Come aumentare la sicurezza di WordPress

Recentemente ho terminato di sviluppare un progetto per il quale ho utilizzato wordpress. Si tratta di un gestionale aziendale con moltissime funzionalità ed all'interno del quale vengono conservate informazioni di una certa confidenzialità; non numeri di carte di credito o codici di lancio di testate nucleari ma comunque dati che richiedono un'accresciuta attenzione.

Così ho fatto una ricerca approfondita su quali possono essere gli accorgimenti da mettere in atto per migliorare la sicurezza di wordpress ed ho pensato di condividere con voi quello che ho potuto raccogliere. Ho trovato cose che già sapevo e mettevo in atto, cose per le quali mi sono detto "... però, non ci avevo pensato.." ed altre che francamente trovo discutibili. Ma vediamole insieme iniziando dalle cose facili.

1. Qualità dell'hosting

Questa é una cosa che incredibilmente non ho trovato nelle mie ricerche, forse perché la danno tutti per scontata, ma non io. **La prima linea di difesa da attacchi informatici é il server stesso.** Se ti serve sicurezza, di certo non la troverai in servizi che offrono hosting "professionale"(?!) per 2 euro l'anno. Da quando ho scritto la serie di tutorial sui pagamenti online, il 100% degli utenti che hanno avuto problemi utilizzavano servizi di hosting "popolari".
Per fare lavori seri, ci vogliono strumenti seri. Cosa pensereste se andaste in ospedale per farvi operare e sul tavolo del chirurgo ci fosse una motosega, degli scalpelli ed un tagliere per il salame?

2. Permessi a livello di server

Permessi di scrittura
Un primo step per elevare la sicurezza é verificare i permessi di scrittura di files e cartelle che devono essere:

- 755 per le cartelle
- 644 per i files

Questa dovrebbe essere un'impostazione di default del server, in caso contrario modifica i permessi.

Impedire il browsing
Un altro step é impedire la navigazione nelle cartelle senza index, aggiungendo all'inizio del file *.htaccess* questa riga.

Ecco due operazioni tanto semplici quanto importanti, non dimenticarle.

3. Depistaggio

Quando scriviamo un'applicazione, una discreta parte di sicurezza é data dal fatto che il malintenzionato non sa nulla del codice, dell'organizzazione del database, delle procedure utilizzate, eccetera.
Ovviamente **non possiamo contare su questa barriera quando utilizziamo del software open source**; anzi il fatto che il funzionamento sia trasparente, fornisce una moltitudine di informazioni al malintenzionato. Per questo motivo dobbiamo evitare, per quanto possibile, che le forze del male possano trarre vantaggio dalla conoscenza di alcuni meccanismi.

Eliminare l'utente admin
Se installiamo wordpress così come ci viene proposto, non sarà un segreto per nessuno che:

- Nella tabella *wp_users* c'é un utente con diritti amministrativi
- Questo utente ha ID 1
- Questo utente - se in fase di installazione non avete scelto un nome diverso - si chiama *admin*

Così diamo un po' troppo vantaggio alle forze del male, non credi?
Crea un nuovo utente con privilegi di amministratore ed un nome diverso e rimuovi l'utente *admin*.

Quello che dobbiamo evitare é che vi sia un utente con diritti amministrativi che si chiami admin e/o abbia id = 1, altrimenti é troppo facile

Modificare il prefisso delle tabelle del database
Se lasciamo che wordpress utilizzi il prefisso di default (wp_), tutto il mondo saprà il nome esatto delle tabelle contenute nel database. Questo evidentemente é una facilitazione per i malintenzionati che volessero tentare delle query injection.
Modificando il prefisso delle tabelle faremo un altro passo verso il miglioramento della sicurezza.
Questa operazione va fatta in fase di installazione modificando la costante nel file *wp-config.php*.
Per gli utenti più esperti é possibile fare questa modifica anche in un secondo tempo, ma bisogna sapere quello che si sta facendo!

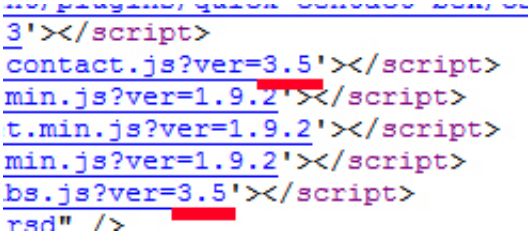
Spostare il file wp-config.php
Dalla versione 2.6, é possibile spostare il file *wp-config.php* in una cartella superiore rispetto a quella dove risiede di default, e wordpress troverà comunque il file.
Questa procedura l'ho trovata citata più volte nelle mie ricerche ed ho sempre avuto dei seri dubbi sul come questo potesse migliorare la sicurezza.
Ho infine trovato, nella documentazione ufficiale di wordpress, che al riguardo c'é una

controversia. C'é chi addirittura ritiene che questo possa peggiorare la sicurezza.
Secondo me non cambia nulla.

Nascondere la versione di wordpress
Per il malintenzionato, conoscere la versione di wordpress, può essere un piccolo vantaggio.
Nell'header normalmente la versione viene mostrata con questo metatag

Nel corso delle mie ricerche ho visto più volte citata la procedura per rimuovere questo metatag dall'header, che é la seguente:

Trovo questa procedura, messa in atto così com'é, semplicemente ridicola! Gli utenti più esperti sanno certamente che a qualsiasi script o foglio di stile per il quale, in fase di queuing non é stata definita una versione, verrà assegnata automaticamente la versione corrente di wordpress.



Dunque se ci si limita a questo, non serve a nulla. **Bisogna anche rimuovere l'aggiunta automatica della versione che applica wordpress,** ma questo é stato bellamente ignorato da tutti.
Aggiungi anche il seguente codice al file *functions.php*

Infine i nostri sforzi di nascondere la versione saranno stati vani se non cancelliamo i file *leggimi.txt, leggimi.html, licenza.txt, licenza.html* dalla root di wordpress. E ricorda di controllare dopo ogni aggiornamento; é probabile che questi files vengano riscritti.

4. Sicurezza password

Modificare le chiavi di salatura
Anche se dovrebbe essere scontato, non dimenticare di modificare le chiavi di salatura contenute nel file *wp-config.php*. La via migliore é quella di generarle tramite le apposite API di wordpress, semplicemente cliccando su questo link.

Password complesse
Anche qui siamo nel campo dell'ovvio. Se sei l'unico utente del sito non c'é problema, dipende

solo da te. Ma se il tuo blog ha diversi utenti (come nel caso di yiw), chi può garantirti che gli altri utenti abbiano delle password sufficientemente robuste? Io ad esempio come password uso *maurizio*, se Nando lo sapesse mi ucciderebbe.
Per risolvere questo problema esistono dei plugin come questo, che obbligano gli utenti ad avere delle password complesse.

Prevenire gli attacchi "brute force"
Per bloccare questo tipo di attacchi é sufficiente installare uno dei numerosi plugin, ad esempio questo. È possibile impostare dopo quanti tentativi falliti di login dal medesimo IP far intervenire un blocco di una durata definibile. È un piccolo accorgimento che vale la pena di mettere in atto.

Proteggere la cartella wp-admin
Possiamo infine sottomettere l'intera cartella *wp-admin* all'autenticazione a livello di server (tramite file *.htaccess* come ho illustrato in questo articolo).
Chiaramente diventa un po' scomodo in quanto bisognerà compilare due login per accedere al backend amministrativo, ma é stato dimostrato che questa tecnica aumenta notevolmente la sicurezza.

Prevenire gli attacchi MITM
E' possibile prevenire gli attacchi MITM (Man in the middle) attivando il protocollo cifrato sul login e/o sull'area amministrativa aggiungendo a *wp-config.php* la seguente riga:

Naturalmente in questo caso il server deve supportate il protocollo ssl (e qui ritorniamo al punto 1 di questo articolo) e bisogna disporre di un certificato.

Conclusione

In questo articolo abbiamo visto una serie di stratagemmi per rendere perlomeno la vita difficile alle forze del male. Possiamo aggiungere che mantenere aggiornato wordpress **e i plugin** é un ulteriore aiuto in questa eterna battaglia. A proposito di plugin, se ne hai le competenze, ti invito a dare sempre un'occhiata al codice. **Non tutti i plugin sono scritti a regola d'arte e in alcuni casi introducono delle vulnerabilità**.
Per concludere ti segnalo un plugin, Better WP Secutity, che provvede a svolgere diverse delle cose che abbiamo visto ed implementa anche funzioni di monitoraggio ed altro. L'unico problema: usa tante risorse.
E tu? Hai altri consigli da dare su questo argomento?