

Indice

Introduzione	
Installare WordPress	
Accedere alle tabelle di WordPress.....	
Cambiare il prefisso delle tabelle di WordPress	
Prima dell'installazione.....	
Cambiamento manuale.....	
Il plugin WP Prefix Table Changer.....	
Preparare il blog	
Cambiare lo username di Admin.....	
Creare un nuovo utente con permessi limitati.....	
Rendere più sicura l'installazione WordPress	
Limitare l'accesso alle cartelle wp-content e wp-includes.....	
Limitare l'accesso a wp-admin.....	
Bloccare tutti ad eccezione del vostro indirizzo IP.....	
Richiedere l'uso della password http: .htpasswd.....	
Il file .htaccess.....	
Il file .htpasswd.....	
Spam	
Crittazione del blog	
Plugin chiave	
Rimuovere la versione di WordPress.....	
Disabilitare la visualizzazione degli errori di WordPress.....	
Altre opzioni di sicurezza	
WPISD – Intercettare le intrusioni.....	
WordPress Plugin Tracker	
WordPress Online Security Scanner.....	
Conclusioni	

Introduzione

Questo documento fornisce tutte le informazioni necessarie per migliorare la sicurezza del vostro blog basato su WordPress. Proviamo a descrivere tutti i passi necessari in un modo chiaro, semplice e comprensibile senza entrare troppo nei dettagli tecnici, così che potete seguire le istruzioni senza avere particolari problemi nell'applicarle al vostro blog.

Tutte le informazioni possono essere trovate su BlogSecurity.net; questa guida serve come una pratico tutorial per mettere al sicuro il vostro blog. Aggiorniamo questo documento regolarmente, quindi controllate di tanto in tanto il sito per avere l'ultima versione disponibile.

Se avete domande, problemi, idee o qualsiasi altra questione relativa a questo documento, [contattateci](#).

Importante: prima di mettere in pratica le tecniche illustrate in questo documento, curatevi di eseguire un backup completo di files e database che compongono il vostro blog. Consultate il documento [5 modi per effettuare l'upgrade di WordPress](#) per maggior aiuto.

Installare WordPress

Accedere alle tabelle di WordPress

Prima ancora di installare WordPress, è importante scegliere il giusto tipo di utente del database, che abbia i permessi giusti. E' fondamentale usare un utente con permessi limitati: questo limita il rischio di perdita dei dati e fornisce un ulteriore livello di sicurezza.

Nota: se il blog è ospitato da un hosting condiviso, potreste non avere accesso da amministratore al database MySQL, per cui potete saltare questo paragrafo.

Prima di tutto, eseguite il login come utente "root" al database MySQL, e create il database che verrà utilizzato da WordPress:

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

Successivamente creiamo un utente del database: questo account disporrà di permessi limitati, ovvero potrà solo accedere al database appena creato, in locale e non in remoto.

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

Assicuratevi che la password scelta sia una password sicura, difficile da indovinare.

Bene, ora prepariamo il file 'wp-config.php'.

Cambiare il prefisso delle tabelle di WordPress

Prima dell'installazione

Creiamo un file chiamato 'wp-config.php' ed inseriamolo nella cartella principale di WordPress. Inseriamo il seguente codice, avendo cura di inserire i valori di esempio qui riportati con quelli che si riferiscono alla vostra installazione.

```
// ** configurazione MySQL ** //
define('DB_NAME', 'wp'); // Il nome del database
define('DB_USER', 'wpuser'); // il nome utente MySQL
define('DB_PASSWORD', 'strongpassword'); // ...la password
define('DB_HOST', 'localhost'); // il 99% delle volte non dovrete cambiare questo
define('DB_CHARSET', 'utf8');
define('DB_COLLATE', '');
// Cambiate SECRET_KEY con un valore a caso, univoco. Non avrete bisogno di ricordarvelo in seguito, per
// cui fate in modo che sia lungo e complicato. Potete visitare https://www.grc.com/passwords.htm
// per trovare valori da inserire, oppure inseritene uno che decidete voi.
define('SECRET_KEY', 'A49D0EA936EFFFE30BAD7BACBA466CC897636F74BAB91128A96C9EF8C25F0
249');
// Cambia questo in modo che sia univoco.
// E' possibile avere multiple installazioni sullo stesso database, dando ad ognuna un prefisso univoco per le
// tabelle.
$stable_prefix = 'wp_4i32aK_'; // Solo numeri, lettere, e underscore (_)!
```

Spero che vi piacciono i colori pastello che abbiamo usato. In pratica, abbiamo creato il nostro file di configurazione WordPress con le informazioni per accedere al database. Avete notato che abbiamo usato il sito "grc.com" per creare una SECRET_KEY bella lunga. Questa serve per le salted password di WordPress 2.5 ESCLUSIVAMENTE, ignorate questa funzione se state usando una versione più vecchia di WordPress. Infine, avrete anche notato che abbiamo scelto un valore casuale di 6 cifre per determinare il prefisso delle tabelle del database.

Infatti per limitare le probabilità di subire attacchi di database injection, dovrete cambiare il prefisso di default wp_ in qualcosa di random come 4i32aK_. Spesso i cracker utilizzano exploit che trovano in internet, distribuiti pubblicamente. Questi exploit molto spesso si basano sul fatto che la maggioranza delle installazioni WordPress usa come prefisso delle tabelle il valore wp_, per cui cambiare questo prefisso renderà le cose molto più difficile ad un eventuale sabotatore.

Per riconoscere facilmente quale prefisso avete utilizzato, potete creare il prefisso in modo che sia preceduto dalla dicitura wp, come wp-4i32aK_ oppure wp4i32aK_. L'importante è modificarlo, in modo che un hacker non possa intuire quale possa essere il prefisso delle tabelle WordPress.

Cambiamento manuale

Se il vostro blog è già stato installato con il prefisso di tabelle standard, è meglio provvedere per cambiarlo. Questo è un processo che può essere abbastanza complicato (al termine di questa sezione troverete informazioni riguardo al nostro plugin per automatizzare questo processo).

Il primo passo è aprire il file **WP_CONFIG.PHP** e modificare la linea

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

Come nell'esempio precedente, inseriamo il prefisso `4i32aK_`. La linea di codice ora dovrebbe essere

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

Fatto questo, dobbiamo rinominare tutte le tabelle WordPress perchè rispecchino il nuovo prefisso; per fare questo usiamo un comando SQL¹ all'interno di un'interfaccia web come PHPMyAdmin o un programma equivalente, perchè WordPress non permette di cambiare il prefisso direttamente.

Quindi queste tabelle:

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

Dovrebbero diventare

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

State pensando che abbiamo finito, giusto? Invece non ancora. WordPress include alcuni valori nel suo database che usano il prefisso delle tabelle. Perchè il blog torni a funzionare, dobbiamo cambiare questi valori.

Dalla tabella `wp_options`² dobbiamo cambiare il valore di un record nel campo `option_name`, da `wp_user_roles` a `4i32aK_user_roles`³.

Ora dovete sostituire due⁴ altri valori nella tabella `wp_usermeta`.

I valori `wp_autosave_draft_ids` e `wp_user_level` per il campo `meta_key` devono essere cambiati per rispecchiare il nuovo prefisso: `4i32aK_autosave_draft_ids` e `4i32aK_user_level`.

Finito! **Ma BlogSecurity ha reso questo processo ancora più semplice con il plugin [WP Prefix Table Changer](#)**. Assicuratevi di aver eseguito il backup del blog e del database prima di eseguire questo plugin. E' in fase Alfa e potrebbe avere dei bug.

¹ Query di esempio: `RENAME TABLE wp_categories TO 4i32a_categories`

² Usiamo il prefisso di default per evitare confusione

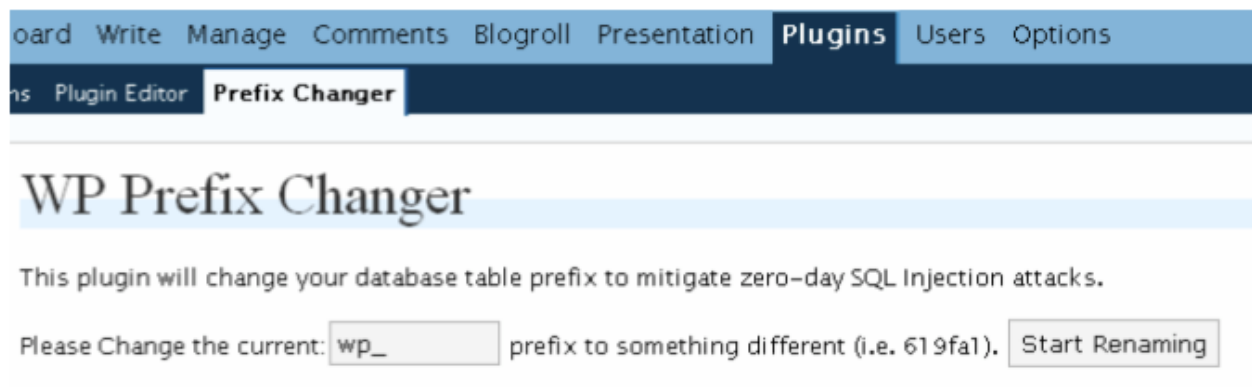
³ `UPDATE 4i32a_options SET option_name='4i32a_user_roles' WHERE option_name='wp_user_roles' LIMIT 1`

⁴ **Nota:** potrebbe accadere che questi campi non esistano, perchè vengono creati nel momento in cui servono in modo automatico da WordPress. In questo caso, quando essi saranno creati vi verrà assegnato il prefisso corrente, aggiornato.

Il plugin WP Prefix Table Changer

Abbiamo creato un plugin chiamato [WP Prefix Table Changer](#), che automatizza il procedimento di rinominare le tabelle di WordPress ed il cambiamento dei valori dei campi citati.

Dopo aver scaricato il plugin da BlogSecurity.net, dovete estrarre i files nella cartella dei plugin di WordPress, che dovrebbe essere **WordPress/wp-content/plugins**. Dopodichè dovete accedere all'interfaccia di amministrazione ed abilitare il plugin WP Prefix Table Changer. Dopo l'attivazione, un nuovo sottomenu comparirà nella pagina dei plugin, chiamato Prefix Changer. Cliccatelo e vedrete la seguente pagina



Come potete vedere questo blog utilizza il prefisso di default wp_. Cambiatelo in qualcosa di casuale, senza significato come il nostro precedente esempio 4i32aK_. Fatto questo, cliccata il pulsante 'Start Renaming'. Il plugin inizia a rinominare tutti i nomi dei prefissi delle tabelle da wp_ alla nuova stringa indicata. Notare che vengono anche cambiati i prefissi delle tabelle generate da plugin di terze parti, poichè anche loro ora necessitano questo trattamento per poter funzionare.

L'ultimo passo eseguito da questo plugin è cambiare il prefisso all'interno del file **WP-CONFIG.PHP**.

Al termine dell'operazione vi verrà visualizzato un messaggio che indica se il procedimento è andato a buon fine. Se la trasformazione è andata bene, i permessi del file WP-CONFIG.PHP saranno modificati e resi di sola lettura (**644**) per motivi di sicurezza. Se questa operazione fallisce, è probabile che il file sia già read-only, perciò dovete fare il cambiamento a mano.

Preparare il blog

Ora il vostro blog è installato e avete implementato diversi ottimi accorgimenti per incrementare la sicurezza, ben fatto! Ora cambieremo il nome dell'utente amministratore di default 'admin' in qualcosa di diverso, per renderlo più sicuro. Creeremo inoltre un account utente WordPress con permessi limitati per eseguire le attività quotidiane. Infine, installeremo il plugin Role Manager per fornire un controllo granulare su quello che è consentito fare ai moderatori ed agli autori registrati nel blog.

Cambiare lo username di Admin

Dovreste rinominare l'account di amministrazione di default da **admin** in qualcosa di più complicato da intuire, dato che tutte le versioni di WordPress disponibili sono vulnerabili alla [User Emulation](#). Cambiare il nome dell'account di amministratore di default aiuterà a difendersi dagli attacchi di brute force.

*Nota: è importante pensare che l'hacker sappia il vostro nome utente, ed inserire una password sicura anche se avete cambiato il nome dell'utente amministratore. **Non ci stanchiamo di ripeterlo!***

Connettiamoci al database MySQL con il nostro account 'wpuser' e cambiamo il nome di default dell'amministratore con la seguente query:

```
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
```

L'account 'admin' è stato cambiato in 'adm'. Naturalmente nella realtà è meglio scegliere un nome un po' meno ovvio.

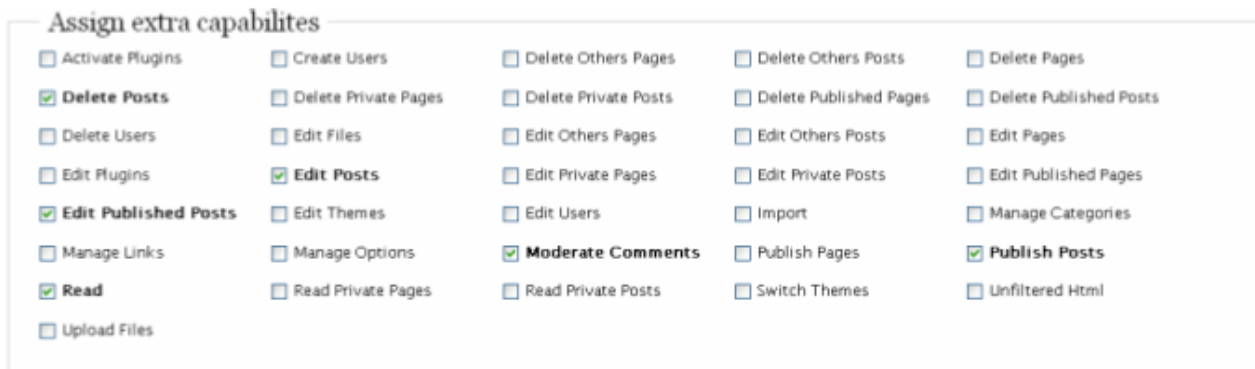
Creare un nuovo utente con permessi limitati

Prima di iniziare questo paragrafo dovrete scaricare il plugin [Role Manager](#), creato da [im-web-gefunden](#). Questo plugin permette di impostare permessi utente granulari per ciascun utente WordPress. Dopo aver attivato il plugin, create un nuovo account utente WP. Sugeriamo di creare il primo account per voi stessi. Rimuovete tutti i permessi dell'account preimpostati e selezionate solo i permessi che avrete bisogno nelle vostre attività giornaliere (ad esempio scrivere post, moderare i commenti, ecc). Assicuratevi che solo l'utente amministratore possa eseguire operazioni privilegiate come abilitare/disabilitare plugin, effettuare l'upload di files, modificare le opzioni, cambiare il tema, effettuare importazioni. Questa operazione può richiedere qualche giorno per essere perfezionata.

Nota: meno operazioni l'account utente è abilitato ad eseguire, più alto sarà il livello di sicurezza. Un account di tipo Contributor è generalmente un buon livello di account per operare.

Il ruolo di Contributor potrebbe non avere abbastanza privilegi di default, ma possiamo aggiungerli usando il plugin Role Manager.

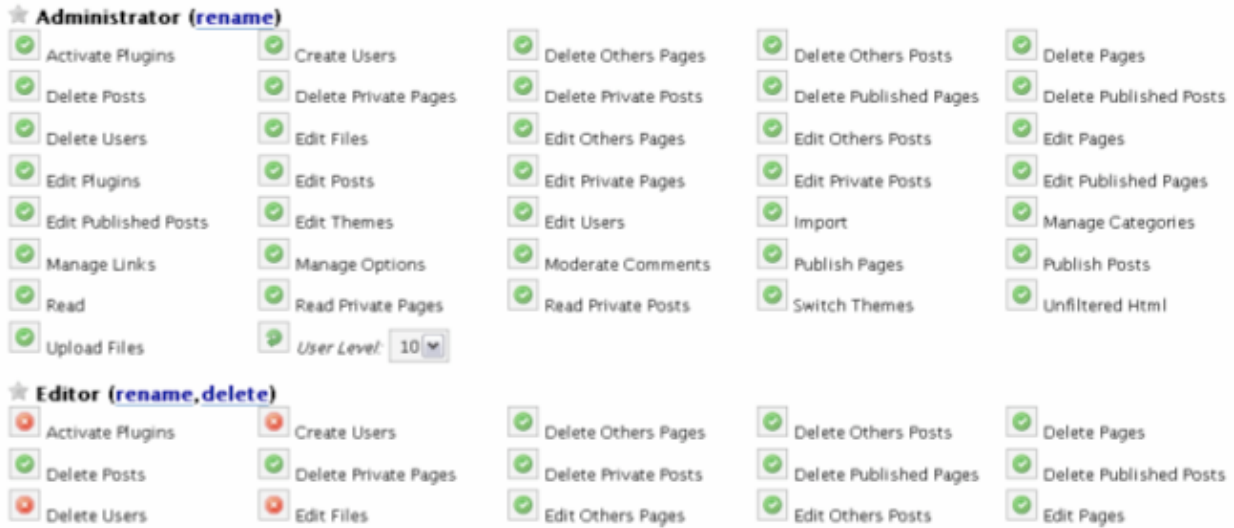
Come standard suggeriamo di dare ai nuovi utenti il livello Contributor, tuttavia questo plugin permette di avere una maggiore flessibilità, come si può vedere nell'immagine



Si veda: [Informazioni sul plugin Role Manager](#)

Se avete più utenti, sarebbe una buona idea pensare a quello che realmente necessitano gli utenti e creare un ruolo utente personalizzato.

Quando create gli utenti, attenzione ad assegnare ad utenti di cui non vi potete fidare completamente ruoli come “upload files” (caricamento file), “general plugin access” (accesso ai plugin), “edit files/pages/posts” (modifica pagine/post/file), “import” (importa), “unfiltered html” (html non filtrato), poichè questi ruoli danno agli utenti molti privilegi che potrebbero rivelarsi controproducenti.



Cambiare i ruoli utente con [Role Manager](#).

Rendere più sicura l'installazione WordPress⁵

Questo paragrafo parla di come proteggere l'area amministrativa dagli accessi non autorizzati. Il processo è semplice per blog con un singolo utente, ma può essere lungo e tedioso per blog multiutente. Dovete decidere se volete o meno sobbarcarvi questo peso per rendere sicuro il vostro blog, ma è un'operazione caldamente consigliata.

Limitare l'accesso alle cartelle wp-content e wp-includes

Possiamo limitare l'accesso a queste cartelle, in modo che sia negato l'accesso ad ogni risorsa contenuta in esse, ad eccezione di immagini, CSS e alcuni file JavaScript.

Dovete inserire il seguente codice nel file **.HTACCESS** delle cartelle **WP-CONTENT & WP-INCLUDES**:

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

Nota: alcuni plugin e template possono richiedere di dover specificare il permesso di accedere ad alcuni file PHP.

Limitare l'accesso a wp-admin

Bloccare tutti ad eccezione del vostro indirizzo IP

Se il vostro blog è a singolo utente, conviene restringere l'accesso alla cartella **WP-ADMIN** basandosi sull'indirizzo IP. Assicuratevi di avere un'indirizzo IP statico (che non cambia mai) prima di eseguire questa operazione. Il file **.HTACCESS** nella cartella **WP-ADMIN** dovrebbe essere:

```
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
```

Salvate il file e provate ad accedere alla cartella wp-admin attraverso un proxy web; l'accesso dovrebbe essere bloccato se tutto è andato a buon fine. Dopodichè provate ad accedere direttamente dal vostro indirizzo IP (a.b.c.d).

Se tutto è andato bene come dovrebbe **WP-ADMIN** sarà raggiungibile solo dal vostro indirizzo IP.

⁵ L'articolo originale si trova all'indirizzo <http://blogsecurity.net/WordPress/article-210607>

Richiedere l'uso della password http: .htpasswd

Sicuramente l'opzione preferita per la sicurezza è usare la protezione basata su password. Questo significa che potrete accedere alla cartella WP-ADMIN attraverso ogni indirizzo IP, ma avrete comunque un ulteriore livello di sicurezza che possa vietare l'accesso agli utenti che non sono autorizzati.

Il file .htaccess

Il file .HTACCESS nella cartella WP-ADMIN dovrebbe essere

```
#questo file dovrebbe risiedere all'esterno della web root.  
AuthUserFile /srv/www/user1/.htpasswd  
AuthType Basic  
AuthName "Blog"  
require user youruser #rendere questo nome utente difficile da indovinare aiuta a rendere più sicuro il  
#server da attacchi di forza bruta.
```

Il file .htpasswd

Questo file⁶ dovrebbe, come già menzionato, essere posizionato in una cartella che è fuori dalla cartella del web server, idealmente dovrebbe essere inserito nella cartella padre della directory di base dove vengono serviti i file HTML.

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

Il file .htpasswd deve essere creato nella cartella per cui è pensato. Assicuratevi che il file sia inserito nella stessa cartella indicata da AuthUserFile in 'wp-admin/.htaccess'.

Ora provate a vedere se tutto funziona a dovere. Quando provate ad accedere al backend di amministrazione del blog, dovrebbe essere richiesto di inserire login e password, prima ancora di giungere al login di WordPress. Se non è così, ricreate il file .htpasswd e controllate che i percorsi siano impostati nel modo corretto.

⁶ Maggiori informazioni su questo file si possono trovare qui:
http://httpd.apache.org/docs/1.3/mod/mod_auth.html

Spam

Una delle grandi idee che stanno dietro il concetto di blogging è la possibilità di permettere il feedback dei lettori, sotto forma di commenti. Sfortunatamente, 2 commenti su 3 sono spam. Per questo sono state messe a disposizione diverse soluzioni, tuttavia alcuni di questi metodi possono essere fastidiosi per l'utente e limitare, a lungo andare, il feedback.

Captcha – Anche conosciute come “quelle immagini difficili da leggere”. Le soluzioni basate su captcha funzionano abbastanza bene per bloccare lo spam, ma solo quando sono implementate nel modo corrente (come abbiamo imparato dal progetto Captcha bugs of the month di [Mustlive's](#)). Lo svantaggio dei captcha è che sono fastidiosi e rappresentano un ostacolo per la maggioranza dei lettori. Chi ha voglia di indossare i propri occhiali e sforzarsi di capire quello che c'è scritto nell'immagine ogni volta che vuole scrivere un commento?

Autenticazione – Prima di poter lasciare un commento, un utente deve registrarsi. Sfortunatamente, esistono programmi automatizzati che riescono ad effettuare questa operazione, per cui l'autenticazione è associata ad un controllo di tipo captcha all'atto della registrazione, in modo che quando abbiamo la conferma che l'utente è umano, non dobbiamo più chiederglielo. Purtroppo però ora l'utente deve eseguire il login ogni volta che vuole lasciare un commento, il che significa dover ricordare una nuova password. [OpenID](#) potrebbe essere un passo avanti in quest'area.

Blacklist – Molti software permettono di usare le blacklist. Questo significa che è possibile bloccare i commenti basandosi su “cattive” parole. Non so se siete mai stati allo stadio ad assistere ad una partita tra Inter e Juventus, ma vi posso assicurare che esistono molti modi di dire la stessa cosa. Le blacklist sono ottime quando implementate in cooperazione con altri metodi, ma sicuramente non sono il massimo come unico metodo di protezione.

JavaScript – Lo SpamBam di BlogSecurity usa uno script lato client eseguito dal vostro browser per assicurarsi che l'utente sta usando un browser valido. Questo è un metodo molto efficace per prevenire lo spam nei commenti, dato che molti spammer usando programmi automatici che non supportano JavaScript. Questo è un sistema molto buono, un approccio davvero interessante ma è richiesto un certo impegno per implementare questa soluzione.

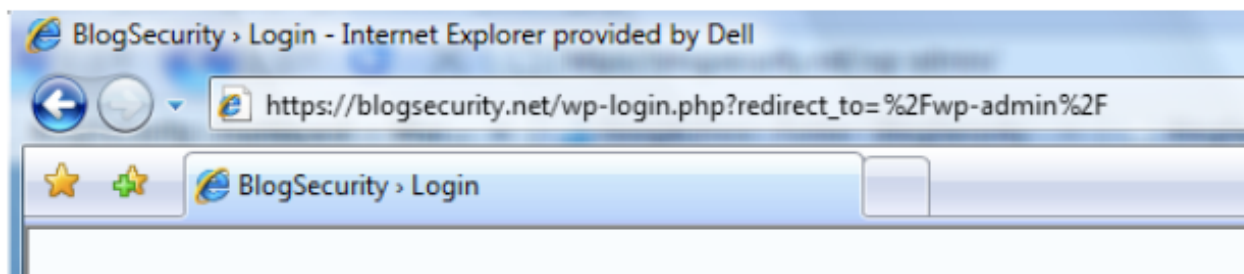
Controlli intelligenti – Questi sono sistemi antispam come Akismet. Usano una serie di controlli e catene di blacklist per controllare se il commento è spam o meno. Ottimi per uso generale, tuttavia per funzionare spediscono tutti i commenti ad un server di terze parti, il che può causare un eccessivo traffico aggiuntivo.

Ci sono un gran numero di plugin anti-spam per WordPress disponibili, tuttavia di solito suggeriamo di usare uno di questi due plugin:

- [Akismet](#) – plugin antispam di Automattic (richiede la registrazione gratuita per ottenere una chiave API per farlo funzionare)
- [SpamBam](#) – plugin antispam di BlogSecurity.

Crittazione del blog

Quando effettuate il login al vostro blog, è importante assicurarvi che questa operazione sia fatta usando 'HTTPS'. Questo evita che i cracker catturino username e password che altrimenti sarebbero inviati attraverso internet in chiaro, non criptati.



Innanzitutto controllate che il vostro host supporti 'HTTPS', puntando il browser all'indirizzo `https://yourblog/`

L'installazione di 'HTTPS' è al di fuori dello scopo di questo documento. Ricordatevi che Google è vostro amico. Andiamo avanti.

Dato che avete già SSL (HTTPS) installato sul vostro blog, tutto ciò che dovete fare è installare un plugin di WordPress che reindiriga verso 'HTTPS' gli accessi a 'wp-login.php' e tutte le altre aree protette del vostro blog.

L'avete indovinato? Sì, BlogSecurity ha pensato anche questa volta ad un plugin per voi. Scaricate una copia di 'bs-wp-encrypt' [qui](#).

Rinominate il file 'bs-wp-https_php.txt' in 'bs-wp-https.php' ed inseritelo nella cartella 'wp-content/plugins/'. Ora andate nel pannello di amministrazione ed abilitatelo. Pronti! Il vostro blog dovrebbe ora riconoscere le aree protette e ridirigerle sul protocollo HTTPS.

Plugin chiave

BlogSecurity fornisce altri interessanti plugin per rendere la vita difficoltosa a possibili utenti malintenzionati. Sugeriamo di usare questi plugin nel vostro blog WordPress.

Rimuovere la versione di WordPress

Il plugin WordPress Noverion (bs-wp-noverion) di BlogSecurity previene la visualizzazione del numero di versione di WordPress. Un altro semplice e molto utile plugin per la sicurezza.

Molti cracker e programmi automatizzati cercano di determinare la versione del software WordPress prima di provare ad eseguire qualche exploit conosciuto. Rimuovere ogni informazione sulla versione del blog engine potrebbe scoraggiare qualche attacker e certamente mitigherà le chance di essere vittima di virus o worm che si basano sulla versione del software per installarsi.

Nota: Questo plugin può generare conflitti con altri plugin che usano la versione di WordPress come informazione per funzionare.

Il plugin 'bs-wp-noerrors' può essere scaricato [qui](#).

Disabilitare la visualizzazione degli errori di WordPress

Questo plugin è stato deprecato a partire da WordPress 2.3.2. Dalla versione 2.3.2, WordPress ha i messaggi di errore disabilitati di default. Questo plugin potrebbe essere utile per versioni vecchie di WP.

Di default WordPress ha la visualizzazione dei messaggi di errore attivata di default:

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

Nota: gli errori del database verranno visualizzati agli utenti, anche se gli errori PHP sono stati disabilitati. Questo plugin disabilita i messaggi di errore del database di WordPress, per evitare che informazioni importanti e sensibili possano essere rivelate, come ad esempio il **prefisso delle tabelle** WP.

Il plugin 'bs-wp-noerrors' è disponibile [qui](#).

Altre opzioni di sicurezza

Questa lista di plugin/servizi può essere di grande aiuto nel mettere sotto chiave la sicurezza del blog.

WPISD – Intercettare le intrusioni

BlogSecurity ha eseguito il porting di PHPIDS (Intrusion Detection System) in WordPress. PHPIDS permette di intercettare diversi tentativi di intrusione. Noi usiamo questo servizio per bloccare attacchi pericolosi. Ogni intrusione è registrata nel database, per cui potete tenere traccia di ogni movimento e prendere le precauzioni/decisioni necessarie. Potete impostare di ricevere una e-mail se è in atto un'intrusione che supera un certo peso (ogni intrusione ha un preciso livello di minaccia).

Potete anche bloccare l'IP del cracker per un certo numero di giorni se il pericolo è alto, comunque WPIDS tenterà di correggere eventuali input maligni.

[Scaricate questo plugin](#) dal sito ufficiale PHPIDS.

Nota: perchè questo plugin funzioni, dovrete disporre di PHP 5.1.6 o superiore. Una nuova versione di WPIDS sarà rilasciata a breve, includendo una nuova feature di BlogSecurity, WP-Lockdown, per cui tornate a trovarci.

WordPress Plugin Tracker

Se avete appena installato il blog, software e plugins dovrebbero essere aggiornati all'ultima versione. Dovreste installare il plugin [WordPress Plugin Tracker](#) per sapere quando viene rilasciata una nuova versione dei plugin. Dopo aver installato ed attivato questo plugin, accedete alla pagina dedicata per sapere se avete aggiornati i plugin all'ultima versione. Dovreste vedere una schermata di questo tipo:

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
Another Wordpress Meta Plugin	2.0.3	2.0.3	Versions are matching, You have latest v
Akismet	2.0.2	2.0.2	Versions are matching, You have latest v
Bad Behavior	2.0.10	2.0.10	Versions are matching, You have latest v
http://BL WordPress Plugin	1.4	1.4	Versions are matching, You have latest v

Questa è una pagina di esempio del plugin WordPress Plugin Tracker.

Se alcuni plugin non sono aggiornati, vi verrà notificato e cliccando il titolo del plugin sulla sinistra sarete portati alla home page del plugin dove potrete trovare l'aggiornamento. In questo modo sarà più facile avere il blog aggiornato.

Nota: WordPress 2.5 ha implementato un sistema di update automatico dei plugin, rendendo inutile il plugin appena descritto.

WordPress Online Security Scanner

BlogSecurity ha scritto un tool che controlla la sicurezza di WordPress, effettuando un check delle comuni vulnerabilità. E' molto adatto per individuare i plugin, controllare vulnerabilità di scripting cross-site e molto altro.

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-NOODERIVS LICENSE

WP-Scanner è un servizio online gratuito ed è stato testato da oltre 5000 blog (ormai abbiamo perso il conto). Per maggiori informazioni cliccare [qui](#).

Conclusioni

Siamo giunti alla fine del nostro whitepaper. Speriamo che vi sia stato utile, e che abbiate successo nell'implementare queste regole di sicurezza. E' sempre un piacere ricevere i vostri commenti e feedback, per cui non abbiate esitazione a [contattarci](#).

Questo documento è stato tradotto in italiano da Copes Flavio <http://copesflavio.com>