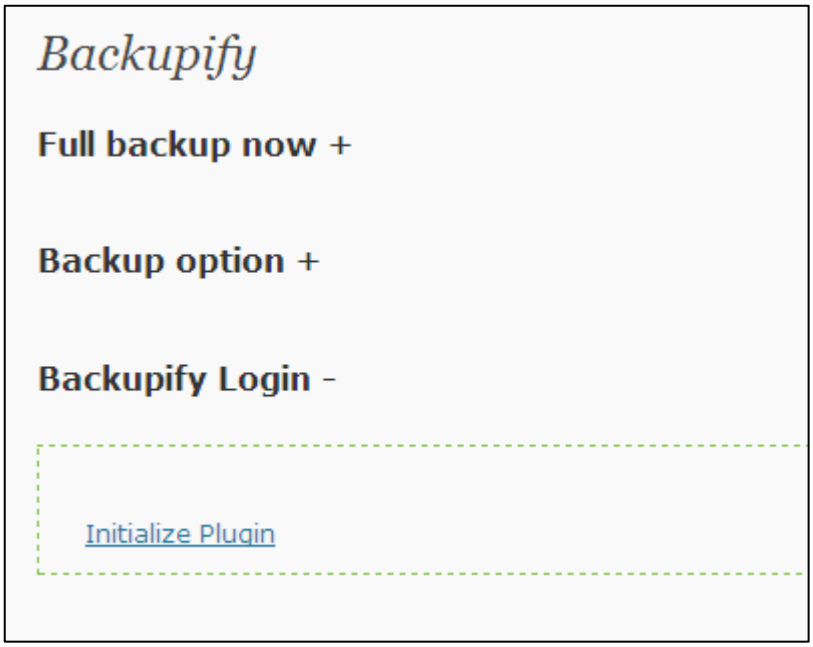
8. Expand the **Backupify Login** menu by clicking on the **Initialize Plugin** link. That will take you to backupify.com.



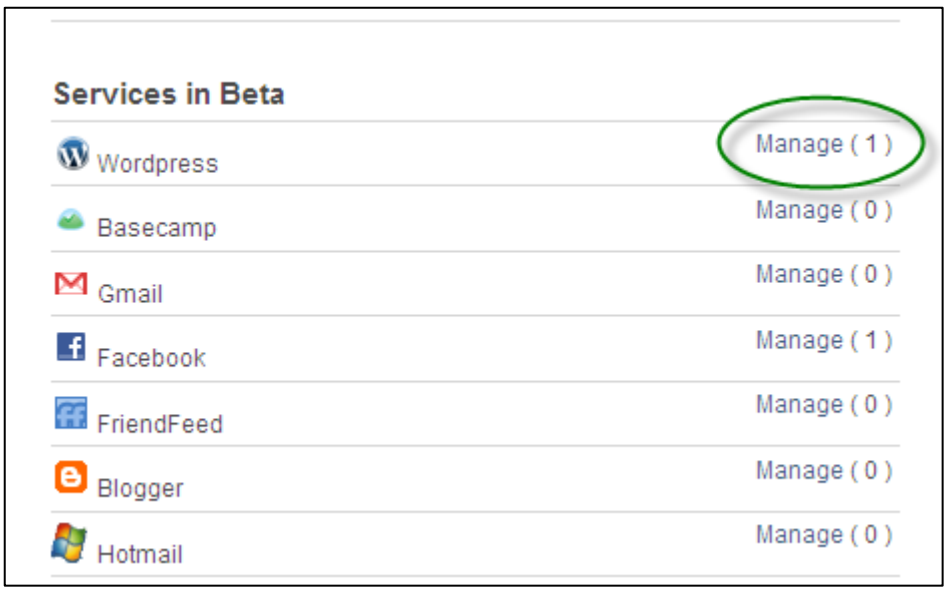
9. Provide your login credentials if you aren't already logged in. Otherwise, your plugin should be activated.



10. Click **Full Backup Now** if you want to start the process immediately.



11. Active services can be accessed from the Backupify site, but free customers can not add more than one. They also have a very low upgrade plan.



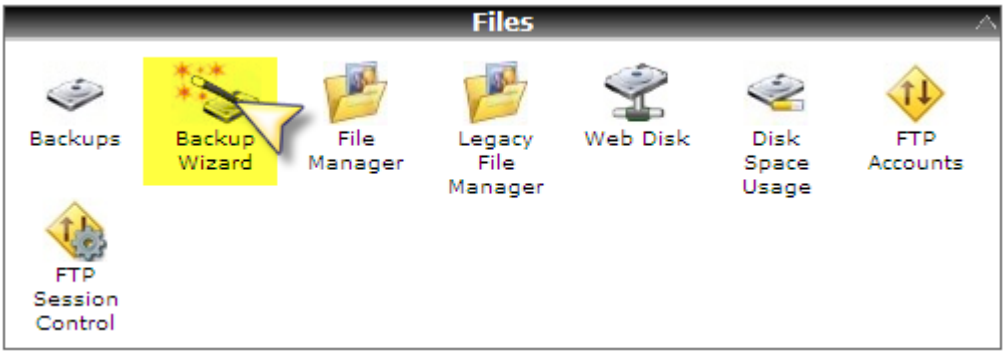
Another good service that's similar to Backupify is BlogVault (http://blogvault.net). BlogVault focuses on easy restoration too.

Manual Backup

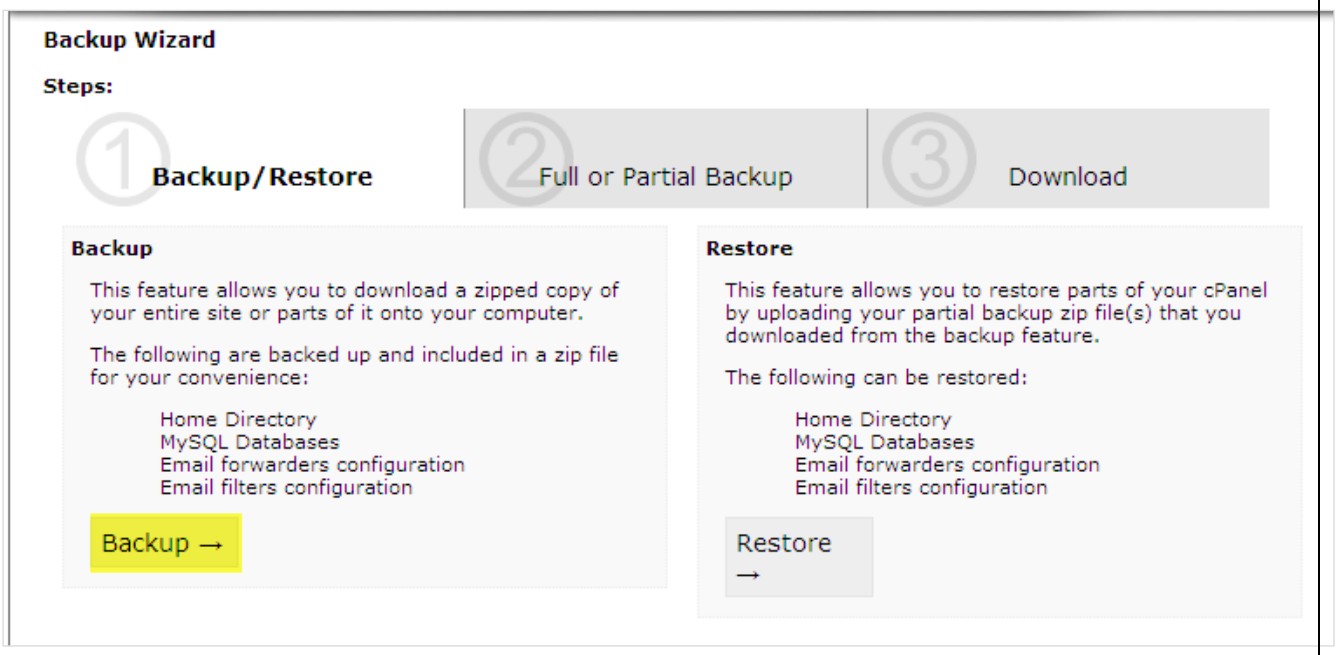
If you don't want to use a third-party service or plugin, you can always back up your blog manually through your server's administration section. Here I describe how to perform a backup from within cPanel. If you don't have cPanel, or you want to back up your blog from phpMyadmin instead, skip to the phpMyadmin section below.

From Within cPanel:

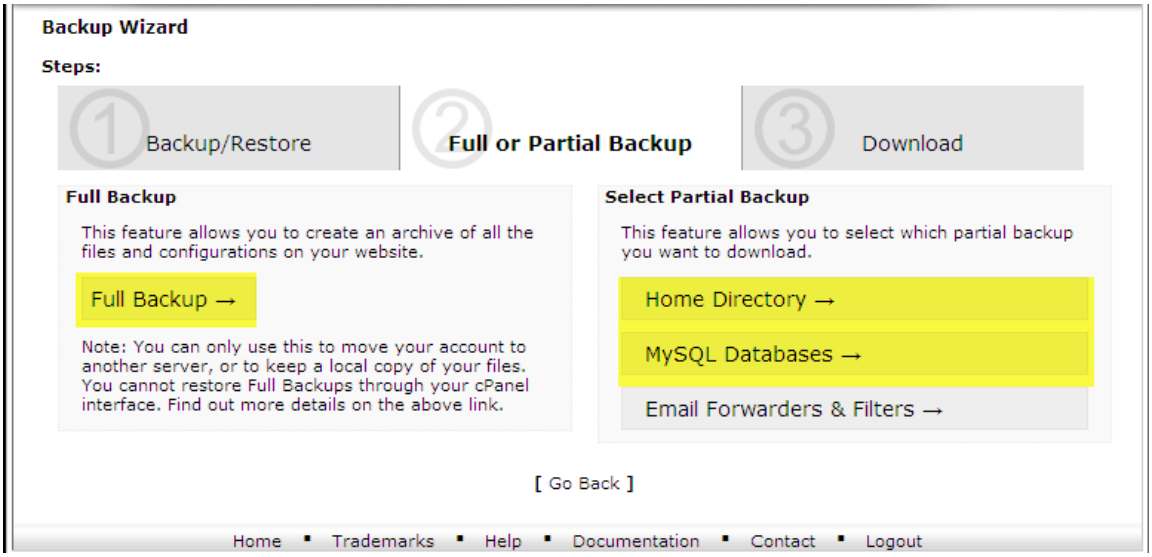
1. Log in to **cPanel**.
2. Click **Backup Wizard**.



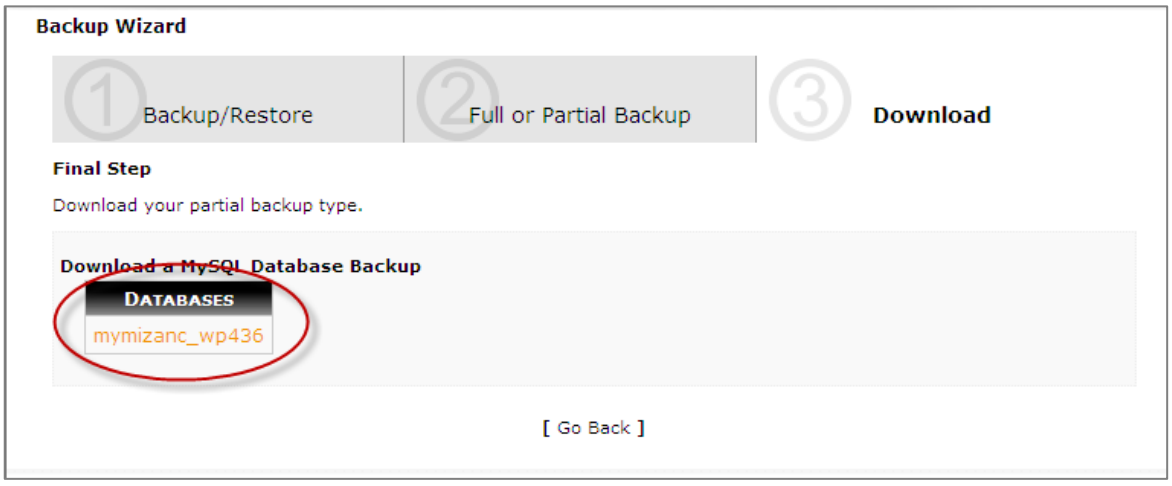
3. Click **Backup**.

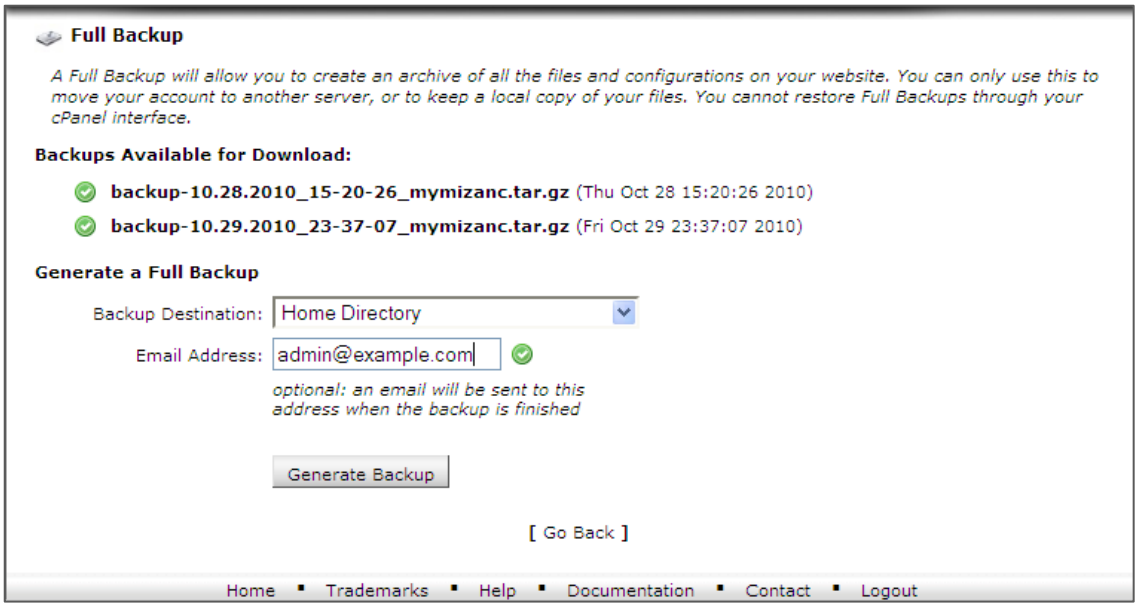


4. If this is the first time you're making a backup, select the **Full Backup** option. Otherwise, you may select the **Partial Backup** option. With the Partial Backup option, you'll need to backup your database and home directory separately. I always prefer to perform a full-backup everytime.



5. Click on the database you're archiving to start the download.



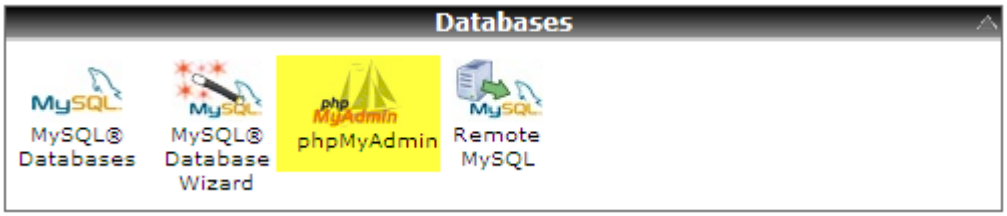


Because archives are kept on your web server, they can be downloaded at any time. It is always good practice to have a local copy of your precious sites in-case something goes terribly wrong with your webhost.

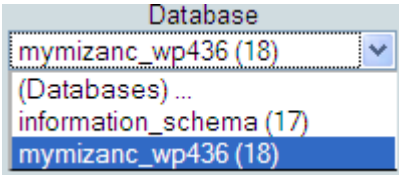
From Within phpMyadmin

PhpMyadmin is a better backup option for experienced users because it lets you pick and choose which items are archived.

1. Log in to your hosting control panel and find the database section.
2. Click **phpMyAdmin** from cPanel.

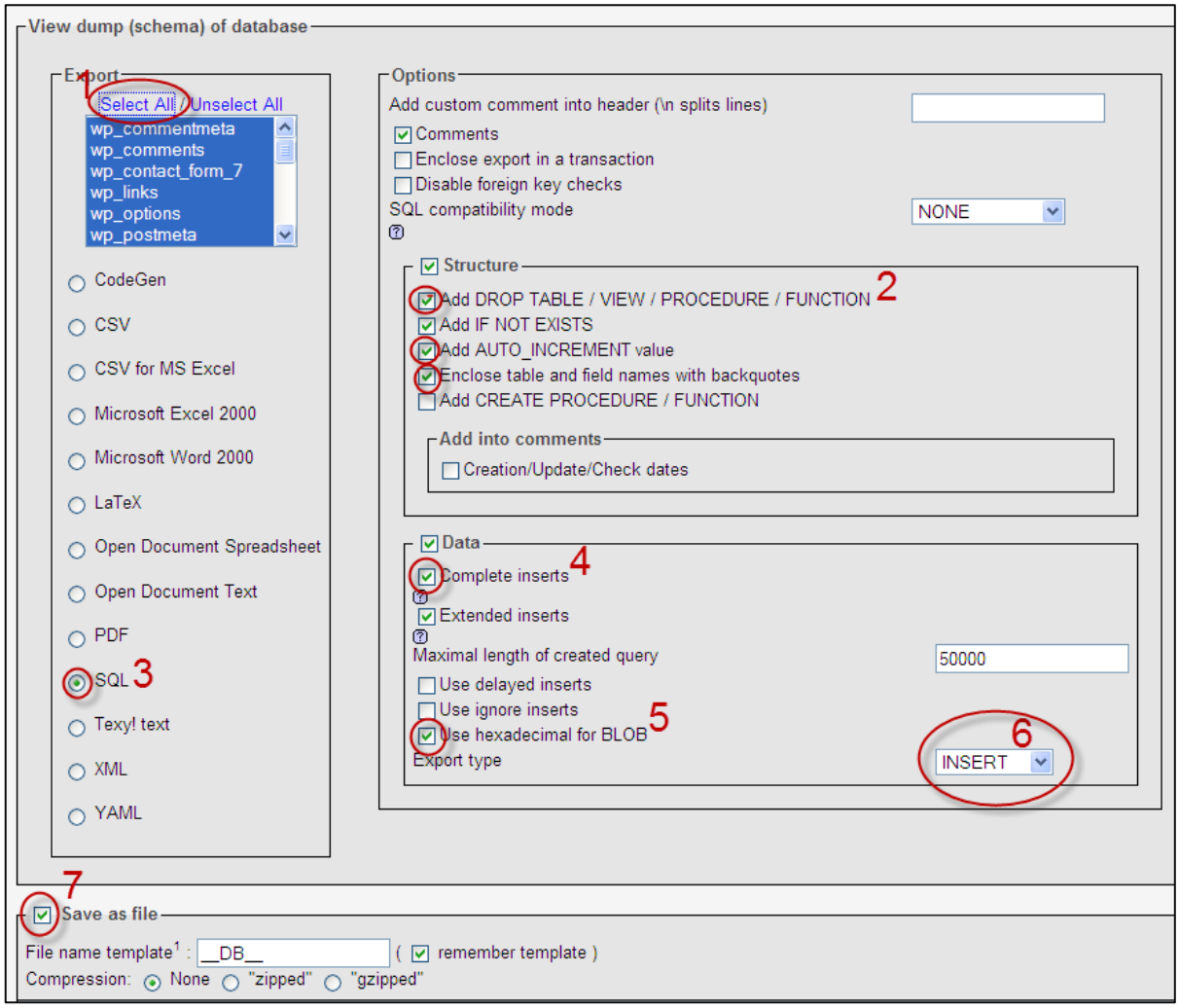


3. Select the database created during your WordPress installation.



4. Click the **Export** tab.

5. Select the marked options in the picture below. Only select or unselect additional options *if you know what you're doing*.



6. Click **Go** to download the archive. If the transfer is successful, your site is successfully backed up.

How Many Backups

Most users tend to replace their old archive with a new one. I don't recommend that since you might later need something located in an old archive that isn't available in a newer archive. I always recommend instead that you keep the three most recent backups on your local computer, at an online storage site, or on a CD. That way, if you later discover your blog needs an older plugin to function for example, and your most recent archive contains the newest version of the plugin only, you can access the functioning version from the first or second archive.

When to Backup

Earlier, I chastised some bloggers for not backing up their site on a regular basis, which ultimately begs the question, "How often should bloggers archive their sites?" I recommend that you make a backup before every major change to your web site if it is a low value site, or once or twice a day if your site is very valuable and updated frequently. The more frequently your site is updated, the more frequently it should be backed up. Ask yourself 'what would it cost me in the way of time or money if my site dissappeared now and I had to re-do all the work completed since my last backup?'. This should give you a good idea of how frequently you need backups.

If you're going to make significant database changes, for instance, backup the database before and after a significant edit. If you make a blog post every day, backup at least weekly.

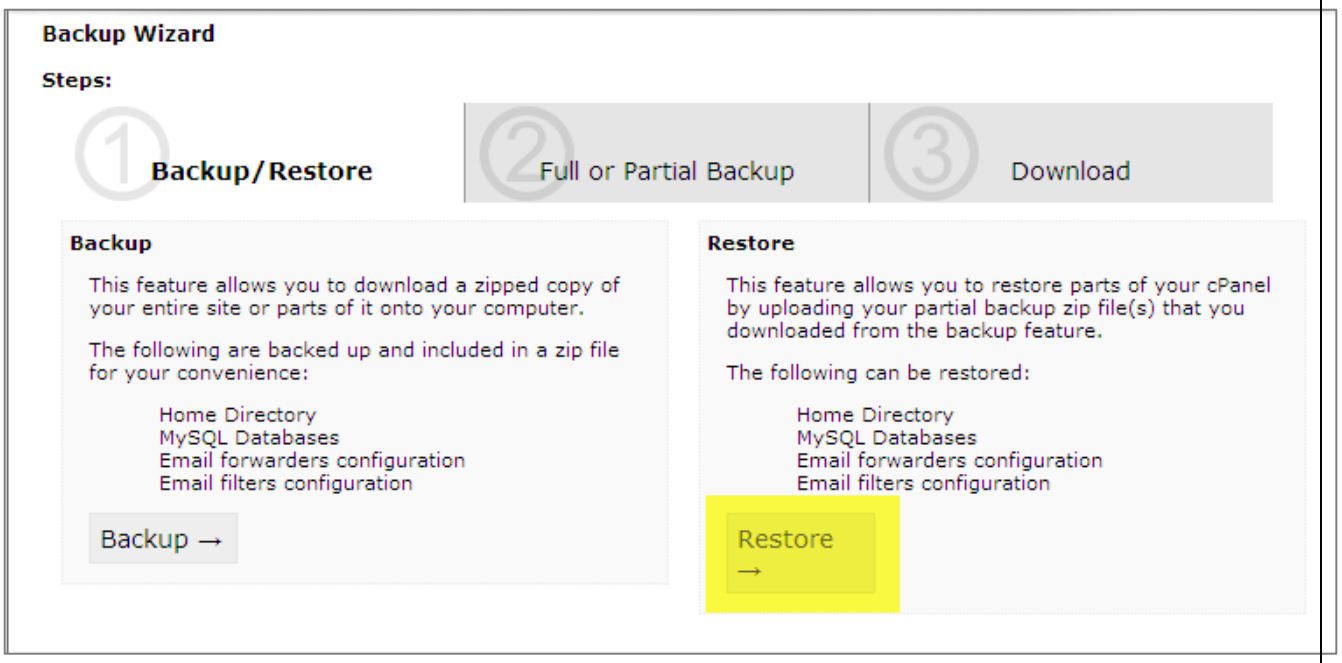
It may help to think about your blog as if it were a huge Microsoft document. Whenever you make a significant change to a large document, you're inclined to save it immediately so you won't lose that change should something go wrong with the computer. Approach your blog with the same diligence. Archive it right after making a significant change so you won't lose that change should something go wrong with your server or with the database itself.

Restore a Backup

Backing up a blog doesn't do much good without the ability to restore it. So here, we describe how to restore a blog to a prior state with a saved archive.

From Within cPanel

1. Return to the cPanel's Backup Wizard.
2. Select **Restore**.

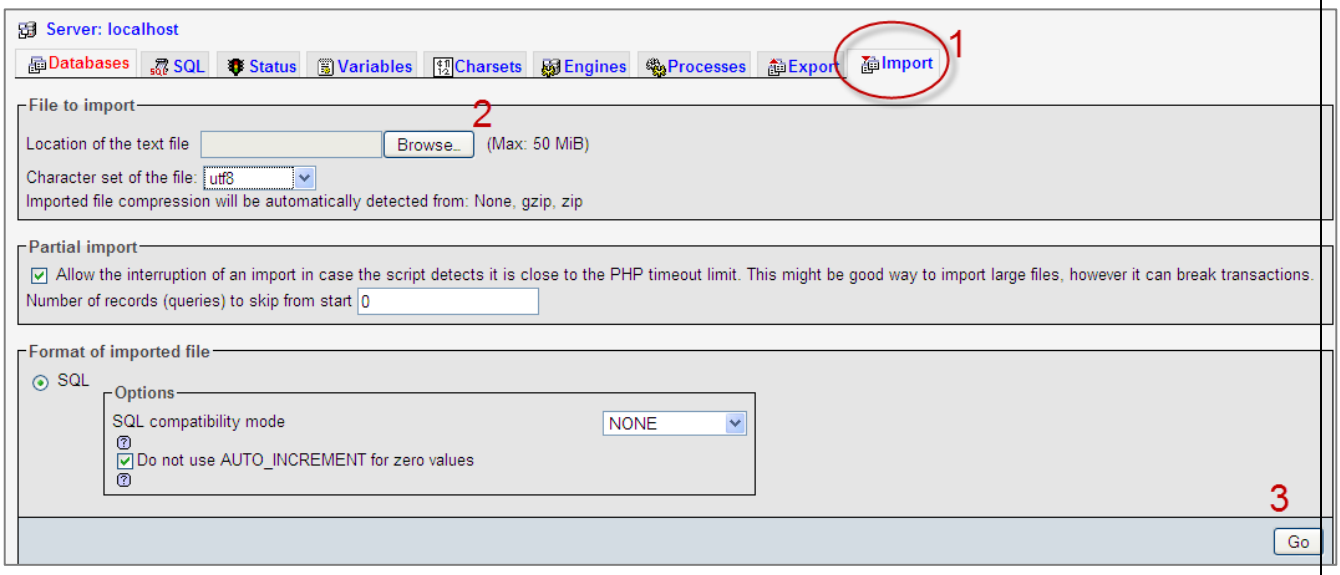


3. Select the proper restore type (MySQL Database).
4. Browse to the saved archive on your local computer.
5. Click **Upload**.

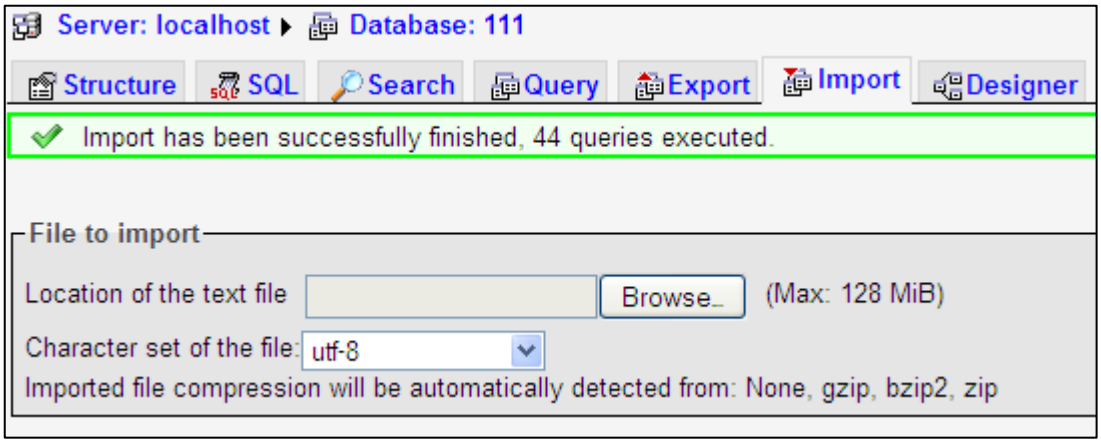
6. Your site should be restored within moments if no error occurs. Some types of errors you may face are network errors or hosting errors if your archive exceeds your host's size limit. Should you experience these errors, ask your site host how you can bypass this limit.

From Within phpMyadmin

1. Login to phpMyadmin.
2. Select your database.
3. Click **Import** and choose a file to restore.



4. Click **Go** to restore the database. Your database will be imported successfully.



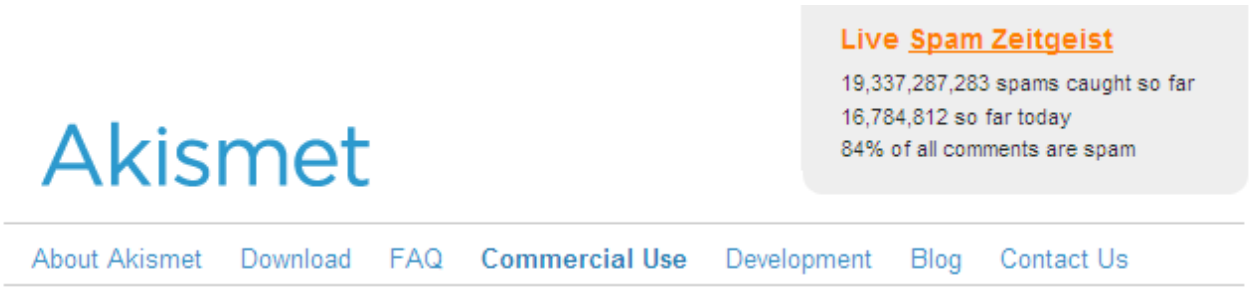
To properly restore a database, you **must** follow backup instructions properly. All it takes is one tiny mistake to corrupt an entire database!! Always make sure your backups are up to date and close to hand!

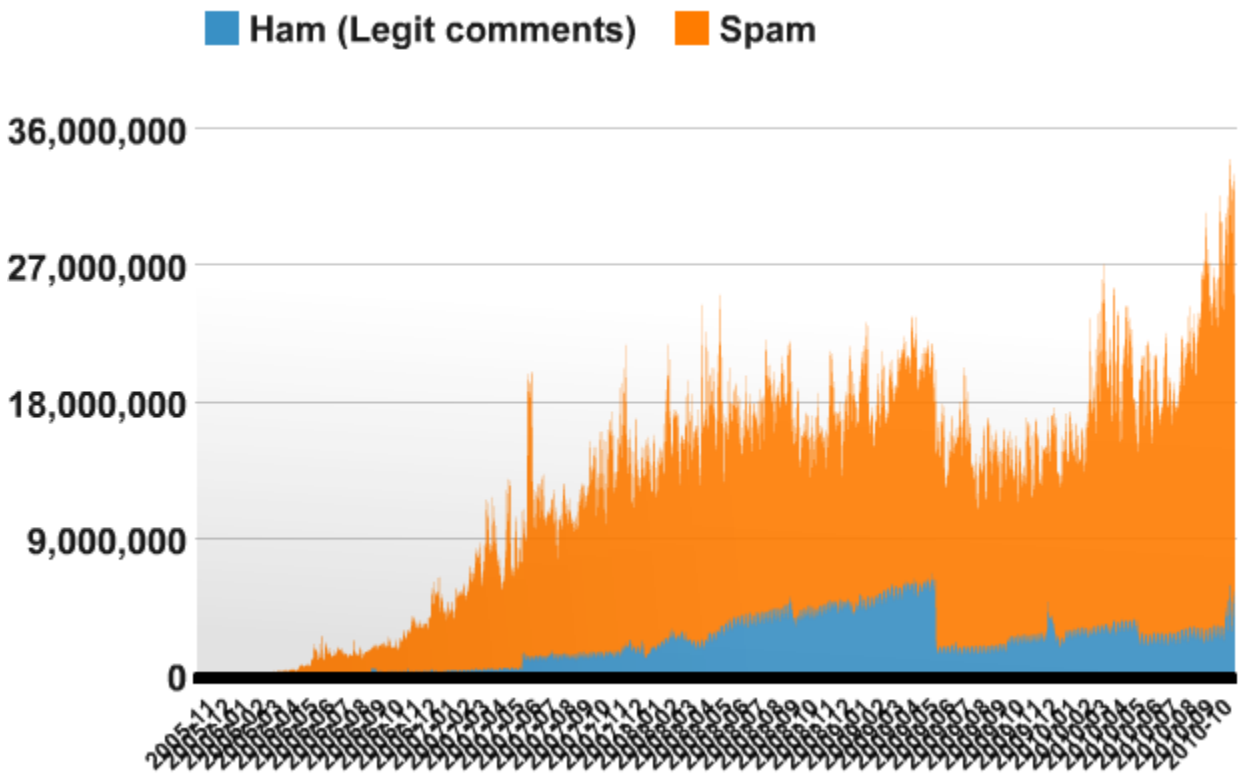
Comment Security

Akismet

Akismet is the de-facto anti-spam plugin for WordPress, and if you blog regularly, you really should have it installed. Akismet maintains a regularly updated database of spammers which it checks whenever a new comment is posted, along with fancy heuristics and other sophisticated profiling mechanics. If a new comment matches a spammer fingerprint, Akismet immediately sends that comment to a blog's spam queue.

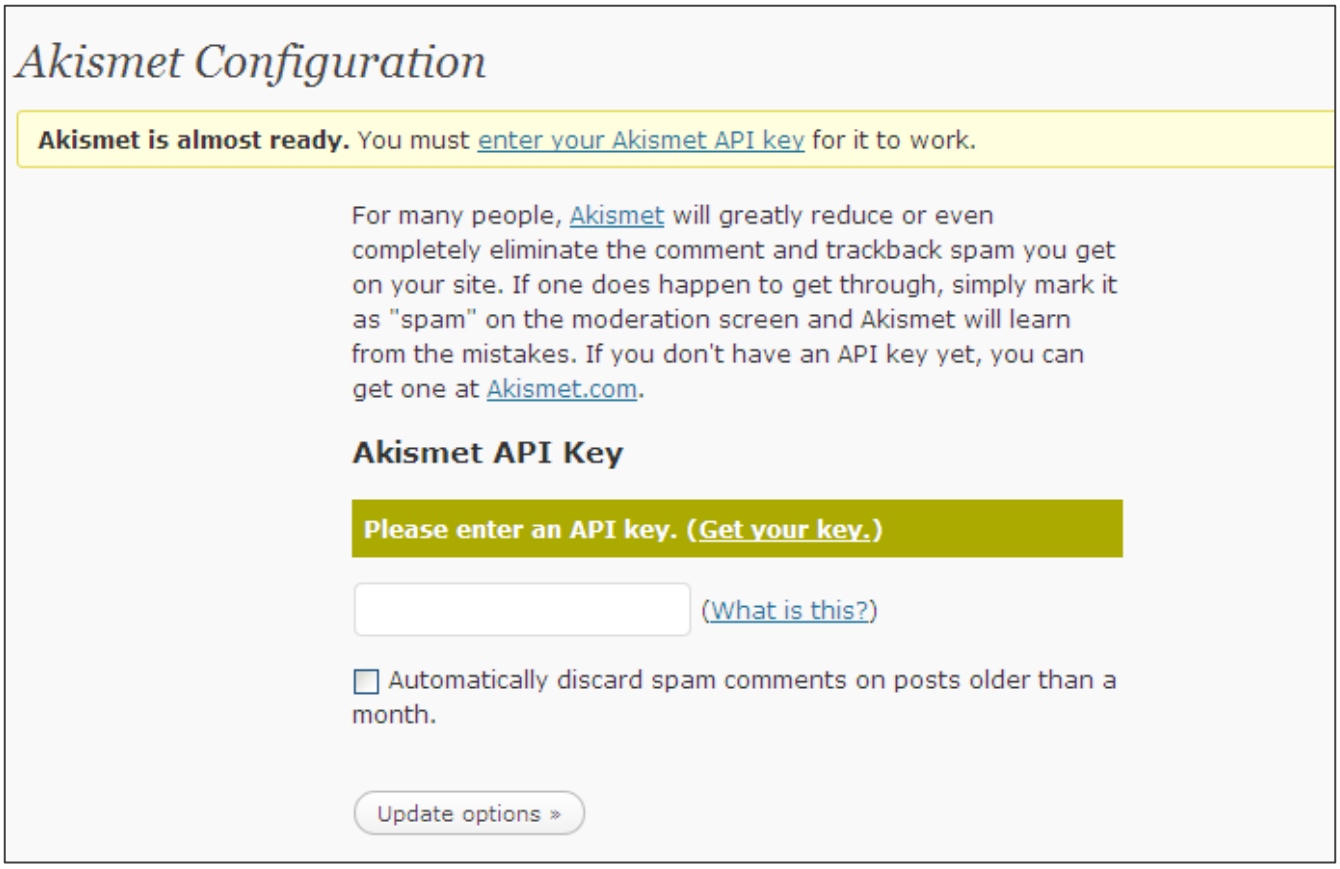
Non-commercial bloggers can use Akismet for free. If you earn at least $500/month, however, Akismet asks you to donate $5.00 to help with server costs and maintenance. WELL worth the money spent if you are in this position!





On the internet, 84% comments are spam!

When you first install WordPress, you'll notice this plugin is already installed. In order to use it, you'll need to get a unique API key and enter it on the plugin's settings page. Get your API Key from http://akismet.com/signup/.



Back at your blog's admin panel, enter your API key and save it with the **Update Options** button. From this point on, Akismet will protect your blog from comment spam. You don't ever need to touch the plugin again. It seamlessly integrates with your comment system and filters out spam on your behalf.

Secure Login Over SSL

SSL is an acronym for Secure Sockets Layer, and it provides a secure connection for transmitting sensitive data online. Data transmitted over SSL can be understood by you and the host you are communicating with only, making it a great security tool for keeping passwords and administration tasks/setting safe from hackers.

Visit http://www.ssl.com/ for more information about SSL.

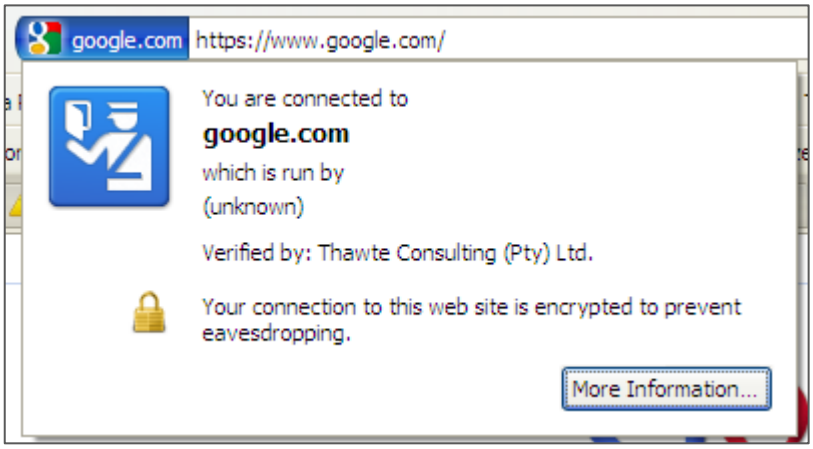
SSL is the same technology used by banks to keep customer information safe. But many web hosts don't offer SSL with their regular plans. You may need to upgrade your current plan if you want to access SSL security. For illustrative

purposes, we'll describe the use of *shared* SSL, which is registered by our web hosting and provided free of charge. With *shared* SSL, we share our web host's existing SSL certificate.

Here's how to enable this technology on your blog.

Ask your web host how you can enable shared SSL. Even though it's generally free, you'll need to set it up before it's available to your blog.

1. Access your blog with **https://** instead of **http://** in the address bar prior to logging in.



2. Next, open your **wp-config.php** file in a text editor.
3. If you want to use SSL for logging in only, add the following line to your wp-config.php file.

```
define('FORCE_SSL_LOGIN', true);
```

4. If you want to access both the login procedure and administration panel over SSL, add the following line instead.

```
define('FORCE_SSL_ADMIN', true);
```

5. You can now log out. Each subsequent request from this point on will be redirected to http**s** (even if http is accessed). In fact, with this configuration, you won't be able to access the site without using http**s**.

While providing SSL connections makes your blog very secure, it's admittedly tough to get. Not all hosts provide it – even the shared version! There will also be a bit of work required on the host side to get this happening. Talk with your host about how to organise this.

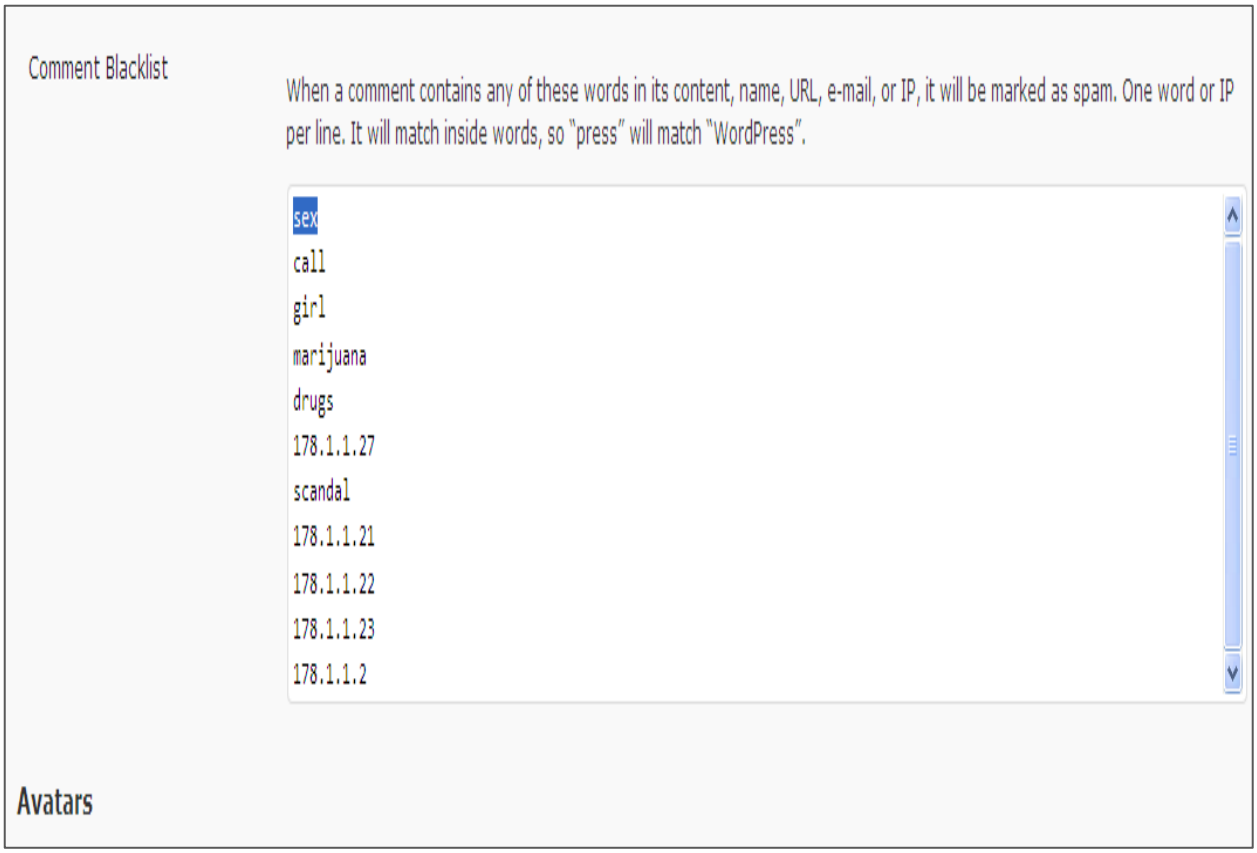
Blocking and Filtering

You never want to block a visitor, especially when your blog is a bustling community. But sometimes it's all you can do when a visitor continuously attacks you. And these attacks don't even need to be successful to cause problems. Visitors who use brute force attacks in an attempt to crack your password, for example, drain server resources and use up all your available bandwidth within a matter of days. This not only makes your host unhappy, it additionally raises your costs.

Here is an easy way to block users based on IP address.

With WordPress's Own Blacklist

1. Log in to WordPress.
2. Go to Settings > Discussion.
3. Locate the comment black list and add the IP numbers or words you want to block (filter). Put each IP or word on a separate line.



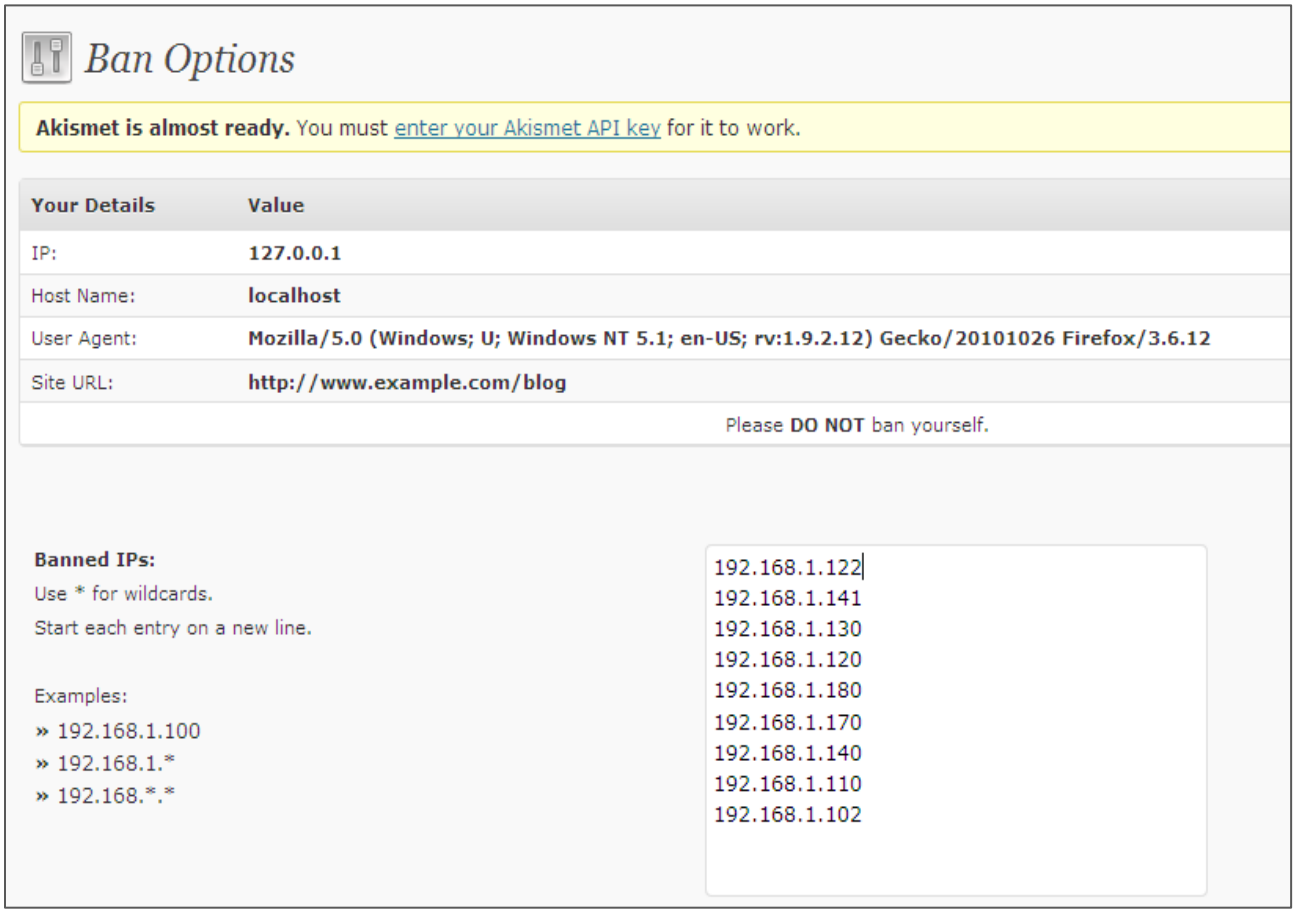
These IP numbers and filtered words are shown as examples only. They may not be what you want to block!

Whenever someone tries to access your blog with one of the listed IP numbers or filtered words, your blog denies them entry. It is true that there are many simple ways for an attacker or annoying surfer to get a new IP for themselves. Even so, this measure will stop some of the problem.

With the WP-Ban Plugin

Another way to block unwanted visitors is with the WP-Ban plugin. This is much better than using WordPress's built-in banner and I recommend it.

Install the WP-Ban plugin from the WordPress plugins repository, or from http://lesterchan.net/portfolio/programming/php/.



WP-Ban IPs and words shown here are for example purposes only.

They may not be what you want to block.

What to Do When You've Been Hacked

One fine morning, you may visit your site and notice a few oddities. The site isn't showing up, it's full of illegal advertisements, or its redirecting to some Godforsaken porno site. It might simply display a page that it's not supposed to. Looks like you've been hacked? What can you do? I don't want to go into too much detail here as the process of recovery can be very complicated depending on what has occurred. The following procedures should give you a head start though and may save you some money in having to call in an expert.

Scan Your computer

Believe it or not, a lot of the problems described above, start from *your* computer via viruses. So update your antivirus software and run a *full* system scan. Follow the instructions given from within your antivirus software to remove existing infections and prevent new ones. Worst case scenario, you may need to format your hard drive (after backing up all your important information to external media) and reinstall your operating system.

Restore Your Blog

You did back up your blog as instructed earlier, right? *Right!?* If you didn't, I bet you wish you had now! Restore your database and WordPress files using cPanel (or other method of choice) as instructed on page 68.

If You Didn't Make a Backup...

Tsk! Tsk! Tsk! Lucky for you, all is not lost. You can install and use the WordPress Exploit Scanner plugin to search for malicious strings in your database and remove them. Note that this plugin doesn't remove anything -- it just finds nasties for you. You have to remove all infections manually.

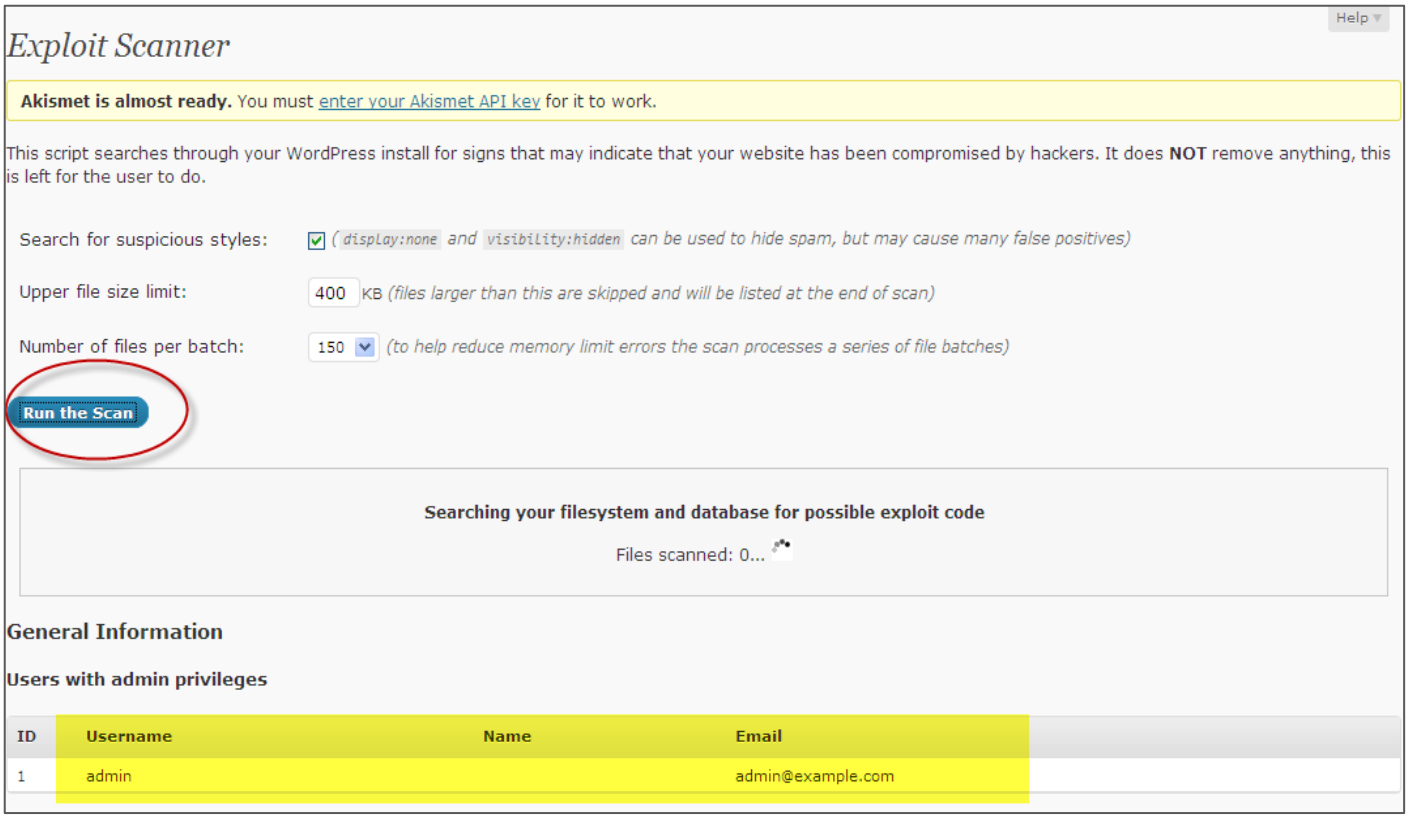
Install the WordPress Exploit Scanner plugin from the WordPress plugins repository, or from http://ocaoimh.ie/exploit-scanner/.

Results

Level Severe (17 matches)

Location / Description	What was matched
wp-content/plugins/wp-security-scan/simplepie.inc:12488 Used by malicious scripts to decode previously obscured data/programs	`$data = base64_decode($data);`
wp-content/themes/parallelus-traject/js/galleryview/jquery.galleryview-2.0-pack.js:16 Often used to execute malicious code	`eval(function(p,a,c,k,e,r){e=function(c){return(c<`
wp-content/themes/parallelus-traject/js/jquery-1.4.2.min.js:16 Often used to execute malicious code	`async:false,dataType:"script"}):c.globalEval(b.text||b.textContent||b.innerHTML||""`
wp-content/themes/parallelus-traject/js/jquery-1.4.2.min.js:132 Often used to execute malicious code	`p;f.indexOf("javascript")>=0)c.globalEval(a);return a},param:function(a,b){function d(i,o`
wp-content/themes/parallelus-traject/js/jquery.metadata.js:2 Often used to execute malicious code	`rn data}if(data.indexOf("{")<0){data=eval("("+data+")")}};var getObject=function(data){if(typeof data!="string"){return data}data=eval("("+data+")");return data};`
wp-content/themes/parallelus-traject/js/jquery.nivo.slider.pack.js:14 Often used to execute malicious code	`eval(function(p,a,c,k,e,d){e=function(c){return(c<`
wp-content/themes/parallelus-traject/php.ini:968 Often used to execute malicious code	`; error_reporting(0) around the eval().`
wp-content/themes/parallelus-traject/theme_admin/HTML-Version/js/jquery-1.4.2.min.js:16 Often used to execute malicious code	`async:false,dataType:"script"}):c.globalEval(b.text||b.textContent||b.innerHTML||""`
wp-content/themes/parallelus-traject/theme_admin/HTML-Version/js/jquery-1.4.2.min.js:132 Often used to execute malicious code	`p;f.indexOf("javascript")>=0)c.globalEval(a);return a},param:function(a,b){function d(i,o`
wp-content/themes/parallelus-traject/theme_admin/HTML-Version /js/jquery.metadata.js:2 Often used to execute malicious code	`rn data}if(data.indexOf("{")<0){data=eval("("+data+")")}};var getObject=function(data){if(typeof data!="string"){return data}data=eval("("+data+")");return data};`
wp-content/themes/parallelus-traject/theme_admin/HTML-Version /js/jquery.nivo.slider.pack.js:14 Often used to execute malicious code	`eval(function(p,a,c,k,e,d){e=function(c){return(c<`
wp-content/themes/parallelus-traject/theme_admin/js/jquery-1.4.min.js:16 Often used to execute malicious code	`async:false,dataType:"script"}):c.globalEval(b.text||b.textContent||b.innerHTML||""`

Example Scanner Result

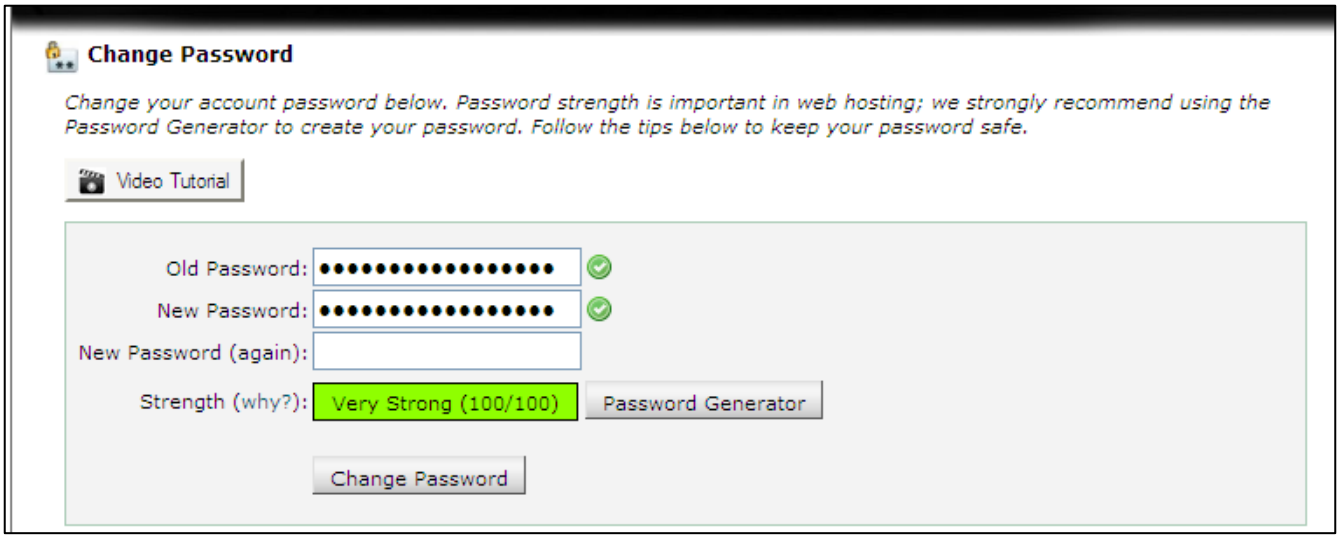


Check the user names as well. If your blog serves many users, check the list one by one.

Change Your Password

When facing a possible hacking, you must change your cPanel, FTPS, MySQL, and WordPress password. If your own computer was infected, change its password too. Then encourage all your blog users to change their own login passwords. It is a huge pain, but very important.

Starting to see how important making your blog as secure as possible is?



Change your Secret Keys

If a hacker stole your password, and that hacker is logged into your blog, they may remain logged in even if you've changed your password.

How? (Their cookies are still valid!!)

Disable all current cookies by generating and inserting a new set of secret keys (refer to page 49 for details).

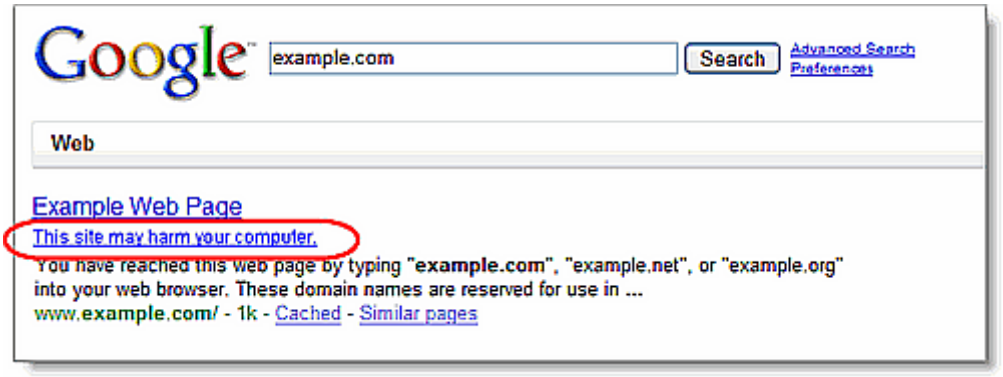
```
 *
 * @since 2.6.0
 */
define('AUTH_KEY',         'z6KY|,%JQo4ys+NAF-sTM#Sj]/h4cS,[yix+Iov-HMx6&3<zj6RY4XEnb;rf=,SI');
define('SECURE_AUTH_KEY',  '|QK0/`Im|n>1V;_lp?Ysi:3L#U!b(Po_{<2+Jg)[To>-$nZ:qn/[-E}`^x]yxBMy');
define('LOGGED_IN_KEY',    '(<bX<hIz{zItd<=Imh~i0AfTK^/UDat|tJN7z| }difNG61A-t4FYK9Ea/-W<3</');
define('NONCE_KEY',        '8Kd5-2)x|=A9P@ii_(Wd F*1R=BIW--^xI^T1fe7)i.J8|#g$_-g_U)|q.&,Y|Mf');
define('AUTH_SALT',        'gH<3Dso-A(7OPl:*p.@a[o[k4}]ZF~H0X9]S@j^MB<=pjx|bX7Mo24-$mu![3s++');
define('SECURE_AUTH_SALT', '#3`f*7GT/&&|.b8BH4[8b[E517/Nm$35spXCKW*Evqq9F#;MbV)*&5|z/4;[d![]');
define('LOGGED_IN_SALT',   'I6S74NQOTkq5*-O@-S_o`.A-pt%5fk$3U_,J,4+zbw{OPq+4}VNj$`18v,PYe!gD');
define('NONCE_SALT',       '@|(_|G[:=o:M^N2B+u?^/=s-F[P%Ti1[-&|Z0_pnjOIQ4>R**+$}ERW/pCX|Gz3K');

/**#@-*/

/**
 * WordPress Database Table prefix.
 *
 * You can have multiple installations in one database if you give each a unique
 * prefix. Only numbers, letters, and underscores please!
 */
```

Check .htaccess

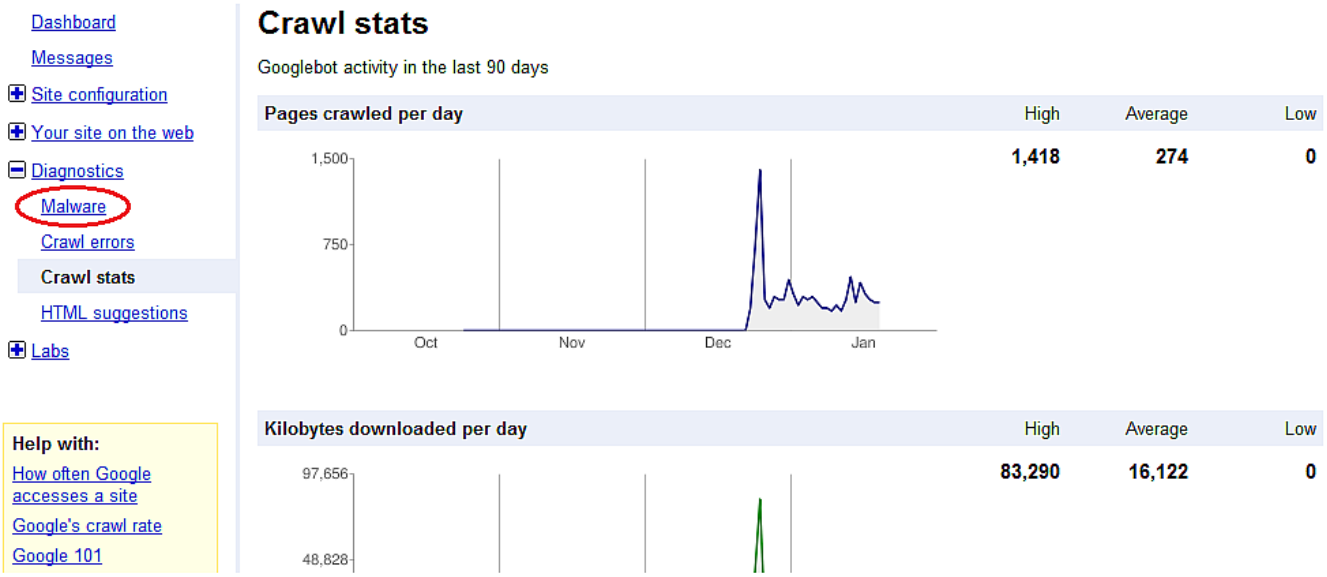
Check your **.htaccess** file to see if it contains injected code. If you see anything that shouldn't be in there, delete it and then save the file.

Google Your Blog

Google blacklists infected websites and it warns visitors who may encounter them in search engine results. Google your site address to see if it accompanies a warning message of some sort.



If you *do* see a warning, run over to Google Webmaster tools and click the Malware link under the Diagnostics link. Then follow the instructions given to remove the warning from your search engine listing.



Google Webmaster Screen

What To Do Now

Assuming you've followed the advice in this book, you can sleep just a little bit easier lnowing you have given your blog its best chance of survival in the wild untammed landscape of cyber space.Your WordPress blog is now more secure than ever, and your only worries now are those associated with generating more traffic, selling to more customers, and all that other 'fun' marketing stuff.

This doesn't mean you can turn your back on potential attacks, however. You *must* remember hackers never stop hacking – especially when they're trying to hack into a highly popular blog. They always look for new vulnerabilities and additional ways to break into websites, so every once in a while, do a little 'security' maintenance.

Check to make sure your blog's plugins are up-to-date, ensure your antivirus software is current, and review the steps outlined in this guide. They say an ounce of prevention is worth a pound of cure, and that couldn't be more relevant to blog security. So enjoy it while you have it, and *maintain* it so you can keep it!

Additional Resources

Codex
Codex is WordPress's documentation. It's one of the first places you should visit if you face a problem with your blog.
URL: http://codex.WordPress.org/Main_Page

Support Forum
WordPress has a support forum as well. Use it to get help immediately if you don't' find the solution in Codex.
URL: http://WordPress.org/support/

WordPress News
The latest news from WordPress.
URL: http://WordPress.org/news/

IRC
Internet Relay Chat, or IRC, is the precursor to both instant messaging and contemporary browser-based chat rooms like those found on Yahoo!, MSN, and innumerable other web sites. It is one of the best ways to get help with Wordpress, fast.
URL: http://codex.WordPress.org/IRC

Mailing List
WordPress has a number of email lists focused on different facets of development. This mailing list is for developers only.
URL: http://codex.WordPress.org/Mailing_Lists

WordPress TV
Your visual resource for all things WordPress related.
URL: http://WordPress.tv/

WordPress Podcast
Podcasting is distributing audio or video content via RSS 2.0, or Atom. Podcast clients such as iTunes, Juice, or CastPodder (linux) allow you to subscribe to RSS/Atom feeds and automatically download content to your portable audio player as it becomes available.
URL: http://codex.WordPress.org/Podcasting

WordPress Planet
An aggregated blog by WordPress.
URL: http://planet.WordPress.org/

WPEngineer.com
WordPress News, Hacks, and Tutorials.
URL: http://wpengineer.com/

Wordcamp
WordCamp is a conference that focuses on everything WordPress.
URL: http://central.wordcamp.org/

Index