



Nous allons installer un serveur web NGINX avec WordPress et MariaDB sur le système d'exploitation Debian 8.8 version 64 bits, sur un ordinateur de récupération possédant une configuration de 2 giga de ram pour un disque dur de 80 giga. Nous verrons, étape par étape, le déroulement de l'installation. Il est recommandé d'avoir une machine client pour effectuer des tests.

## Sommaire

1	Mettre à niveau Debian .....	3
1.1	Se connecter en SuperUtilisateur (root) .....	3
1.2	Mettre à jour le système .....	3
1.3	Installation des paquets NGINX .....	3
1.4	Installation des paquets PHP 5 .....	4
1.5	Installation du FTP (VSFTPD) .....	6
2	Configuration de NGINX .....	6
3	Configuration du serveur FTP .....	6
3.1	Un serveur FTP ? Pourquoi faire ? .....	6
3.2	Les préparatifs .....	6
3.3	Modifier le fichier vsftpd.conf .....	8
3.4	Paramétrer les utilisateurs .....	10
3.5	Utiliser le service FTP .....	11

## 1 Mettre à niveau Debian

### 1.1 Se connecter en SuperUtilisateur (root)

*(Le SuperUtilisateur, aussi surnommé root, est le nom de l'administrateur du serveur)*

Pour se connecter en t'en que SuperUtilisateur, taper la commande **« su »** et entrer le mot de passe. Si vous n'avez pas choisis de mot de passe lors de l'installation, celui-ci sera par défaut **« root »**.

### 1.2 Mettre à jour le système

Pour effectuer une mise à jour, il faut utiliser les fichiers disponibles dans les dépôts APT présent dans le fichier `/etc/apt/source.list`.

Pour cela, exécuter les commandes suivantes :

**« apt-get update »** qui recherchera les paquets disponibles.

**« apt-get upgrade »** qui effectuera la mise à jour.

### 1.3 Installation des paquets NGINX

*(Apache est un logiciel libre. C'est un serveur http populaire)*

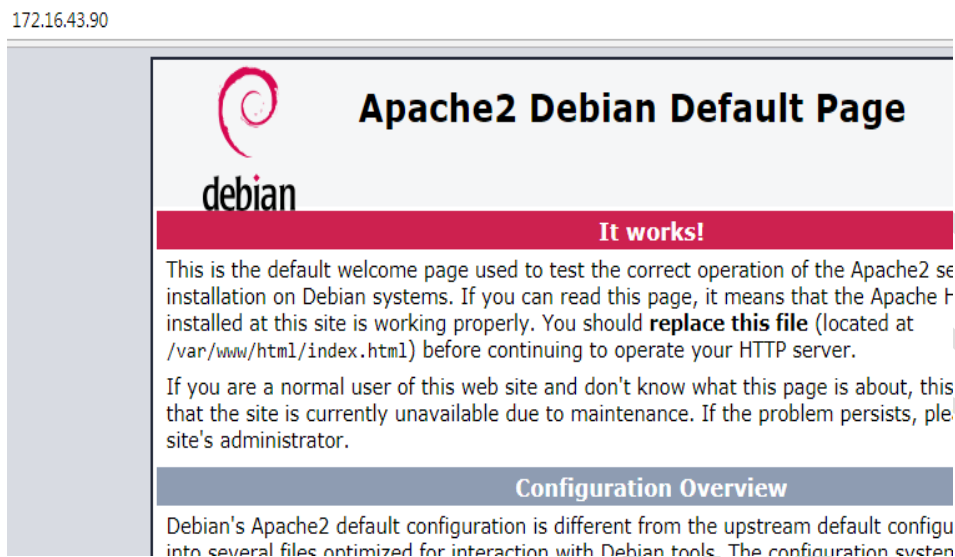
Il faut installer un serveur http qui permettra d'afficher nos différentes pages internet. Ici, on utilisera NGINX.

Pour l'installer, taper la commande suivante :

**apt-get install nginx**

Vérifier que l'installation s'est bien effectuée en entrant dans votre navigateur l'adresse suivante : **http://IP\_DE\_VOTRE\_SERVEUR/**

Si s'est bien effectué, le navigateur affiche une page similaire à celle-ci :



## 1.4 Installation des paquets PHP 5

(PHP est un langage de script conçu pour le développement d'application web.)

L'intérêt d'installer PHP 5 sur le serveur est simple. Il permet aux sites disposant d'une partie dynamique en PHP, de pouvoir être héberger sur notre serveur.

Pour l'installer, taper la commande suivante :

**`apt-get install php5`**

En général, le PHP utilise une base de données nommé MySQL. Pour l'installer, taper la commande suivante :

**`apt-get install mariadb-server`**

Il faut ensuite définir le mot de passe de l'utilisateur root :

**`sudo mysql_secure_installation`**

Limiter l'accès root uniquement en local... si vous rencontrez l'erreur

**`ERROR 1698 (28000): Access denied for user 'root'@'localhost'`**

Soyez bien en Administrateur ou taper les commandes avec **`sudo`**.

**`sudo mysql -u root`**

Pour assurer le bon fonctionnement de PHP 5 avec les bases de données MySQL, il faut installer des librairies dénommé PHP5-MySQL. Pour cela, taper la commande suivante :

***apt-get install php5-mysql***

Pour gérer les bases de données de MySQL, installer le script PHP permettant cela. Il se nomme PHPMyAdmin. Pour l'installer, taper la commande suivante :

***apt-get install phpmyadmin***

Cette installation réclame de choisir le serveur web à reconfigurer automatiquement. Ici, **cocher la case NGINX** grâce à la barre espace puis valider.

Pour pouvoir voir PHPMyAdmin grâce au serveur web, créer un lien PHPMyAdmin dans le répertoire /var/www. Pour effectuer cela, taper la commande suivante :

***ln -s /usr/share/phpmyadmin /var/www/phpmyadmin***

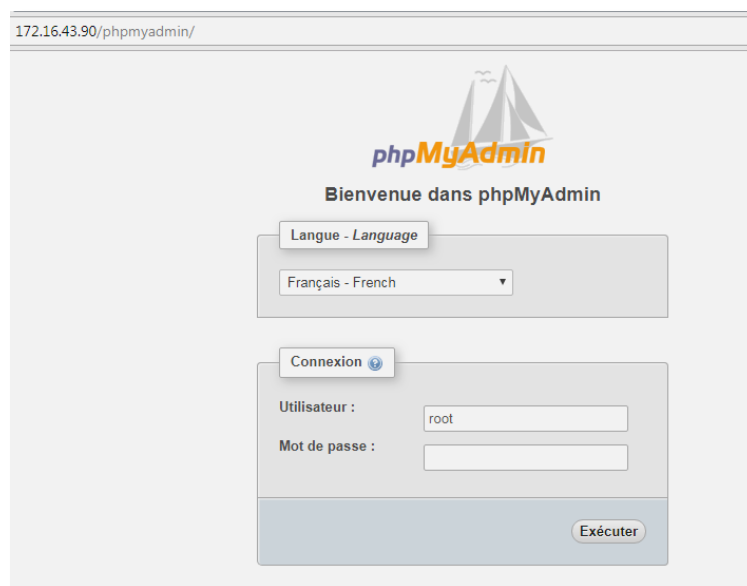
Afin de vérifier que l'installation s'est bien déroulée, taper dans votre navigateur l'adresse suivante : ***http://IP\_DE\_VOTRE\_SERVEUR/phpmyadmin/***

Si le formulaire ne s'affiche pas et que vous avez une **erreur 404**, il se peut que ça vienne du fichier **/etc/apache2/sites-available/000-default.conf**. Dans ce cas modifiez **DocumentRoot /var/www/html** par **DocumentRoot /var/www/** puis relancez le serveur apache :

***service nginx restart***

Et réactualisez ***http://IP\_DE\_VOTRE\_SERVEUR/phpmyadmin/***.

Si tout s'est bien passé, le navigateur affiche une page similaire à celle-ci :



## 1.5 Installation du FTP (VSFTPD)

VSFTPD est un serveur FTP très sécurisé. Il permet l'échange de fichier entre des interfaces.

Afin de pouvoir ajouter des fichiers sur le site du serveur web, il faut installer VSFTPD grâce à la commande suivante :

***apt-get install vsftpd***

## 2 Configuration de NGINX

## 3 Configuration du serveur FTP

### 3.1 Un serveur FTP ? Pourquoi faire ?

Le serveur web est désormais opérationnel. Il faut désormais pouvoir mettre des fichiers dans ce serveur. Pour cela, il est possible d'utiliser un serveur FTP sécurisé tel que VSTPD (Very Secure File Transfert Protocol Daemon).

### 3.2 Les préparatifs

Utilisez les commandes ***apt install vsftpd*** et ***apt upgrade vsftpd*** afin d'avoir les paquets nécessaires à jour pour le serveur FTP.

Commencer par créer un répertoire qui contiendra tous nos fichiers. Pour cela, utiliser la commande suivante :

***Mkdir /etc/vsftpd***

Afin d'avoir la possibilité de revenir en arrière, effectuer une sauvegarde comme ceci :

***cp /etc/vsftpd.conf /etc/vsftpd.conf.bak***

***cp /etc/pam.d/vsftpd /etc/pam.d/vsftpd.bak***

Florian MOREAU

## 3.3 Modifier le fichier vsftpd.conf

Afin d'utiliser nos utilisateurs virtuels, il faut reconfigurer le fichier vsftpd.conf. Pour cela, supprimer le fichier ainsi :

***rm /etc/vsftpd.conf***

Puis recrée le comme ceci :

***# nano /etc/vsftpd.conf***

Puis coller le texte de la page suivante à l'intérieur du nouveau fichier crée : (Les explications sont écrites après chaque « # ». Cela s'appelle des commentaires. Ils ne sont pas utiles au bon fonctionnement du fichier.)





```
# Ceci configure VSFTPD en mode "standalone"
listen=YES

# On désactive les connexions anonymes
# et on active les non-anonymes (c'est le cas des utilisateurs virtuels) :
anonymous_enable=NO
local_enable=YES

# Pour des raisons de sécurité on interdit toute action d'écriture :
write_enable=NO
anon_upload_enable=NO
anon_mkdir_write_enable=NO
anon_other_write_enable=NO

# 'guest_enable' est très important: cela active les utilisateurs virtuels !
# 'guest_username' fait correspondre tous les utilisateurs virtuels à
# l'utilisateur 'virtual' que nous avons défini plus haut, et au home
# correspondant : '~virtual/'.
guest_enable=YES
guest_username=www-data

# On définit les droits par défaut des fichiers uploadés
anon_umask=022

# On veut que les utilisateurs virtuels restent chez eux : '~virtual/'
# (attends, on leur a fait un toit, c'est pas pour rien !)
chroot_local_user=YES

# On définit le nombre maximum de sessions à 200 (les nouveaux clients
recevront
# un message du genre: "erreur : serveur occupé").
# On définit le nombre maximum de sessions par IP à 4
max_clients=200
max_per_ip=4

#####
# Debian customization #
# (ou adoptons la Debian attitude) #
#####
# Some of vsftpd's settings don't fit the Debian filesystem layout by
# default. These settings are more Debian-friendly.
#
# This option should be the name of a directory which is empty. Also, the
# directory should not be writable by the ftp user. This directory is used
# as a secure chroot() jail at times vsftpd does not require filesystem
# access.
secure_chroot_dir=/var/run/vsftpd
#
# This string is the name of the PAM service vsftpd will use.
pam_service_name=vsftpd
#
# This option specifies the location of the RSA certificate to use for SSL
# encrypted connections.
rsa_cert_file=/etc/ssl/certs/vsftpd.pem

# Permet d'utiliser les configurations individuelles pour chaque utilisateur
user_config_dir=/etc/vsftpd/vsftpd_user_conf
```

Sauvegarder et quitter.

## 3.4 Paramétrer les utilisateurs

Grâce au chroot, cela renforce la sécurité, car le répertoire est considéré comme étant un répertoire racine, il n'est donc pas possible de remonter la hiérarchie.

Chrouter les utilisateurs dans leur répertoire respectif. Pour cela, il suffit de créer le répertoire qui servira à contenir nos différents fichiers par utilisateur. Et il faut créer le fichier pour chaque utilisateur dans notre base de données. Pour cela, voici les commandes à effectuer :

```
mkdir /etc/vsftpd/vsftpd_user_conf/
```

```
nano /etc/vsftpd/vsftpd_user_conf/user1
```

Puis, ajouter cela dans le fichier avant d'enregistrer et de fermer :

```
anon_world_readable_only=NO
local_root=/home/test1/public_html
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES
anon_other_write_enable=YES
```

Redémarrer le service FTP comme ceci :

```
/etc/init.d/vsftpd restart
```

Pour ajouter un utilisateur, ajoutez le login et son mot de passe dans login.txt puis exécuter :

```
db5.3_load -T -t hash -f /etc/vsftpd/login.txt /etc/vsftpd/login.db
```

Ajoutez les informations de l'utilisateur dans  
/etc/vsftpd/vsftpd\_user\_conf/[NOM\_DE\_L\_UTILISATEUR]

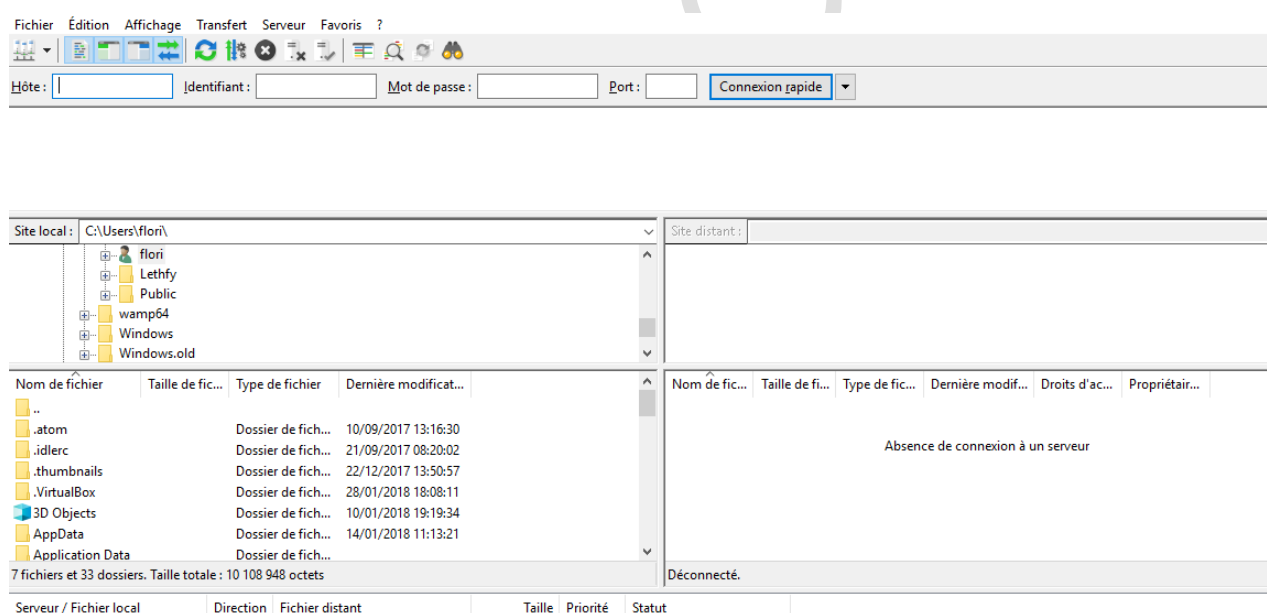
```
/etc/init.d/vsftpd restart
```

## 3.5 Utiliser le service FTP

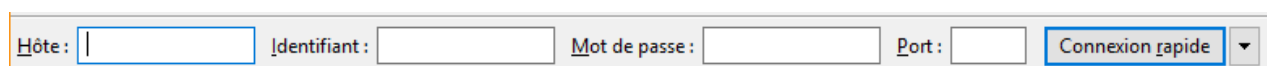


Pour utiliser le service FTP avec une interface sous Windows sur le même réseau que le serveur, il faut installer FileZilla. Pour cela, rendez-vous sur le site officiel du logiciel.

Une fois le logiciel installé, vous aurez une interface du même type que celle-ci :



Il suffit de se rendre sur cette barre d'outil :



Puis, entrer l'hôte sous cette forme : ftp://IP\_DU\_SERVEUR avec l'identifiant et le mot de passe définie auparavant. Si cela ne fonctionne pas, crée un nouvel utilisateur depuis le serveur avec la commande **useradd IDENTIFIANT**



Désormais, le serveur web sous Debian 8.8 est fonctionnel. Il peut recevoir des fichiers d'utilisateur se trouvant sur le même réseau que lui grâce à un service FTP.

Florian MOREAU