

**Site WordPress**

SECURITE

03/05/2021



| **Objet du document :** | SECURITE |
| --- | --- |
| **Projet concerné :** | Site WordPress |
| **Rédacteur :** | Alexandre GUASCH |

##### Gestion des versions du document

| **Version** | **Date** | **Rédacteur** | **Commentaire** |
| --- | --- | --- | --- |
| 1.0 | 03/05/2021 | A. Guasch | Version de base |

**Temps d’intervention - WordPress:**

0.5 jours

**Temps d’intervention - Autres sites :**

0.125 jours

**Sommaire**

[1 Contexte 2](#_Toc71036934)

[2 La base pour WordPress 3](#_Toc71036935)

[2.1 Nettoyage des fichiers bavards 3](#_Toc71036936)

[2.2 Isolation du fichier de configuration 3](#_Toc71036937)

[2.3 Durcissement de la configuration 4](#_Toc71036938)

[2.3.1 Désactiver l’éditeur de fichiers pour les extensions et thèmes 4](#_Toc71036939)

[2.3.2 Générer des nouvelles clés de sécurité 4](#_Toc71036940)

[2.3.3 Création d’un dossier personnalisé pour les images 4](#_Toc71036941)

[2.4 Plugins de sécurité WordPress 5](#_Toc71036942)

[2.4.1 WordFence 5](#_Toc71036943)

[2.4.2 WP Sentry 6](#_Toc71036944)

[2.4.3 WPS Hide Login 6](#_Toc71036945)

[3 Durcissement des entêtes http (tout type de site) 7](#_Toc71036946)

[3.1 Apache ou Nginx ? 7](#_Toc71036947)

[3.2 Nginx – en détails 7](#_Toc71036948)

[3.3 Nginx – Concis 10](#_Toc71036949)

[3.4 Apache 11](#_Toc71036950)

[3.5 Scanner le site 11](#_Toc71036951)

[4 Durcissement des droits de lecture et écriture des fichiers et dossiers du WordPress 12](#_Toc71036952)

[5 Prévention contre le hotlinking (tout type de site) 13](#_Toc71036953)

[5.1 Apache 13](#_Toc71036954)

[5.2 Nginx 13](#_Toc71036955)

[6 Sauvegardes automatiques (tout type de site) 14](#_Toc71036956)

[6.1 Spécificité WordPress - Complément 14](#_Toc71036957)

# Contexte

Ce document a pour objectif de décrire comment durcir la sécurité d’un WordPress et de son environnement, en passant par la configuration du serveur et l’installation de plugins de sécurité.

# La base pour WordPress

## Nettoyage des fichiers bavards

À la racine du projet, supprimer les fichiers : (ou les rendre inaccessibles depuis le htaccess)

* license.txt
* readme.html
* wp-config-sample.php

## Isolation du fichier de configuration

1. Déplacer le fichier wp-config.php à l’extérieur de la racine du projet
2. À la racine du projet, créer un nouveau wp-config.php contenant les lignes suivantes :  
     
   **<?php**

**/\*\* Absolute path to the WordPress directory. \*/**

**if ( !defined('ABSPATH') )**

**define('ABSPATH', dirname(\_\_FILE\_\_) . '/');**

**/\*\* Location of your WordPress configuration. \*/**

**require\_once(ABSPATH . '../wp-config.php');**

-------------------------------------------------------------------------------------------

**/!\ Important :** Dans cet exemple, le fichier wp-config a été déplacé au dossier du dessus. Si votre emplacement est différent, modifier le chemin   
'../wp-config.php' par celui correspondant.  
-------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------

**/!\ Important :** Si une anomalie avec open\_basedir apparaît, dans le fichier de configuration de PHP, ajouter le nouveau chemin dans la directive open\_basedir.  
  
exemple : (ici /var/www/ pour l’exemple courant)  
**open\_basedir = "/var/www/monprojet/;/var/www/;/tmp/"**  
-------------------------------------------------------------------------------------------

1. Enregistrer le fichier
2. La configuration est désormais sécurisée et de nouveau opérationnelle

## Durcissement de la configuration

Pour effectuer cette manipulation, ouvrir le fichier wp-config.php de WordPress.

### Désactiver l’éditeur de fichiers pour les extensions et thèmes

Dans le fichier wp-config.php, ajouter la ligne suivante :

**define( 'DISALLOW\_FILE\_EDIT', true );**

Cela aura pour effet d’empêcher toute édition de fichiers PHP / CSS / JS directement depuis l’administration du WordPress.

### Générer des nouvelles clés de sécurité

1. Utiliser le générateur de clés à l’adresse : <https://api.wordpress.org/secret-key/1.1/salt/>
2. Récupérer les clés générés pour remplacer les anciennes présentes dans le wp-config.php
3. Enregistrer le fichier

-------------------------------------------------------------------------------------------

**/!\ Note :** Quand le faire ?  
- À chaque migration de serveur

- Idéalement une fois par mois, il est possible de l’automatiser avec le plugin Salt Shaker : <https://wordpress.org/plugins/salt-shaker/>   
-------------------------------------------------------------------------------------------

### Création d’un dossier personnalisé pour les images

-------------------------------------------------------------------------------------------

**/!\ Important :** À réaliser avant tout import de média dans la Galerie.  
-------------------------------------------------------------------------------------------

1. Créer un dossier nommé files ou docs ou autre selon les besoins du projet.
2. Placer le dossier à la racine du projet
3. Dans le fichier wp-config.php, ajouter la ligne suivante :  
     
   **define('UPLOADS', 'files');**

## Plugins de sécurité WordPress

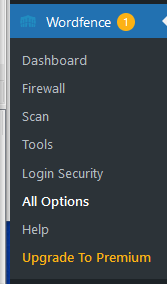
### WordFence

Cette extension une protection de type pare-feu et effectue une veille sur les dernières mises à jour de WordPress ainsi que des extensions et thèmes installés.

Ainsi, lorsqu’une mise à jour de sécurité est détectée, l’administrateur est notifié par email.

<https://fr.wordpress.org/plugins/wordfence/>

Une fois installé, il est nécessaire de configurer le plugin en suivant les étapes suivantes :

1. Via l’admin du WordPress, accéder à la section « All Options » de WordFence  
   
2. Déplier la section « Email Alert Preferences »  
   Changer les valeurs suivantes :  
   - Alert me with scan results of this severity level or greater : **medium**
3. Déplier la section « Brute Force Protection »  
   Changer les valeurs suivantes :  
   - Lock out after how many login failures : 5  
   - Lock out after how many forgot password attempts : 5  
   - Amount of time a user is locked out : 4 hours
4. Dans le coin supérieur droit du navigateur, cliquer sur « Save Changes »

### WP Sentry

Reporte les erreur PHP et JavaScript.

<https://wordpress.org/plugins/wp-sentry-integration/>

### WPS Hide Login

Permet de changer l’adresse de connexion et d’interdir l’accès au wp-admin pour les utilisateurs non connectés.

<https://wordpress.org/plugins/wps-hide-login/>

# Durcissement des entêtes http (tout type de site)

## Apache ou Nginx ?

Pour rappel, Nginx offre un gain de performance de 20% sur les temps d’exécution des scripts PHP comparés à Apache et est un service de Proxy modulable.  
  
Pour le présent chapitre, la configuration de Nginx est rédigée dans une version détaillée, suivi d’Apache, dans une version plus concise.

## Nginx – en détails

1. Se connecter sur le serveur via SSH
2. Se rendre dans **/etc/nginx/sites-enabled/**
3. Ouvrir le fichier de configuration lié au site
4. Entre les accolades de la directive « Server », ajouter les lignes d’entête rédigées en orange ci-dessous :  
     
   **add\_header Content-Security-Policy "default-src https;script-src 'self' https://www.google-analytics.com;" always;**  
   -------------------------------------------------------------------------------------------

**/!\ Note :** Content-Security-Policy sert à autoriser l’exécution des fichiers statiques aux sources de confiance uniquement. Ainsi, en cas d’attaque XSS, les sources externes de l’attaquant ne seront pas exécutées sur le site.  
  
Pour plus d’informations sur le fonctionnement de Content-SecurityPolicy :  
<https://www.keycdn.com/support/content-security-policy>   
-------------------------------------------------------------------------------------------

-------------------------------------------------------------------------------------------

**/!\ Important :** Pour cette directive, il est impératif au préalable de lister l’ensemble des ressources externes utilisées pour ensuite reporter leur domaine dans l’entête.  
-------------------------------------------------------------------------------------------

**add\_header Referrer-Policy "strict-origin" always;**

-------------------------------------------------------------------------------------------

**/!\ Note :** Referrer--Policy sert à choisir quelle source récupérer lorsqu’un utilisateur arrive sur le site.   
-------------------------------------------------------------------------------------------

**add\_header X-XSS-Protection "1; mode=block" always;**  
-------------------------------------------------------------------------------------------

**/!\ Note :** X-XSS-Protection sert à activer la protection XSS, bien qu’activé par défaut sur plusieurs navigateurs, cette méthode est appliquée de manière préventive.   
-------------------------------------------------------------------------------------------

**add\_header Strict-Transport-Security "max-age=63072000; includeSubdomains; preload" always;**  
-------------------------------------------------------------------------------------------

**/!\ Note :** Strict-Transport-Security indique aux navigateurs web que le présent site doit strictement et uniquement utiliser du HTTPS. Cela assure qu’aucune connexion HTTP non sécurisée n’aboutisse.  
  
Pour plus d’informations sur le fonctionnement de Strict-Transport-Security :  
<https://www.keycdn.com/support/http-strict-transport-security>   
-------------------------------------------------------------------------------------------  
  
**add\_header X-Frame-Options "SAMEORIGIN" always;**  
-------------------------------------------------------------------------------------------

**/!\ Note :** X-Frame-Options ajoute une protection clickjacking, c’est à dire, une protection contre le vol de clics d’utilisateurs. Cette option bloque l’utilisation du site depuis des frames et iframes via un site tiers.   
-------------------------------------------------------------------------------------------

**add\_header Expect-CT "max-age=604800, enforce, report-uri='https://www.example.com/report'" always;**  
-------------------------------------------------------------------------------------------

**/!\ Note :** Expect-CT oblige le navigateur à vérifier l’authenticité et la transparence du certificat utilisé par le site.   
  
Pour plus d’informations sur le fonctionnement de Expect-CT :  
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Expect-CT>   
-------------------------------------------------------------------------------------------  
-------------------------------------------------------------------------------------------

**/!\ Important :** remplacer le lien **www.example.com/report** par le lien du site qui doit recevoir les échecs liés au certificat  
-------------------------------------------------------------------------------------------

**add\_header X-Content-Type-Options "nosniff" always;**

-------------------------------------------------------------------------------------------

**/!\ Note :** X-Content-Type-Options protège les navigateurs contre le sniffing de données. Cette directive a pour effet d’aider le contenu d’un téléchargement à se diriger vers le bon chemin, réduisant le danger d’une interception.  
-------------------------------------------------------------------------------------------

**add\_header Permissions-Policy "geolocation=(),midi=(),sync-xhr=(),microphone=(),camera=(),magnetometer=(),gyroscope=(),fullscreen=(self),payment=()" always;**  
-------------------------------------------------------------------------------------------

**/!\ Note :** Permissions-Policy verrouille certaines fonctionnalités du navigateur pour sécuriser l’utilisateur. (comme par exemple, la lecture automatique de vidéos ou l’utilisation d’une webcam)  
-------------------------------------------------------------------------------------------

1. Enregistrer le fichier
2. Redémarrer le service Nginx

## Nginx – Concis

1. Se connecter sur le serveur via SSH
2. Se rendre dans /etc/nginx/sites-enabled/
3. Ouvrir le fichier de configuration lié au site

Entre les accolades de la directive « Server », ajouter les lignes d’entête rédigées en orange ci-dessous :  
  
**add\_header Content-Security-Policy "default-src https;script-src 'self' https://www.google-analytics.com;" always;  
add\_header Referrer-Policy "strict-origin" always;**

**add\_header X-XSS-Protection "1; mode=block" always;  
add\_header Strict-Transport-Security "max-age=63072000; includeSubdomains; preload" always;  
add\_header X-Frame-Options "SAMEORIGIN" always;  
add\_header Expect-CT "max-age=604800, enforce, report-uri='https://www.example.com/report'" always;**  
**add\_header X-Content-Type-Options "nosniff" always;**

**add\_header Permissions-Policy "geolocation=(),midi=(),sync-xhr=(),microphone=(),camera=(),magnetometer=(),gyroscope=(),fullscreen=(self),payment=()" always;**

1. Enregistrer le fichier
2. Redémarrer le service Apache

-------------------------------------------------------------------------------------------

**/!\ Important :** En cas d’erreur après redémarrage du service Nginx OU si le site affiche une page blanche sans style CSS, merci de lire le bloc « Important » pour la directive   
«Content-Security-Policy » présent dans la version détaillée pour Nginx.  
-------------------------------------------------------------------------------------------

## Apache

1. Se connecter sur le serveur via SSH
2. Se rendre dans /etc/apache2/sites-enabled/
3. Ouvrir le fichier de configuration lié au site
4. Ajouter les lignes d’entête rédigées en orange ci-dessous :  
     
   **Header always set Content-Security-Policy "default-src https;script-src 'self' https://www.google-analytics.com;"  
   Header always set Referrer-Policy "strict-origin"  
   Header always set X-XSS-Protection "1; mode=block"  
   Header always set Strict-Transport-Security "max-age=63072000; includeSubdomains; preload"  
   Header always set X-Frame-Options "SAMEORIGIN"  
   Header always set Expect-CT "max-age=604800, enforce, report-uri="https://www.example.com/report"  
   Header always set X-Content-Type-Options "nosniff"  
   Header always set Permissions-Policy "geolocation=(),midi=(),sync-xhr=(),microphone=(),camera=(),magnetometer=(),gyroscope=(),fullscreen=(self),payment=()"**
5. Enregistrer le fichier
6. Redémarrer le service Apache

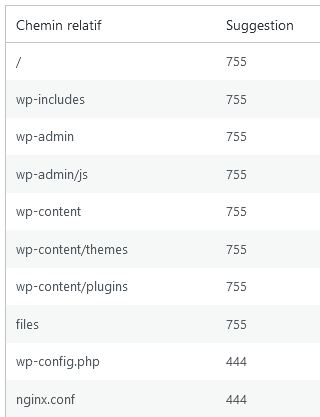
## Scanner le site

Après avoir configuré Apache ou Nginx, on vérifie que tout fonctionne via le site <https://securityheaders.com/> en saisissant l’adresse du site à vérifier.

On ajuste ensuite les paramètres nécessaires jusqu’à obtenir la note optimale pour notre site. (B ou supérieur)

En complément, on vérifie la sécurité de manière plus approfondie sur le site sucuri : <https://sitecheck.sucuri.net/>

# Durcissement des droits de lecture et écriture des fichiers et dossiers du WordPress



Se connecter sur le serveur via SSH puis aller dans le dossier du projet pour ensuite saisir les commandes suivantes :

**chmod 0755 /**

**chmod 0755 files**

**chmod 0755 wp-includes**

**chmod 0755 wp-admin**

**chmod 0755 wp-admin/js**

**chmod 0755 wp-content**

**chmod 0755 wp-content/themes**

**chmod 0755 wp-content/plugins**

**chmod 0444 wp-config.php**

**chmod 0444 nginx.conf**

# Prévention contre le hotlinking (tout type de site)

## Apache

Dans le fichier .htaccess, ajouter les lignes suivantes :

**RewriteEngine on**

**RewriteCond %{HTTP\_REFERER} !^$**

**RewriteCond %{HTTP\_REFERER} !^http(s)?://(www.)?yourdomain.com [NC]**

**RewriteRule .(jpg|jpeg|png|gif)$ http://dropbox.com/hotlink-placeholder.jpg [NC,R,L]**

## Nginx

Dans le fichier de configuration du site (généralement dans /etc/nginx/sites-enabled), ajouter les lignes suivantes :

**location ~ .(gif|png|jpe?g)$ {**

**valid\_referers none blocked ~.google. ~.bing. ~.yahoo yourdomain.com \*.yourdomain.com;**

**if ($invalid\_referer) {**

**return 403;**

**}**

**}**

Redémarrer le service Nginx

-------------------------------------------------------------------------------------------

**/!\ Important :** remplacer **yourdomain.com** par le domaine du site  
-------------------------------------------------------------------------------------------

# Protection anti-aspiration de site (tout type de site)

Cette protection a pour objectif de protéger le site contre les robots aspirateurs de site qui ont pour but final de cloner le site pour ensuite le plagier, dans sa version HTML / CSS / JS et avec toutes ses images.

## Apache

1. Récupérer le fichier .htaccess opérationnel a été créé à l’adresse suivante :
2. Placer le fichier .htaccess dans le dossier parent du projet
3. Retourner à la racine du projet
4. Ouvrir le .htaccess du WordPress
5. Utiliser la directives « options » pour exploiter le .htaccess du dossier parent
6. Enregistrer le fichier

# Sauvegardes automatiques (tout type de site)

Généralement gérées par l’hébergeur ou un ingénieur réseaux, les sauvegardes automatiques (backup) sont incontournables car elles permettent de restaurer le site en cas d’attaque, de panne matériel ou encore dans le cadre d’un changement d’hébergement.

**Recommandations :**

* une sauvegarde complète chaque nuit
* sauvegardes de type incrémentales

## Spécificité WordPress - Complément

WordPress propose des extensions, en complément des systèmes de backup serveur.

Grâce à ces extensions, il est possible d’extraire une sauvegarde du site pour ensuite la stocker dans un environnement différent du site comme Amazon S3, Google Cloud Storage, Google Drive, Dropbox, OneDrive ou d’autres solutions adaptées au besoin du client.

**Liste des extensions de sauvegarde :**

* Duplicator : <https://wordpress.org/plugins/duplicator/>
* WP Time Capsule : <https://wptimecapsule.com/index.html>
* BackupBuddy : <https://ithemes.com/purchase/backupbuddy/>
* UpdraftPlus : <https://wordpress.org/plugins/updraftplus/>
* BackUpWordPress : <https://wordpress.org/plugins/backupwordpress/>
* BackWPup : <https://wordpress.org/plugins/backwpup/>
* WP BackItUp : <https://www.wpbackitup.com/>