**Hardening WordPress**

(or, How Not To Get Hacked, Then What to Do When You Are)

# Resources

- Codex <http://codex.wordpress.org/Hardening_WordPress>
  <http://codex.wordpress.org/Brute_Force_Attacks>
- Blog.Sucuri.net <http://blog.sucuri.net/category/wordpress-security>
- WPSecure <http://wpsecure.net/basics/>
- WPVulnDB.com <http://WPVulnDB.com/>
- healthy dose of paranoia

# Preparing For War

Sun Tzu, “The opportunity to secure ourselves against defeat lies in our own hands, but the opportunity of defeating the enemy is provided by the enemy himself.”

## Know Thine Enemy

### What?

- bulk password (dictionary attack)
  - POST /wp-login.php
  - POST /xmlrpc.php <https://wordpress.org/support/topic/brute-forcing-via-xmlrpc>
  - <http://blog.sucuri.net/2014/07/new-brute-force-attacks-exploiting-xmlrpc-in-wordpress.html>

- vulnerable plugin
  - **All-in-One SEO** <http://www.tripwire.com/state-of-security/top-security-stories/all-in-one-seo-pack-wordpress-plugin-vulnerabilities/>
  - Contact Form 7 <http://blog.sucuri.net/2014/08/database-takeover-in-custom-contact-forms.html>
  - Custom Contact Forms <http://arstechnica.com/security/2014/08/critical-wordpress-plugin-bug-affects-hundreds-of-thousands-of-sites/>
  - MailPoet <http://arstechnica.com/security/2014/07/mass-exploit-of-wordpress-plugin-backdoors-sites-running-joomla-magento-too/>
  - MainWP Child (90,000)
  - **WordPress SEO** by Yoast <http://www.wordfence.com/blog/2015/03/vulnerability-in-wordpress-seo-by-yoast-upgrade-immediately/>
  - **WP eCommerce**
  - **WP-Slimstat (1.3 million)**
  - WPTouch (20 million) <http://www.zdnet.com/wordpress-plugin-vulns-affect-over-20-million-downloads-7000031703/>

- vulnerable theme
  - RevSlider (100k sites)
  - TimThumb <http://arstechnica.com/security/2014/06/running-wordpress-got-webshot-enabled-turn-it-off-or-youre-toast/>

form spambot
comment spam
contact form spam

DDOS, SQL injection, XSS, etc.

### When?
<http://en.wikipedia.org/wiki/WordPress#Vulnerabilities>
*May 2003 WP debuts*
2007-2008 WP core vulnerabilities (backdoor)
*Dec 2008 WP v2.7 adds one-click update feature*
2013 vulnerable plugins, targeting Top 50
*2013 WP v3.7 adds auto-update for patches (X.X.n, e.g. 3.9.0 to 3.9.1)*
2014 brute force attacks, targeting wp-login and XML-RPC
*2014 Auttomatic acquires BruteProtect*

### Who?
script kiddies
hacker mafia -> mafia hackers
state hackers

### Why?
Willie Sutton, "because that's where the money is."

WordPress "has grown to be the largest self-hosted blogging tool in the world, used on millions of sites and seen by tens of millions of people every day."

"WordPress was used by more than 22.0% of the top 10 million websites as of August 2013. WordPress is the most popular blogging system, at more than 60 million websites."

commonality = predictability
users infrequently update
botnets make it easy, low-risk, automated

conscription (for later use e.g. DDOS, for resale)
content manipulation (spam links, IFRAME injection e.g. fake AV scams, click selling)
steal user profiles (for spam, identity theft)
= to make $

## Principles of War

### Basic Training
- for more, see Michele Butcher’s session

1. Acquire software only from trusted sources (WP core, plugins, theme)
2. Minimize vulnerabilities by avoiding & removing unnecessary plugins
3. Stay up to date (WP core, plugins, theme)
4. Regular backups

5. Strong passwords (WP admin, MySQL, FTP)
6. Rotate keys & salts <https://api.wordpress.org/secret-key/1.1/salt>
7. No ‘admin’ account
8. Different DB prefix (not wp_*)
9. Secure access (SSL, SFTP)
10. Consider security plugins (but watch for conflicts & overhead)

### Attack in Order
SecuritySage presentation: <http://www.anticlue.net>
hardening checklist: <http://www.anticlue.net/SecuritySage/HardeningWorkPressChecklist.xlsx>

1. Recon
   Gather Offsite Detail
   Harvest Onsite Details
2. Scan
   Scan Vulnerabilities
   Map Weaknesses
3. Exploit
   Target Vulnerabilities
   Increase Privileges
4. Leverage
   Exploit Access
   Deposit Payload
   Profit!

### Defense (Vulnerability) In Depth
Theme
Plugins
Core WordPress
MySQL DB
Web Server Daemon
Server Firewall
Server OS
Network Firewall
Network
DNS

Primacy of Defense: The lower in the stack you can intercept, the better

### Security vs. Convenience
Auto-updates
Inline editing
Easy access to dashboard

### Border Security
- Database restrictions

Avoid multi-site unless strongly justified (shared database access)

Limit active user to SELECT, INSERT, UPDATE and DELETE (ALTER needed for major point releases

- Access Control

Basic authentication on /wp-admin
<http://codex.wordpress.org/Brute_Force_Attacks#Password_Protect_wp-login.php>

Limit logins by IP
<http://codex.wordpress.org/Brute_Force_Attacks#Limit_Access_to_wp-admin_by_IP>

.htaccess rules

```
  # Stop spam attack logins and comments
  <IfModule mod_rewrite.c>
RewriteEngine On
RewriteCond %{REQUEST_METHOD} POST
RewriteCond %{REQUEST_URI} .(wp-comments-post|wp-login)\.php*
RewriteCond %{HTTP_REFERER} !.*yourwebsitehere.com.* [OR]
RewriteCond %{HTTP_USER_AGENT} ^$
RewriteRule (.*) http://%{REMOTE_ADDR}/$ [R=301,L]
  </ifModule>

  # Block WordPress xmlrpc.php requests
  <Files xmlrpc.php>
  order deny,allow
  deny from all
  </Files>

  # Block WordPress wp-config.php requests
  <Files wp-config.php>
  order deny,allow
  deny from all
  </Files>

  # disable HTTP Track Attack (XST)
  RewriteEngine On
  RewriteCond %{REQUEST_METHOD} ^TRACE
  RewriteRule .* - [F]
```

Plugin enforcement (iThemes Security, Wordfence)

Disable file editing in wp-config.php

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

### Counter Espionage

*Change what is expected, hide what is knowable.*

Block robot browsing

robots.txt:

```
User-agent: *
Disallow: /wp-content/plugins/
Disallow: /wp-admin/
Disallow: /wp-content/
Disallow: /wp-includes/
Disallow: /wp-
Disallow: /xmlrpc.php
```

Change DB table prefix (not wp_*)

Disable WP version display (via theme functions.php or via plugin)

```
// remove version info from head and feeds
function complete_version_removal() {
   return '';
}
add_filter('the_generator', 'complete_version_removal');
```

Relocate wp-config.php (outside web root, can be one-level above index.php)

Relocate core WP files (McCreary multi-tenant method)
<http://jason.pureconcepts.net/2012/08/wordpress-multitenancy/>
<http://jason.pureconcepts.net/2013/04/updated-wordpress-multitenancy/>

1. Install WP into subdirectory (e.g. /core)
2. Follow Codex instructions for 'Giving WordPress Its Own Directory'
3. Copy wp-config.php to site root (/)
4. Edit subdirectory wp-config.php to include via * $_SERVER['DOCUMENT_ROOT']*
5. Move subdirectory to core path (e.g. /usr/local/wordpress/4.0)
6. Symlink subdirectory to new core path

```
ln -s /usr/local/wordpress/4.0 core
```

7. Site now loads index.php, which looks to /core/ through symlink, which references back to originating site's wp-config via *$_SERVER['DOCUMENT_ROOT']*

* ’Update' of core WordPress is now the same as 'replace symlink with pointer to different version'

```
rm core; ln -s /usr/local/wordpress/4.1 core
```

Read-lock everything outside wp-content/uploads

chmod -R 640 ll chmod -R ga-w (depends on server user/daemon scheme)

## Security Drills

vulnerability scan / penetration testing

brobot I itsoknoproblembro DDOS toolkit

Flunym0us <http://code.google.com/p/flunym0us/>

Kali <https://www.kali.org>

WPScan <http://wpscan.org/>

WP Security Scan <http://hackertarget.com/wordpress-security-scan/>

WordPress Auditor <https://github.com/0pc0deFR/Bulk_Tools/tree/master/WordPress%20Auditor>

WordPress Sploit framework <https://github.com/0pc0deFR/wordpress-sploit-framework>

prevention

- All In One WP Security & Firewall <https://wordpress.org/plugins/all-in-one-wp-security-and-firewall/>
- Better WP Security -> iThemes Security <https://wordpress.org/plugins/better-wp-security/>
- BruteProtect (cloud based login blocking) <https://wordpress.org/plugins/bruteprotect/>
- Sucuri <https://wordpress.org/plugins/sucuri-scanner/>

<http://www.wpbeginner.com/opinion/reasons-why-we-use-sucuri-to-improve-wordpress-security/>

- WordFence <https://wordpress.org/plugins/wordfence/>

detection

- Exploit Scanner <https://wordpress.org/plugins/exploit-scanner/>
- TAC (Theme Authenticity Checker) <https://wordpress.org/plugins/tac/>

TimThumb Vulnerability Scanner

comment spam prevention

- Akismet (built-in, annual fee/site) <https://wordpress.org/plugins/akismet/>
- Antispam Bee <https://wordpress.org/plugins/antispam-bee/>
- Bad Behavior <https://wordpress.org/plugins/bad-behavior/>
- Cookies for Comments <https://wordpress.org/plugins/cookies-for-comments/>
- Hashcash <https://wordpress.org/plugins/hashcash/>
- Stop Spam Comments <https://wordpress.org/plugins/stop-spam-comments/>

## Blessed are the sysadmins

Network-level security

DDOS mitigation at network edge

Firewall tuning to blunt specific threats

IDS rules

Server-level security

fail2ban: protect against bulk / DDOS attacks via IP blocking

mod_security: recipes to intercept attacks

    - suPHP: limit script execution by site owner (prevent neighbor attacks)

- Specialist hosting (e.g. WPEngine, DreamPress) and proxy/DNS (e.g. CloudFlare)

## responding to an attack

- check for telltales
    - recent modification dates
    - Base64 encoding (obfuscation)
- contact web host
- check & archive logs (learn the attack vector)
- block IP (plugin, web server module, firewall)
- quarantine 'bad' files for forensic review (outside web root)
- scan site files (e.g. WordFence)
- revert DB (prior to initial attack to eliminate backdoors)
- change passwords, salts (wp-config.php) <https://api.wordpress.org/secret-key/1.1/salt>