U4 networks

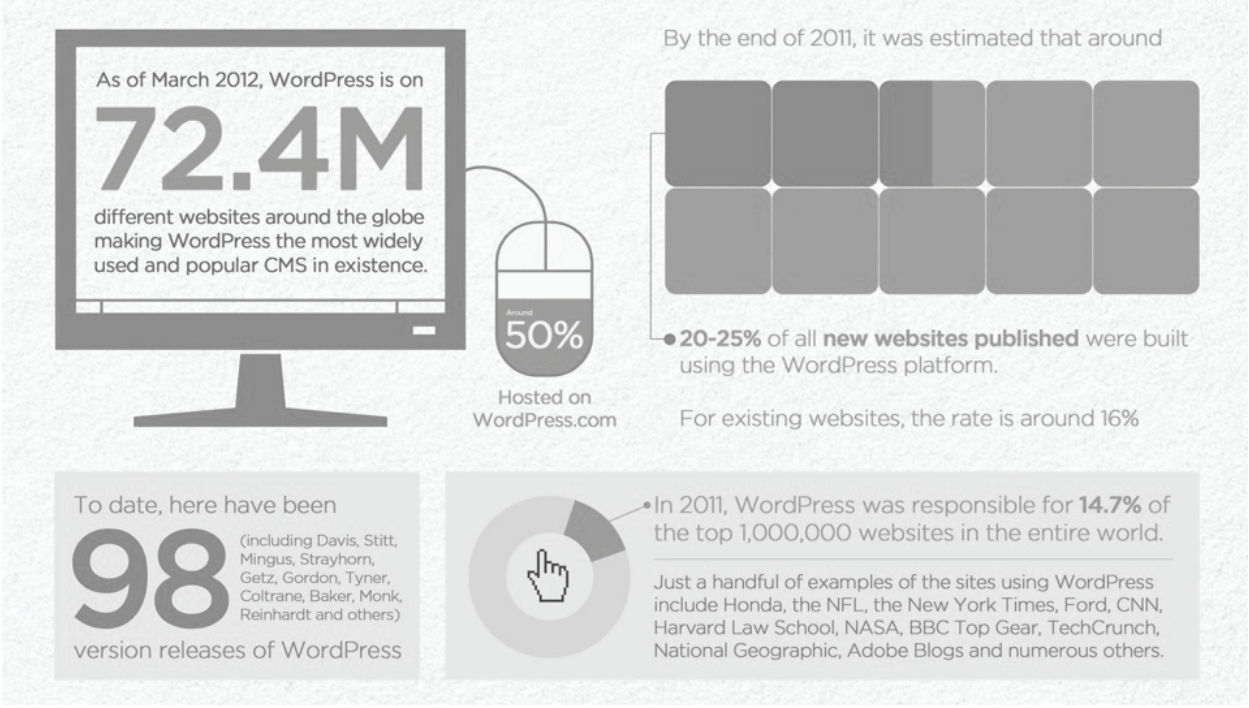
NOTE DE SYNTHESE

# SÉCURITÉ WORDPRESS

# Sommaire

## Introduction

**WordPress** est un système de gestion de contenu (généralement appelé 'Content Management System' ou '**CMS**'). Ecrit **en PHP**, il repose sur une base de données **MySQL**. Il est distribué gratuitement selon les termes de la licence **GNU GPL.**




Statistiques issues de l'étude Yoast d'avril 2012

Basé sur une architecture modulaire particulièrement souple, **WordPress s'appuie sur une librairie d'environ 20 000 plugins et de nombreux widgets**. Ce CMS permet également de procéder facilement à des extensions de fonctionnalités, permettant ainsi une grand nombre d'utilisations possibles : site vitrine ou 'corporate', sites e-commerce, weblogs, catalogues en ligne, mini-sites (events), extranets, etc.

**WordPress est aujourd'hui le CMS (Content Management System) le plus utilisé au monde** : 15% des sites Internet sont réalisés à l'aide de WordPress (plateforme + self-hosted) et 1 lancement de site sur 4 intègre désormais ce CMS (source : étude WordpressMu.org avril 2012). A l'échelle du Web (self-hosted), cela représenterait environ 8% des sites Internet.

L'appréhension de la sécurité des CMS open-source nécessite une approche différente des environnements développés de manière spécifique ou sur-mesure (dans le jargon : 'from scratch'). Cette distinction est induite par un certain nombre de contraintes inhérentes aux CMS 'libres' et par le mode de fonctionnement même des CMS. En effet, WordPress est un CMS sous licence GNU/GPL (libre et open-source), ce qui signifie que n'importe qui peut avoir accès aux codes sources et donc l'analyser pour essayer d'y déceler des failles potentielles. **WordPress étant le moteur CMS le plus répandu au monde, cela fait de lui une cible privilégiée.**

C'est pourquoi **des mises à jour sont constamment déployées pour corriger les failles détectées** (et ce avec une réactivité importante) mais personne n'est à l'abri d'une attaque exploitant une faille non-rectifiée. Pour conserver un niveau de sécurité maximale sur le moteur WordPress, il est donc nécessaire de mettre en place une veille active sur les différents correctifs mis à disposition par l'éditeur, et de les déployer rapidement. Ce type d'action n'est possible que dans le cadre d'un service de maintenance/suivi proposant un haut niveau de réactivité.

**La force** de WordPress et sa popularité proviennent de sa très **grande flexibilité**, de son amovibilité, de ses nombreuses fonctionnalités natives et de sa capacité à être profondément personnalisable. Ces atouts sont autant de faiblesses au niveau de la sécurité : plus le système est ouvert et malléable (et donc souple, évolutif et performant), plus nombreuses sont les failles potentielles.

Au-delà des considérationsentourant la sécurité du moteur, **il est essentiel de prendre en compte les aspects de sécurité liés à l'utilisation et l'exploitation de plugins** ou d'outils tiers et d'apporter un soin particulier à la vérification des failles potentielles que peuvent induire l'emploi de ces plugins, de manière intentionnelle ou non.

Ceci étant dit, WordPress dispose de bases élémentaires (notamment sa souplesse d'exploitation) qui lui permettent de tendre à satisfaire les principaux critères de sécurité généralement établis, tant que celle-ci n'est pas 'critique'. En effet, même s'il n'est pas le CMS réputé le plus sûr du marché, c'est souvent l'absence de dispositions nécessaires qui pose problème (et non l'emploi même du CMS). Pour preuve, **de nombreusesentreprises et institutions à visibilité mondiale s'appuient sur WordPress pour tout ou partie de leur sites et/ou applications internet…**

**Structures de premier plan utilisant WordPress**

### Médias

- CNN
- New York Times
- Le Monde
- La Presse
- CBS
- Time
- Owni
- L'actualité
- ONF
- TED
- The bangor Daily News
- The New York Observer
- Radio Okapi
- Radio Canada
- Wall Street Journal
- Wired
- TechCrunch
- GigaOm
- BoingBoing
- Forbes
- Métro
- Yahoo
- NBC Sport

...

### Entreprises

- Ebay
- Flickr
- Yahoo!
- Ford
- Sony
- GE
- Samsung
- UPS
- VW
- GM
- Moment Factory
- Nasa

...

### Éducation

- HEC Montréal
- CSDM
- Mc Gill
- MIT
- Harvard
- Cornell
- Berkeley

...

## I/ Identification des facteurs de risque

Dans le cadre de la mise en place d'une politique de sécurité, il faut prendre en compte la problématique et les besoins dans leur globalité, sans se focaliser uniquement sur le moteur (noyau) de WordPress. En effet, il est préalablement nécessaire d'**identifier les différents facteurs de risques** :

- La sécurité et la protection du moteur (noyau) WordPress.
- Le contrôle et la protection des plugins et widgets utilisés ou développés.
- La vérification des droits et accréditations des utilisateurs.
- La vérification et la protection de l'hébergement.
- La vérification des droits d'accès au FTP et à la base de données.
- Le manque de formation et de sensibilisation des administrateurs aux questions de sécurité.

Sans enter dans le détail, on rappellera que les actions malveillantes exploitent différents types de failles :

- **Injection SQL** : Execution de requêtes SQL malicieuses.

- **XSS** (Cross-Site Scripting) : Injection de code HTML malicieux.

- **CSRF** (Cross-Site Request Forgeries) : Exécution de commandes involontaires aux utilisateurs accrédités.

- **RFI** (Remote File Inclusion) : Insertion de fichiers ou de programmes malicieux permettant de récupérer des informations confidentielles (codes d'accès, etc.) et d'exécuter du code frauduleux.

- **Upload de fichier** : Faille dans un script d'upload permettant de télécharger des fichiers malicieux.

- **Attaque par force brute** : Permet de révéler les mots de passe à faible niveau de sécurité (mots courants, peu de caractères, etc.).

...

D'autres attaques ciblent directement le ou les administrateurs :

- **Le fishing** (ou hameçonnage): la technique consiste à faire croire à la victime qu'elle s'adresse à un tiers de confiance — banque, administration, etc. — afin de lui soutirer ses identifiants.

- **Le social engineering** : technique proche du hameçonnage, à la différence que la personne mal intentionnée exploite la hiérarchie sociale, en se faisant passer par exemple pour un chef de service.

- **Le piratage du/des ordinateur(s) ou terminaux mobiles du ou des administrateur(s) :** la prudence est de mise, et il convient de ne pas stocker ses mots de passe en clair sur son ordinateur ou dans ses e-mails (professionnels et personnels).

...

**NB :** 2007 et 2008 furent les années noires de WordPress en terme de sécurité (étude Sécunia 2007). Depuis, les rapports de vulnérabilité se font plus rares, l'éditeur prenant de plus en plus en compte les problématiques sécuritaires, dans le cadre des différentes évolutions du CMS.

## II/ Prévention, protection et bonnes pratiques

La protection d'un site Internet réalisé via le CMS WordPress passe par l'application de dispositions adéquates au niveau du code source, du suivi du site/moteur, de la gestion des droits, des plugins/widgets utilisés, de la formation des administrateurs ou encore de l'infrastructure d'hébergement...

### a) Choix des plugins :

Lors du développement et de la création du site internet, il fortement recommandé de conserver un **oeil critique sur les plugins** qui pourront être utilisés, notamment sur les aspects liés à la sécurité et à leur pérennité. Les problèmes relatifs à la surexposition d'un moteur CMS (liée au fait que son code source soit disponible) s'appliquent également aux plugins. Dans la mesure du possible, il est donc **important de pas en installer à outrance** et de bien les sélectionner, notamment dans le cas de plugins ayant des fonctions d'upload ou laissant aux utilisateurs la possibilité d'insérer des données en BDD.

En complément, **il est recommandé d'installer des plugins dits « de sécurité »**, permettant de protéger le site Internet contre certains types d'attaques, ou d'actions malveillantes.

### b) Thèmes et templates :

Dans la mesure du possible, il faut **limiter l'utilisation de thèmes réalisés par des tiers**. Les templates sont souvent créés par des infographies ou des intégrateurs ayant peu d'expérience sur les protections et les bonnes pratiques de développement. Ces derniers peuvent par conséquent créer des failles à leur insu.

### c) Développement :

Pour les besoins d'un site, il est souvent nécessaire de développer soi-même de nouvelles fonctionnalités ou de nouveaux plugins/widgets. Ces modifications pouvant générer des failles, il convient de les apporter avec **soin et prudence.**

### d) Mesures de protection :

Pour garantir un niveau de sécurité optimale, il est possible (et même essentiel) d'appliquer des **mesures de protection supplémentaires :**

- Suppression des accès par défaut (suppression du profil 'admin' proposé par défaut sous WordPress).
- Double authentification sur le répertoire d'administration.
- Filtrage par IP pour les zones sensibles (administration, etc.).
- Protection des fichiers de configuration (encryptage, blocage de l'affichage).
- Activation du cryptage d'une partie des données via le fichier de configuration.
- Protection des répertoires de WordPress (/wp-includes/, /wp-admin/ et une partie de /wp-content/).
- Modification du préfixe proposé par défaut pour les tables de la base de données (wp_).
- Anonymisation des signatures permettant d'identifier le CMS.
- Masquer les données qui pourraient permettre d'identifier de manière précise les versions en place (version du moteur WordPress, versions et noms des plugins).
- Ne pas afficher les erreurs PHP.
- Se prémunir contre les attaques de type force brute en limitant le nombre de tentatives de connexion erronées.

### e) Login et Mot de passe :

Toute politique de sécurité (CMS et 'from scratch') induit la définition d'une gestion rigoureuse des mots de passe et données sensibles, au sein de laquelle il faut respecter des règles simples. Par exemple, ne pas employer de **noms d'utilisateurs** trop communs (admin, administration, user, root, nom du site,

etc.), générer des **mots de passe** complexes (au moins 12 caractères alphanumériques aléatoires avec des caractères spéciaux), ne jamais les communiquer par courriel, exclusivement les stocker sur des **bases cryptées** ou sur papier, etc.

#### f) Hébergement :

L'idéal est d'héberger le site Internet sur une infrastructure d'hébergement ayant mis en place des **mesures de protection actives.** Elles permettent d'appliquer une barrière supplémentaire à de nombreuses attaques (DDoS, injection SQL, XSS…). En cas de piratage, cette infrastructure doit être en mesure de rétablir rapidement un backup, tout en fournissant les logs permettant de localiser la faille et éventuellement la provenance de l'attaque.

On préférera ainsi faire appel à un **prestataire dédié** aux entreprises (en opposition aux hébergeurs grand public), proposant notamment un service d'infogérance active (Managed Hosting). On peut par exemple citer les sociétés Ecritel, Oxeva, ou encore U4 : : HOSTING.

#### g) Maintenance :

Pour conserver une sécurité maximale, il est conseillé de maintenir **une veille active** sur le site Internet et de procéder aux mises **à jour du site et à l'application des patchs correctifs du moteur WordPress** dès leurs publications. Cette procédure est également à **appliquer sur les différents plugins tiers utilisés.** En 2007, une étude (Blog Security) a révélé que 98% des blogs WordPress étaient potentiellement exploitables, car ils étaient notamment basés sur des versions obsolètes...

## III/ Amélioration des sécurités passives et actives de WordPress :

Il existe un certain nombre de plugins **permettant d'améliorer de manière significative la sécurité de WordPress.** Leur mise en place permet de réduire et de limiter un certain nombre d'attaques et de failles potentielles. Il vaut mieux **privilégier la qualité des plugins utilisés** plutôt que de les installer en masse. En effet, la **surprotection d'un site via l'installation de plugins** peut au final amener à un **défaut de protection, en raison des conflits entre les plugins** ou d'une faille sur l'un d'entre eux.

### 3 plugins de référence :

#### a) Wordfence

- Scan régulièrement les fichiers « core » de WordPress en les comparant avec les versions officielles afin de détecter les éventuelles altérations.
- Limite et bloque les bots crawler trop agressifs (scans, aspirateurs, etc.).
- Vérifie le niveau de sécurité des mots passes créés par les utilisateurs.
- Vérifie les URLs inclus dans les commentaires.
- Vérifie les fichiers à partir d'une base de signature de 44.000 malwares connus.
- Applique une sécurité sur les interfaces de connexion (protection contre les attaques de type force brute, blocage de l'affichage de données permettant d'identifier un utilisateur WordPress).

#### b) BulletProof Security

- Protège contre les attaques de type Injection SQL, XSS, RFI, CSRF et Base64.
- Filtre et bloque les meta-tags WordPress.
- Vérifie les droits sur les fichiers et dossiers – CGI / DSO SAPI.

#### c) Chap Secure Login

- Empêche la transmission en clair des mots de passe.

## Conclusions

Les CMS open-source (en particulier WordPress) restent **des solutions très puissantes et offrent de nombreuses fonctionnalités**… au prix d'une plus grande vulnérabilité que les solutions développées de manière spécifique ou sur-mesure. En effet, ces dernières offrent la possibilité d'intégrer des mesures de sécurité plus draconiennes et performantes (limitation des droits d'accès à la base de données, mise en place d'honey-pot ou administration factice, scan récurrent des fichiers sources, etc.).

Si la sécurité est un critère primordial (marques 'sensibles' fortement exposées, secteur bancaire, administrations publiques, e-commerce, …) on préféra des approches spécifiques, qui nécessiteront en revanche des investissements plus importants (à fonctionnalités et dynamisation égales, de l'ordre de + 30% à +100% et plus). **La sécurité** n'échappe pas à la règle… Elle a un prix.

Néanmoins, **WordPress reste parfaitement adapté à la très grande majorité des sites/projets Web.** De plus, il est possible de **fortement réduire les risques de sécurité** inhérents à l'emploi de ce CMS au moyen de dispositions complémentaires : application de mesures de protection, anonymisation des sources, limitation des plugins exploités, maintenance proactive, etc. Pour éprouver les dispositifs mis en oeuvre, on pourra faire appel à un tiers spécialisé - de type 'white hat' - et lui faire réaliser un audit de sécurité pour s'assurer qu'aucune faille ne subsiste. C'est la solution ultime (prévoir un budget de 3000€ à 50 000€ et plus, selon les paramètres à contrôler).

Rappelons également qu'une grande partie des attaques ciblées sont issues de techniques de hameçonnage ou d'ingénierie sociale. Une part importante de la sécurité repose donc sur la **responsabilité-même des administrateurs,** lequels se doivent d'être sensibilisés et formés aux questions sécuritaires. De cette manière, il est ainsi possible de **limiter les attaques extérieures tout comme les actes de négligence.**

## Sources :

Wikipedia, Wordpress.com, Wordpress.org, WordpressMU.org, Yoast.com, U4 : : networks, comScore, mediashman.com, fr2.php.net, Journal du Net, bugs.php.net, Blog Security, Sécunia, ...