

Installation de wordpress

Au préalable il faudra installer apache2. cf. Installation apache2 sur debian + https

Installer php

php et les extensions utile pour wordpress

```
apt install php php-mysqli php-cli php-curl php-gd php-mbstring php-xml  
php-xmllrpc -y
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Mon Dec 30 17:25:33 CET 2024 on tty1  
user@debian-pc:~$ su -  
Mot de passe :  
root@debian-pc:~# apt install php php-cli php-curl php-gd php-mbstring php-xml php-xmllrpc -y
```

Installation de mariadb

Wordpress a besoin d'une base de donnée.

```
apt install mariadb-server mariadb-client
```

```
Paramétrage de php-gd (2:8.3+35+0-20240521.3+deb12u1) ...  
Traitement des actions différées (« triggers ») pour php8.3-cli (8.3.15-1+0~  
Traitement des actions différées (« triggers ») pour libapache2-mod-php8.3 (8.3.15-1+0~  
Traitement des actions différées (« triggers ») pour php8.4-cli (8.4.2-1+0~2  
Traitement des actions différées (« triggers ») pour php8.4-phdbg (8.4.2-1+0~2  
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u  
root@debian-pc:~# apt install mariadb-server mariadb-client
```

Lancement du service et démarrage avec l'OS

```
systemctl start mariadb  
systemctl enable mariadb
```

```
Paramétrage de libcg1-fast-perl (1:2.15-1) ...  
Paramétrage de mariadb-plugin-provider-snappy (1:10.11.6-0+deb12u1) ...  
Traitement des actions différées (« triggers ») pour man-db (2.11.  
Traitement des actions différées (« triggers ») pour libc-bin (2.3  
Traitement des actions différées (« triggers ») pour mariadb-server  
root@debian-pc:~# systemctl start mariadb  
root@debian-pc:~# systemctl enable mariadb  
Synchronizing state of mariadb.service with SysV service script with
```

Sécuriser l'installation de mariadb et répondre aux questions.

```
mysql_secure_installation
```

```
root@debian-pc:~# systemctl start mariadb
root@debian-pc:~# systemctl enable mariadb
Synchronizing state of mariadb.service with SysV service
Executing: /lib/systemd/systemd-sysv-install enable mar
root@debian-pc:~# mysql_secure_installation_
```

Les questions sont:

- Entrer le mot de passe de root
- Switch to unix-socket -> n
- Changer le mot de passe de root -> n
- Supprimer l'utilisateur anonyme -> n
- Ne pas permettre la connexion à distance de root -> n
- Supprimer la base de donnée de test -> y
- Recharger les privilèges des tables -> y

et la configuration est terminée.

```
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
```

```
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'

Switch to unix_socket authentication [Y/n] n_
```

```
You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
```

```
go a bit smoother. You should remove them before moving into
production environment.

Remove anonymous users? [Y/n] y
```

```
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.
Disallow root login remotely? [Y/n] n

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.
Remove test database and access to it? [Y/n] n

... Skipping.

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.
Reload privilege tables now? [Y/n] y

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@debian-pc:~#
```

Il est nécessaire de configurer de la base de donnée. Créer une base de données pour wordpress
Connexion à mariadb:

```
mysql -u root -p
```

Une fois connecté, créer une base de données, un utilisateur avec les privilèges nécessaires. Entrer les lignes une par une:

```
CREATE DATABASE wordpress;

CREATE USER 'Nouas'@'localhost' IDENTIFIED BY 'Nouas710';

GRANT ALL PRIVILEGES ON wordpress.* TO 'Nouas'@'localhost';

FLUSH PRIVILEGES;

EXIT;
```

Télécharger wordpress

Aller dans le répertoire où vous souhaitez installer wordpress

```
cd /var/www/html
```

Commande pour télécharger wordpress

```
wget https://wordpress.org/latest.tar.gz
```

Extraire les fichier du .tar

```
tar -xvzf latest.tar.gz
```

Supprimer le fichier .tar.gz

```
rm latest.tar.gz
```

Donner les permission appropriées

```
chown -R www-data:www-data /var/www/html  
chmod -R 755 /var/www/html
```

A la création des fichiers wordpress, il est conseillé d'ouvrir le fichier **wp-sample-config.php** et remplacer les informations de la base de donnée puis renommer le fichier en **wp-config.php**.

Configuration d'apache

Configurer apache pour pour wordpress: Créer le fichier wordpress.conf

```
nano /etc/apache2/sites-available/wordpress.conf
```

et ajouter le contenu:

Le chemin de DocumentRoot indique le chemin vers le fichier de wordpress et a redirection ssl.

```
<VirtualHost *:80>
    Redirect permanent / https://192.168.XXX.XXX
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/wordpress
    ServerName VOTRE IP

    <Directory /var/www/html/wordpress>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Puis enregistrer et quitter avec : **Ctrl + o** et **Ctrl + x**.

Activer le site, les modules et recharger apache 2

```
a2ensite wordpress.conf

a2enmod rewrite

systemctl restart apache2
```

Le fichier 000-default.conf peut être désactivé car il est remplacé par wordpress.conf.

```
a2dissite 000-default.conf
```

Le fichier default-ssl.conf est aussi modifié : **DocumentRoot /var/www/html/wordpress**

```
nano /etc/apache2/sites-available/default-ssl.conf
```

```
<VirtualHost *:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html/wordpress_

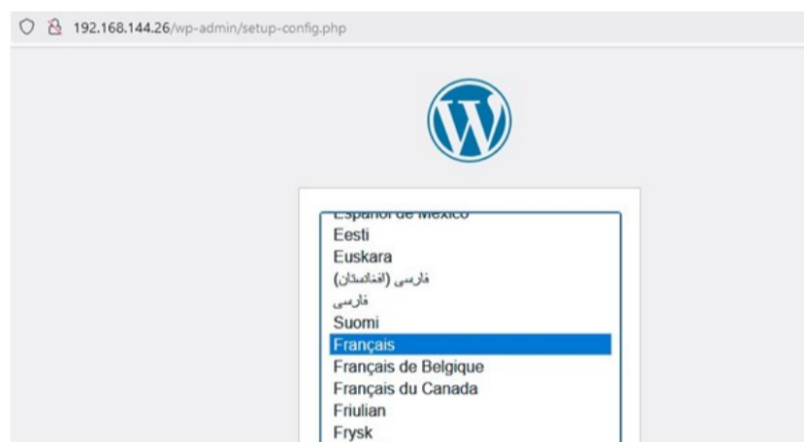
    # Available loglevels: trace8, ..., trace1, debug, info, error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. Look for
    # examples in the conf-available directory.
```

Finaliser l'installation

Dans le navigateur, taper l'adresse ip du site. La page d'installation s'affiche. La première étape est la sélection de la langue.



Puis des informations seront demandées, elles correspondent aux identifiants et mot de passe déjà mis dans les différentes étapes.



La fenêtre suivante demande d'entrer le nom de la base de donnée, utilisateur et mot de passe, adresse de la base et préfixe des tables.

Vous devez saisir ci-dessous les détails de connexion à votre base de données. Si vous ne les connaissez pas, contactez votre hébergeur.

Nom de la base de données

Nouas710

Le nom de la base de données avec laquelle vous souhaitez utiliser WordPress.

Identifiant

Nouas710

Votre identifiant MySQL.

Mot de passe

•••••

Afficher

Votre mot de passe de base de données.

Adresse de la base de données

localhost

Si localhost ne fonctionne pas, demandez cette information à l'hébergeur de votre site.

Préfixe des tables

wp_

Si vous souhaitez faire tourner plusieurs installations de WordPress sur une même base de données, modifiez ce réglage.

Envoyer



La page suivante demande: titre du site, identifiant de connexion, mot de passe et une adresss e-mail.

Informations nécessaires

Veuillez renseigner les informations suivantes. Ne vous inquiétez pas, vous pourrez les modifier plus tard.

Titre du site

Nouas710

Identifiant

Nouas710

Les identifiants ne peuvent utiliser que des caractères alphanumériques, des espaces, des tirets bas (`_`), des traits d'union (`-`), des points et le symbole `@`.

Mot de passe

!Nouas!710!!

Masquer

Forte

Important : Vous aurez besoin de ce mot de passe pour vous connecter. Pensez à le stocker dans un lieu sûr.

Votre e-mail

b@gmail.com

Vérifiez bien cette adresse e-mail avant de continuer.

Visibilité par les moteurs de recherche


☐

Demander aux moteurs de recherche de ne pas indexer ce site

Certains moteurs de recherche peuvent décider de l'indexer malgré tout.



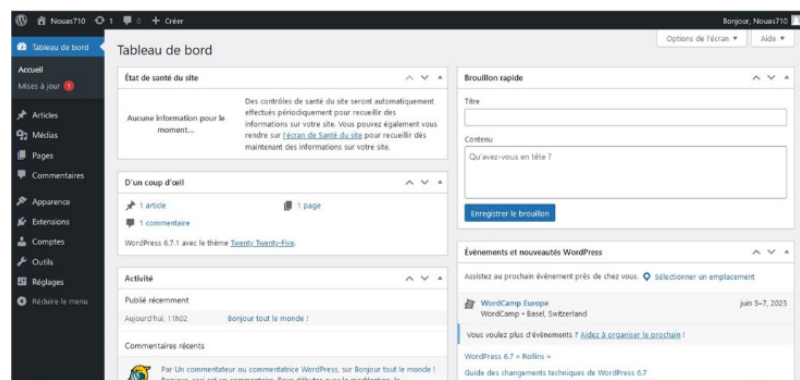
A partir de là, il est possible d'ouvrir la page de connexion de l'administration wordpress.



The image shows the WordPress login interface. At the top is the WordPress logo. Below it is a form with the following elements:

- A label "Identifiant ou adresse e-mail" above a text input field.
- A label "Mot de passe" above a password input field with an eye icon for toggling visibility.
- A checkbox labeled "Se souvenir de moi".
- A blue button labeled "Se connecter".

La page d'administration:



Pour l'ajout de page

J'ai remarqué que l'ajout de page posait un problème. Il y a un message d'erreur lors de l'enregistrement.

Dans ce cas, je propose de décommenter les lignes suivantes dans le fichier php.ini.

```
nano /etc/php/8.2/apache2/php.ini
```

```
url_rewriter.tags  
url_rewriter.hosts
```

```
; URL rewriter function rewrites URL on the fly by using  
; output buffer. You can set target tags by this configuration.  
; "form" tag is special tag. It will add hidden input tag to pass values.  
; Refer to session.trans_sid_tags for usage.  
; Default Value: "form="   
; Development Value: "form="   
; Production Value: "form="   
url_rewriter.tags  
; URL rewriter will not rewrite absolute URL nor form by default. To enable  
; absolute URL rewrite, allowed hosts must be defined at RUNTIME.  
; Refer to session.trans_sid_hosts for more details.  
; Default Value: ""   
; Development Value: ""   
; Production Value: ""   
url_rewriter.hosts  
; Transparent output compression using the zlib library  
; Valid values for this option are 'off', 'on', or a specific buffer size  
; to be used for compression (default is 4KB)  
; Note: Resulting chunk size may vary due to nature of compression. PHP  
; outputs chunks that are few hundreds bytes each as a result of  
; compression. If you prefer a larger chunk size for better
```