

# WordPress 白皮书

BlogSecurity.net 著

Andor Chen @ [WordZine](#) 翻译

## 目录

目录.....	1
简介.....	2
安装 WordPress.....	2
设置 WordPress 表的权限 .....	2
更改数据库表的前缀.....	3
安装之前.....	3
手动修改.....	4
WP Prefix Table Changer 插件.....	5
为博客做好准备.....	6
更改管理员用户名.....	6
创建受限权限的用户.....	7
强化 WordPress 安装.....	9
限制 wp-content 和 wp-includes.....	9
限制 wp-admin.....	9
限制所有但除了自己的 IP.....	9
需要密码 - .htpasswd.....	10
.htaccess 文件 .....	10
.htpasswd 文件 .....	10
SPAM .....	11
Blog 加密 .....	12
重要插件 .....	13
去除 WordPress 版本信息 .....	13
去除数据库错误信息.....	13
超越安全 .....	14
WPIDS -侵入检测 .....	14
WordPress Plugin Tracker – 今天，你升级了嘛？ .....	14
WordPress 在线安全扫描 .....	15
结束语 .....	15

## 简介

本白皮书为你提供了提高博客安全性的所有必要信息，我们试图将步骤简化以便理解，没有过多的讨论纯技术层面的问题，所以你可以很容易的按照我们的步骤实施而不必担心会出错。此书所列举的内容可以在 [BlogSecurity.net](http://BlogSecurity.net) 找到，这里只是一个简要的向导。我们会积极地升级本书，请及时的关注我们。

如果你有任何的疑问，问题，好的点子或其他任何相关的问题，请和我们[联系](#)。

**注意：**在使用本书中任何的方法之前请为你的 WordPress 做一个完整的备份，包括 WordPress 文件和数据库。请参照“[升级 WordPress 的五个安全步骤](#)”一文。

## 安装 WordPress

### 设置 WordPress 表的权限

在安装 WordPress 之前有必要选择拥有合适权限的数据库用户种类，这一步很重要，最好选择被限制的用户权限。这样可以缓解数据库欺诈的问题，为安全加上了一道坚固的防线。

*注意：如果你使用的不是自己的主机，那么你可能就没有 MySQL 的 root 权限，这时可以跳过这一步。*

首先，使用 root 用户登录 MySQL 新建一个用于安装 WordPress 的数据库：

```
$ mysql -u root
mysql> CREATE database wp;
Query OK, 1 row affected (0.00 sec)
```

接下来创建一个用户，这个用户将被限制权限，只能连接数据库。同时，我们确保这个用户只能在本地操作数据库，而不能从远程操作。

```
mysql> GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP
-> ON wp.*
-> TO 'wpuser'@'localhost'
-> IDENTIFIED BY 'strongpassword';
Query OK, 0 rows affected (0.01 sec)
```

请确保用户的密码足够复杂，某些 WordPress 版本会将密码用于其他的地方。

好了，下面准备设置 wp-config.php 。

## 更改数据库表前缀

### 安装之前

接下来在 WordPress 根目录中新建文件 `wp-config.php`，将设置改为我们刚刚创建的用户信息：

```
// ** MySQL settings ** //
define('DB_NAME', 'wp'); // The name of the database
define('DB_USER', 'wpuser'); // Your MySQL username
define('DB_PASSWORD', 'strongpassword'); // ...and password
define('DB_HOST', 'localhost'); // 99% chance you won't need to change this value
define('DB_CHARSET', 'utf8');
define('DB_COLLATE', '');

// Change SECRET_KEY to a unique phrase. You won't have to remember it later,
// so make it long and complicated. You can visit https://www.grc.com/passwords.htm
// to get a phrase generated for you, or just make something up.
define('SECRET_KEY', 'A49D0EA936EFFFE30BAD7BACBA466CC897636F74BAB91128A96C9EF8C25F0
249');
// Change this to a unique phrase.
// You can have multiple installations in one database if you give each a unique prefix

$table_prefix = 'wp_4i32aK_'; // Only numbers, letters, and underscores please!
```

请留意用粉红色标记出的文字。我们使用刚刚新建用户的信息设置了配置文件，包括用户名、密码、数据库名等。同时我们使用 `grc.com` 为 `SECRET_KEY` 创建了一个很长的字符串，这一功能只在 WordPress 2.5 中提供，所以如果你使用的是旧版本这一步就请忽略吧。最后，我们随机的选择了一个六位数的字符串添加到数据库表前缀中。

为了缓和注入式安全威胁，有必要将 WordPress 默认的前缀改为更加随即的字符串，像 `4i32aK_`。很多的袭击者通常会使用一些常用信息，对于 WordPress，他们常常依赖于常用的数据库表前缀 `wp_` 来实施攻击。更改前缀会使得攻击者更难展开攻势。

我们可以查看数据库中的表来查看我们所使用的前缀，像 `wp_4i32aK_` 或 `wp4i32aK_` 之类的。最重要的一点是确保要将它改为一个攻击者不容易猜到的前缀。

## 手动修改

如果你要为一个已经使用了预设前缀的博客更改前缀的话，过程可能比较乏味，但你可以跳到本节末尾查看我们提供的很酷的自动更改前缀的插件。

首先打开 **WP-CONFIG.PHP**，修改以下这一行：

```
$table_prefix = 'wp_';
```

将它修改为前例中所使用的 **4i32aK\_**：

```
$table_prefix = '4i32aK_';
```

现在我们需要把 WordPress 数据库中的所有表重命名以确保前缀和我们刚刚修改的一致。由于 WordPress 不允许直接修改，我们要用到类似 PHPMyAdmin 的工具，通过 SQL 命令<sup>1</sup>来修改。

下面这些表：

```
wp_categories, wp_comments, wp_link2cat, wp_links, wp_options, wp_post2cat,  
wp_postmeta, wp_posts, wp_usermeta, wp_users
```

应该修改为：

```
4i32aK_categories, 4i32aK_comments, 4i32aK_link2cat, 4i32aK_links, 4i32aK_options,  
4i32aK_post2cat, 4i32aK_postmeta, 4i32aK_posts, 4i32aK_usermeta, 4i32aK_users
```

这时你可能要问了，这样做对吗？事实正式如此。WordPress 中的一些值也用到了前缀，所以我们要先把这些值修改了。

在 `wp_options`<sup>2</sup> 表中，我们要把 `option_name` 字段的记录从 `wp_user_roles` 修改为 `4i32aK_user_roles`<sup>3</sup>。

现在还要把 `wp_usermeta` 表里的另外两个值<sup>4</sup>也修改了。`Meta_key` 字段的 `wp_autosave_draft_ids` 和 `wp_user_level` 要修改为 `4i32aK_autosave_draft_ids` 和 `4i32aK_user_level`。

OK 了！不过 BlogSecurity 开发了 [WP Prefix Table Changer](#) 插件，可以将这一过程简化。请在使用该插件之前进行备份，这还是一个 Alpha 版本的插件，难免存在 bugs。

<sup>1</sup> 查询示例：`RENAME TABLE wp_categories TO 4i32a_categories`

<sup>2</sup> 我们使用了原有的前缀，以免和新的前缀混淆

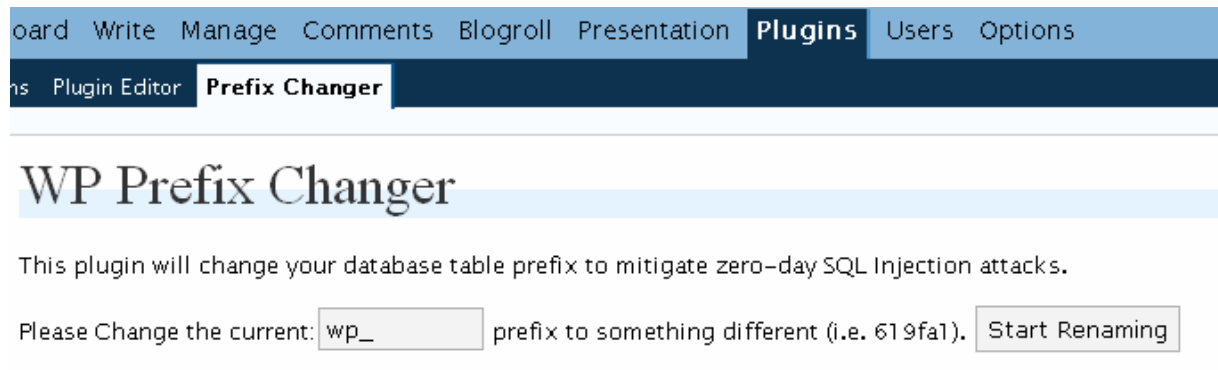
<sup>3</sup> `UPDATE 4i32a_options SET option_name='4i32a_user_roles' WHERE option_name='wp_user_roles' LIMIT 1`

<sup>4</sup> **注意：**有可能这一字段还没有出现，因为他们只有在需要时才会被创建，一旦被创建会被自动的赋予正确的值

## WP Prefix Table Changer 插件

我们开发了 [WP Prefix Table Changer](#) 插件，可以自动的完成上面的所说的修改数据库表前缀和所要修改的值。

下载之后，将文件解压到插件目录 **WORDPRESS/WP-CONTENT/PLUGINS** 中，然后登陆后台激活该插件。激活之后会在 **Plugins** 菜单下创建一个 **Prefix Changer** 子菜单，点击之后会看到类似下面的页面：



正如你所看到的，在这个博客中使用了默认的前缀，将它修改为某些随机的或具有意义的字符创，像我们前面使用过的 **4i32aK\_**。修该完之后点击 “**Start Renaming**”，插件就开始将所有表的前缀从 **wp\_** 修改为你所设置的值。请注意，第三方组件的表也同时被修改了，它们也需要新的前缀。

插件的最后一步是修改 **WP-CONFIG.PHP** 中所设置的前缀。

成功或失败之后都会得到一个信息，如果成功了 **WP- CONFIG.PHP** 文件为了安全起见会被设置为只读属性 (**644**)，如果失败了该文件可能会使只读属性，而且需要手动修改。

## 为博客做好准备

现在博客已经安装了，同时也做了一些基本的安全防范措施，很好！接下来我们要修改默认的 admin 用户，然后创建一个用于日常使用的用户。最后我们会安装 **Role Manager** 插件来细分各种用户所被赋予的权限。

## 修改管理员用户名

你应该把默认的管理员用户名从 **admin** 修改为一个更难被猜到的名字，因为现行的 WordPress 版本都很容易因为[用户枚举](#)而被攻击。这样做可以缓和暴力破解密码的攻击。

*注意：你应该假设袭击者会知道你的用户名，所以确保密码足够的复杂。我们已经强调过很多次了！*

使用 “wpuser” 连接到 MySQL，修改默认的管理员用户名：

```
wp $ mysql -u wpuser -p
mysql> use wp;
UPDATE 4i32aK_users SET user_login='admin', user_login='adm';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

我们已经将 “admin” 更改为 “adm” 了。显然你应该使用一些不那么显而易见的用户名。

## 创建受限权限的用户

进行本步之前要下载一份 [im-web-gefunden](#) 的 [Role manager](#) 插件。该插件可以为每个用户细分用户权限。激活之后，创建一个新用户并赋予相应的用户权限。建议首先为你自己创建一个用户，只为其分配日常处理所需的权限，像发表文章，验证评论等。确保只有管理员才有权限处理比较大的任务，像激活/卸载插件，上传文件，设置选项，更换主题，导入等操作。

*注意：你的用户获得的权限越少，你的博客就会更安全。“投稿”用户是很好的基础用户权限。*

默认的“投稿”用户可能没有足够的权限，但我们可以通过 [Role Manager](#) 增加所需权限。

我们建议赋予新用户“投稿”的权限，同时我们可以通过该插件做一些权限的扩展。如下所示。

### Assign extra capabilities

<input type="checkbox"/> Activate Plugins	<input type="checkbox"/> Create Users	<input type="checkbox"/> Delete Others Pages	<input type="checkbox"/> Delete Others Posts	<input type="checkbox"/> Delete Pages
<input checked="" type="checkbox"/> <b>Delete Posts</b>	<input type="checkbox"/> Delete Private Pages	<input type="checkbox"/> Delete Private Posts	<input type="checkbox"/> Delete Published Pages	<input type="checkbox"/> Delete Published Posts
<input type="checkbox"/> Delete Users	<input type="checkbox"/> Edit Files	<input type="checkbox"/> Edit Others Pages	<input type="checkbox"/> Edit Others Posts	<input type="checkbox"/> Edit Pages
<input type="checkbox"/> Edit Plugins	<input checked="" type="checkbox"/> <b>Edit Posts</b>	<input type="checkbox"/> Edit Private Pages	<input type="checkbox"/> Edit Private Posts	<input type="checkbox"/> Edit Published Pages
<input checked="" type="checkbox"/> <b>Edit Published Posts</b>	<input type="checkbox"/> Edit Themes	<input type="checkbox"/> Edit Users	<input type="checkbox"/> Import	<input type="checkbox"/> Manage Categories
<input type="checkbox"/> Manage Links	<input type="checkbox"/> Manage Options	<input checked="" type="checkbox"/> <b>Moderate Comments</b>	<input type="checkbox"/> Publish Pages	<input checked="" type="checkbox"/> <b>Publish Posts</b>
<input checked="" type="checkbox"/> <b>Read</b>	<input type="checkbox"/> Read Private Pages	<input type="checkbox"/> Read Private Posts	<input type="checkbox"/> Switch Themes	<input type="checkbox"/> Unfiltered Html
<input type="checkbox"/> Upload Files				

参考：[Role Manager](#) 获得更多信息

如果你的博客有很多个用户，为不同用户分配不同的权限是很好的主意。

创建用户时小心不要为不信任的用户赋予上传文件，插件处理，编辑文件/页面/文章，导入和未过滤的 HTML 等操作的权限。

★ **Administrator (rename)**

<input checked="" type="checkbox"/> Activate Plugins	<input checked="" type="checkbox"/> Create Users	<input checked="" type="checkbox"/> Delete Others Pages	<input checked="" type="checkbox"/> Delete Others Posts	<input checked="" type="checkbox"/> Delete Pages
<input checked="" type="checkbox"/> Delete Posts	<input checked="" type="checkbox"/> Delete Private Pages	<input checked="" type="checkbox"/> Delete Private Posts	<input checked="" type="checkbox"/> Delete Published Pages	<input checked="" type="checkbox"/> Delete Published Posts
<input checked="" type="checkbox"/> Delete Users	<input checked="" type="checkbox"/> Edit Files	<input checked="" type="checkbox"/> Edit Others Pages	<input checked="" type="checkbox"/> Edit Others Posts	<input checked="" type="checkbox"/> Edit Pages
<input checked="" type="checkbox"/> Edit Plugins	<input checked="" type="checkbox"/> Edit Posts	<input checked="" type="checkbox"/> Edit Private Pages	<input checked="" type="checkbox"/> Edit Private Posts	<input checked="" type="checkbox"/> Edit Published Pages
<input checked="" type="checkbox"/> Edit Published Posts	<input checked="" type="checkbox"/> Edit Themes	<input checked="" type="checkbox"/> Edit Users	<input checked="" type="checkbox"/> Import	<input checked="" type="checkbox"/> Manage Categories
<input checked="" type="checkbox"/> Manage Links	<input checked="" type="checkbox"/> Manage Options	<input checked="" type="checkbox"/> Moderate Comments	<input checked="" type="checkbox"/> Publish Pages	<input checked="" type="checkbox"/> Publish Posts
<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read Private Pages	<input checked="" type="checkbox"/> Read Private Posts	<input checked="" type="checkbox"/> Switch Themes	<input checked="" type="checkbox"/> Unfiltered Html
<input checked="" type="checkbox"/> Upload Files	<input checked="" type="checkbox"/> <i>User Level:</i> 10			

★ **Editor (rename, delete)**

<input checked="" type="checkbox"/> Activate Plugins	<input checked="" type="checkbox"/> Create Users	<input checked="" type="checkbox"/> Delete Others Pages	<input checked="" type="checkbox"/> Delete Others Posts	<input checked="" type="checkbox"/> Delete Pages
<input checked="" type="checkbox"/> Delete Posts	<input checked="" type="checkbox"/> Delete Private Pages	<input checked="" type="checkbox"/> Delete Private Posts	<input checked="" type="checkbox"/> Delete Published Pages	<input checked="" type="checkbox"/> Delete Published Posts
<input checked="" type="checkbox"/> Delete Users	<input checked="" type="checkbox"/> Edit Files	<input checked="" type="checkbox"/> Edit Others Pages	<input checked="" type="checkbox"/> Edit Others Posts	<input checked="" type="checkbox"/> Edit Pages

使用 [Role Manager](#) 更改用户权限

## 强化 WordPress 安装<sup>5</sup>

本节将讨论如何避免管理区域被越权操作。这一步很容易在单用户博客或多用户博客中实现，但对于多用户博客可能要麻烦一些。你可以自行决定是否要为确保博客安全而进行这一步，但我们建议你应该这么做。

### 限制 wp-content 和 wp-includes

这一步我们会限制这些目录的权限，基本上拒绝所有的东西，除了对图片，CSS 和一些 JavaScript 文件的请求。

把以下的代码放到你的 `.HTACCESS` 文件中，这些 `.HTACCESS` 文件应该放在 `WP-CONTENT` 和 `WP-INCLUDES` 目录下：

```
Order Allow,Deny
Deny from all
<Files ~ ".(css|jpe?g|png|gif|js)$">
    Allow from all
</Files>
```

*注意：你可以为某些主题或插件赋予 PHP 请求的权限。*

### 限制 wp-admin

#### 限制所有但除了自己的 IP

如果你的是一个单一用户博客，你可能需要限制通过 IP 连接 `WP-ADMIN` 的权限。请确保你所使用的是静态 IP。`WP-ADMIN` 中的 `.HTACCESS` 文件如下：

```
Order deny,allow
Allow from a.b.c.d #That's your static IP
Please add some example for allowed ip ranges
Deny from all
```

保存文件，然后试图通过代理访问 `wp-admin` 目录，应该会被限制连接，然后使用自己的 IP 再次连接。

如果设置的一切正常，`WP-ADMIN` 会被限制连接，除了通过你所设置的 IP。

---

<sup>5</sup> 原文在此：<http://blogsecurity.net/WordPress/article-210607/>

## 需要密码 - .htpasswd

当然，推荐的选项是设置密码保护，这意味着你仍然可以在任何地方连接到 `wp-admin` 目录，但我们增加了一道防线，以防越权操作。

### .htaccess 文件

WP-ADMIN 中的 `.HTACCESS` 文件如下：

```
#this file should be outside your webroot.  
AuthUserFile /srv/www/user1/.htpasswd  
AuthType Basic  
AuthName "Blog"  
require user youruser #making this username difficult to guess can help mitigate password  
brute force attacks.
```

### .htpasswd 文件

正如已经说明的，该文件<sup>6</sup> 可以放到网站目录之外的地方，上层目录是个很好的选择。

```
$ htpasswd -cm .htpasswd blog  
New password:  
Re-type new password:  
Adding password for user blog
```

`.htpasswd` 文件已经在当前目录中创建了，请确保本文件的地址符合 `wp-admin/.htaccess` 中 `AuthUserFile` 所设置的地址。

现在测试一下看看是否已经工作了。当你试图登录博客时会要求你输入用户名和密码以获得连接权限。如果没有出现的话，查看一下加密密码文件，然后查看所提供的地址是否正确。

---

<sup>6</sup> 关于该文件的更多说明：[http://httpd.apache.org/docs/1.3/mod/mod\\_auth.html](http://httpd.apache.org/docs/1.3/mod/mod_auth.html)

## SPAM

博客系统的一个很强大的功能是可以留言的方式获得反馈，但很不幸的是三个留言中有两个都是垃圾留言。所以为了和垃圾留言斗争，我们开发了很多的方法，然而一些方法可能会让读者们很苦恼，这样就使获得的反馈量变少了。

**图像验证码** - 就是常说的“不容易读的烦人的图片”。图像验证码可以很好的组织垃圾留言，但只在正确的实现方法之前提下（我们从 [Mustlive](#) 的图像验证码 bugs 中获知）。图像验证码主要缺点是它骚扰到我了，或者已经骚扰了大多数的人。谁 TMD 想每次留言都要戴上眼睛斜视着阅读这些图片验证码。

**验证** - 留言需要注册用户。自动化的程序通常可以自动的处理，所以图像验证码系统现在处于统一战线了。至少我们只需要输入验证码，对吧，但现在我们需要能够留言而每次都要登陆，还要多记住一个密码！[OpenID](#) 可能是一个好的解决方法。

**黑名单** - 很多软件都提供了黑名单功能，这意味着你可以通过某些“坏的”单词来阻止垃圾留言。我不知道你们有没有现场观看过阿森纳和曼联的比赛？同一事物我们可以通过很多方法表述，和其他方法相结合时黑名单或许是很好的方法，但单独使用时就不怎么强大了。

**JavaScript** - BlogSecurity 的 SpamBam 插件通过客户端的浏览器脚本来确认读者是否正在使用一个有效地浏览器。就目前而言这是一个很好的防止垃圾留言的方法，大多数的垃圾留言制造者都是通过一些不支持 JS 的自动化程序来实现的。我很喜欢这个插件，虽然我认为这是一个很酷的方法，但还是有待加强功能。

**Smart Checks** - 一些类似于 Akismet 的垃圾防治系统。它们有大量的核对列表和一系列的黑名单来验证是否为垃圾留言。平常使用起来很好，但你的留言要被送到第三方服务，会导致过量的流量损失。

有很多的插件可供选择，我们建议你使用以下的一种或两种：

[Akismet](#) - Automattic 的防垃圾留言插件(需要提供 API key)

[SpamBam](#) - BlogSecurity 的防垃圾留言插件

## Blog 加密

登录博客时请确保是通过 HTTPS 协议，这样可以防止袭击者通过 HTTP 协议的明文（未加密）传送来抓取用户名和密码。



首先检查你的博客是否支持 HTTPS 协议，很简单，在你的地址栏中输入以下地址：

`https://yourblog/`

安装 HTTPS 协议已经超出本书所讨论的范围了，记住，Google 就在你身边。我们继续。

一旦安装了 SSL(HTTPS)，我们还需要安装一个插件，使得我们登录时自动转向对应的 HTTPS 协议地址。

你猜怎么着，BlogSecurity 就提供了这样的插件，你可以在[这里](#)获取 bs-wp-encrypt 插件。

下载后将 bs-wp-https\_php.txt 重命名为 bs-wp-https.php 然后上传到 wp-content/plugins/ 目录，然后激活之，现在你的博客已经能够识别转向区，并且会自动转向 HTTPS。

## 重要插件

BlogSecurity 提供了一些插件可以让袭击者的日子不是那么好过，我们建议你们安装这些插件。

### 去除 WordPress 版本信息

BlogSecurity Wordpress Noverion 插件 (bs-wp-noverion) 可以防止 WordPress 版本漏洞，另一个简单但超级实用的插件。

很多的袭击者或自动程序会在实施攻击以前试图获取软件的版本，去除 WordPress 的版本信息可以使某些基于特定版本进行袭击的袭击者们失去信心。

*注意：该插件可能对那些依赖于 WordPress 版本信息的插件产生影响。*

bs-wp-noerrors 插件可以在[这里](#)下载。

### 去除数据库错误信息

该插件不支持 **WordPress 2.3.2**，因为其本身已经将错误信息关闭了。但还是可以在旧的 **WordPress** 中使用。

WordPress 默认的将错误信息打开了：

```
function show_errors() {  
    $this->show_errors = true;  
}
```

*注意：虽然 PHP 错误信息被关闭，但数据库错误信息仍然会显示。该插件可以关闭 WordPress 数据库错误信息，以防某些敏感信息被暴露出来，比如数据库前缀。*

bs-wp-noerrors 插件可以在[这里](#)下载。

## 超越安全

下面的插件或服务可以增强博客的安全性。

### WPIDS - 侵入检测

BlogSecurity 把 PHPIDS (Intrusion Detection System) 移植到了 WordPress。PHPIDS 可以检测到多种侵入企图。我们使用这一工具阻止危险的袭击。每个侵入都会登录数据库，所以你可以进行跟踪或采取一些处理步骤。如果影响超过了预设值的话你会收到一封警告邮件（每个入侵都有其威胁指数）。如果威胁指数很大的话你可以将袭击者的 IP 封上数日，同时 WPIDS 总是会试图防范坏的输入。你可以在 PHPIDS 的[官方网站](#)获取该插件。

注意：为了使该插件能够顺利的运行至少需要你的系统安装了 PHP 5.1.6。一个包含 BlogSec's recent addition, WP- Lockdown 的新版本即将发布，所以请及时的关注我们。

### WordPress Plugin Tracker – 今天，你升级了嘛？

如果你的博客程序或插件是直接从开发者的站点获得的，那么你可能已经使用了最新版本，你可以安装 [WordPress Plugin Tracker](#) 插件来跟踪插件，查看是否使用了最新版本。安装并激活插件之后，运行该插件查看你是否正在使用最新的插件，截图如下：

#### Plugin Release Tracker

Track the releases of the plugins you have installed in your website

Move WP Plugins Tracker to Plugins SubMenu

Plugin	Your Version	WPPDB Version	Status
<a href="#">Another Wordpress Meta Plugin</a>	2.0.3	2.0.3	Versions are matching, You have latest v
<a href="#">Akismet</a>	2.0.2	2.0.2	Versions are matching, You have latest v
<a href="#">Bad Behavior</a>	2.0.10	2.0.10	Versions are matching, You have latest v
<a href="#">http:BL WordPress Plugin</a>	1.4	1.4	Versions are matching, You have latest v

如果插件的版本过时了，你会被该插件提醒，点击左边的插件标题就会直接转到对应的插件页面，然后选择是否升级。很简单的保持插件最新的方法。

## WordPress 在线安全扫描

BlogSecurity 开发了一个安全检查工具，来检查你的博客是否存在基本的安全隐患。在处理枚举插件，跨站脚本漏洞等方面做的很出色。

### WordPress Version Leak

Test	Result
wp-links-opml.php	Version Leak: WordPress 2.2.1
wp-rss.php	Version Leak: WordPress 2.2.1
wp-commentsrss2.php	Version Leak: WordPress 2.2.1
wp-rdf.php	Version Leak: WordPress 2.2.1
wp-rss2.php	Version Leak: WordPress 2.2.1

According to wp-scanner this blog is running the latest version of WordPress.

### WordPress Template XSS Checks

Test	Result
wp-xss-3	WordPress Template Vulnerable to XSS: /?

This blog uses a template that is vulnerable to Cross-Site Scripting Attacks. See [Vulnerable WP Themes](#) for more information.

### WordPress Plugins Found

Test	Result
wp-plugins[1]	wp-backup
wp-plugins[2]	subscribe-to-comments.php
wp-plugins[4]	wp-contact-form
wp-plugins[0]	wp-cache2
wp-plugins[5]	sitemap
wp-plugins[3]	Akismet

Please check out [WordPress BlogWatch](#) for the latest vulnerabilities in WordPress plugins. More work will be done in this area for future releases.

EXCEPT WHERE OTHERWISE NOTED, CONTENT AND TOOLS ON THIS SITE ARE LICENSED UNDER THE [ATTRIBUTION-NONCOMMERCIAL-NODERIVS LICENSE](#)

WP-Scanner 是一个免费的在线服务，至今已经测试了 5000 余个博客（事实上统计数字已经丢失了），点击[这里](#)获取更多信息。

## 结束语

到该结束的地方了，我们希望你喜欢本书，同时已经成功的处理了这而安全问题。欢迎你将你的信息或故事[反馈给我们](#)。

## 关于翻译

本书由 Andor Chen 于 2008 年 4 月 22 日翻译，原文版权归 BlogSecurity.net 所有，本中文译本版权归 Andor Chen 所有，并保留最终解释权。本译本基于“[署名-非商业性使用-相同方式共享 3.0 Unported](#)”发布。