

GUIDE D'INSTALLATION

SAE 2.03 - INSTALLATION DE SERVICES RESEAUX

INFO
INFORMATIQUE

VICTOR GOUPIL
FEYZA TINASTEPE
MAÏLYS DEMOL-PARK
UNIVERSITE SAVOIE MONT BLANC

Préface

Welcome to this user guide on hosting a WordPress website on a RaspberryPi 3, using DietPi and Nginx.

Our goal is to help you install all the required components, with efficiency and with as much clarity as possible. This user guide has been designed to help you learn how to use the RaspberryPi computer with a 8GB microSD card.

To ensure that your system is well designed, secured, and lightweight, we suggest that you use DietPi as an operating system and NGINX as a HTTP server.

This guide contains detailed instructions, how-to tips, screenshots, and tips to walk you through the different steps. This guide is designed to be accessible to everyone, regardless of their expertise or technical background. Command-Line interface (CLI) knowledge is not necessarily needed but having a basic understanding of a few basic commands will save you some time.

Nevertheless, if you're a beginner, you'll find simple and clear information to get started quickly.

Table of Contents

Préface	1
1. Introduction	3
1.1. Purpose	3
1.2. Scope	3
1.3. System Organization	3
1.4. Technical support	4
2. Installation procedures	5
2.1. Describing the main steps	5
2.2. Before installing: Prerequisites	5
2.3. Preparing to install	Error! Bookmark not defined.
3. Installation DietPi	7
3.1 Configuration de la langue et de la région	7
3.2 Définir les logins	10
3.3 Configuration de la WiFi et l'adresse Ip	10
4. Installation Nginx	13
4.1 Configuration	13
4.2 Vérification	17
5. Installation	19
5.1 MySQL	19
5.2 WordPress	20
6. Sécurisation	25
7. Testing	30
7.1. Vérification de la connexion sur WordPress	30
8. Reference Materials (error messages, troubleshooting, specific configurations, etc.)	30
8.1 Erreur possible avec le téléchargement de DietPi:	30
8.2 Erreur possible avec l'installation de nginx:	30
9. Glossary	32
10. Index	32

1. Introduction

1.1. [Purpose of this guide](#)

This guide is designed to help you create a WordPress website, hosted on a Raspberry Pi 3 computer. It aims to provide clear guidance by offering step-by-step instructions for both installation and configuration. At the end, you will be able to create your WordPress website, relying on a robust system.

1.2. [Scope](#)

The Raspberry Pi 3 is a popular website hosting system due to its efficiency and versatility, especially matched with the DietPi operating system and the NGINX HTTP server. The RaspberryPi is also a low-cost option for those willing to self-host their websites.

DietPi is a lightweight operating system, without any graphical user interface (GUI), allowing it to be very fast, reliable and to occupy a very small portion of the 8Gb SD card.

NGINX is a lightweight HTTP server which use a very small amount of computing power. NGINX is also able to fully rely on multicores processors, such as the 4-cores CPU of the RaspberryPi 3.

1.3. [System Organization](#)

The manual is split into important sections:

- ◆ Prerequisites
- ◆ Installing and configuring DietPi
- ◆ Installing and configuring NGINX
- ◆ Installing and configuring WordPress
- ◆ Securing the Raspberry Pi and WordPress

1.4. Technical support

Technical information about Raspberry Pi and related services are available 24 hours a day online :

<https://www.raspberrypi.org/contact/>, including :

- ◆ The latest release information
- ◆ Support Forums
- ◆ Show-me tutorials
- ◆ Product documents
- ◆ Answers to frequently asked questions

For additional questions, click the Contact Support tab at the top of the page.

If your issue is urgent, please call one of the offices listed below. You will be routed to the first available technician, who will gladly assist you.

For less urgent cases, use our online Support Request Portal at [Contact us - Raspberry Pi](#)

2. Installation procedures

2.1. Describing the main steps

To set up a Raspberry Pi system with the DietPi operating system and a NGINX HTTP server, you will start by preparing the operating system. Download the DietPi OS and flash it onto a microSD card. Once done, boot up the Raspberry Pi, complete the initial setup, and connect to the internet.

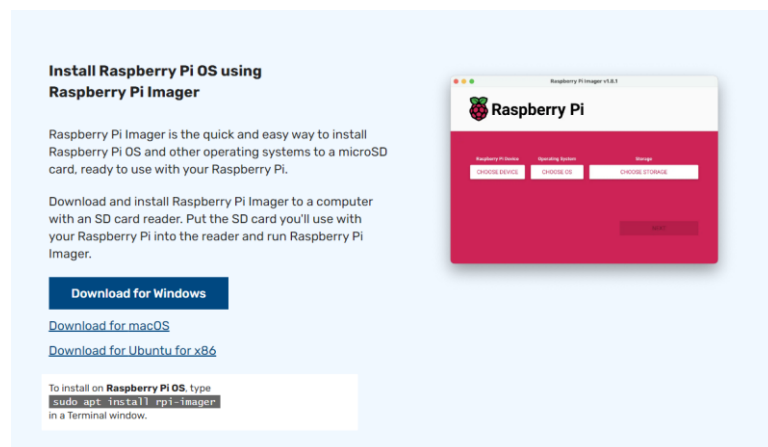
Then, install NGINX, MySQL and WordPress. After that, a few more steps are needed to secure and optimize your server and self-hosted website.

2.2. Before installing: Prerequisites

This installation guide requires the use of a Raspberry Pi model 2,3 or 4 computer. In this guide, we will be using a model 3.

- **Raspberry Pi Imager**

This software is required to flash the operating system on your SD card and is available on the official RaspberryPi website.



- **DietPi**

You will need to go on the DietPi website, scroll down to “Download”, and choose the correct OS version according to your computer’s model.



- **Other Softwares**

You will also need:

- Visual Studio Code
- Putty

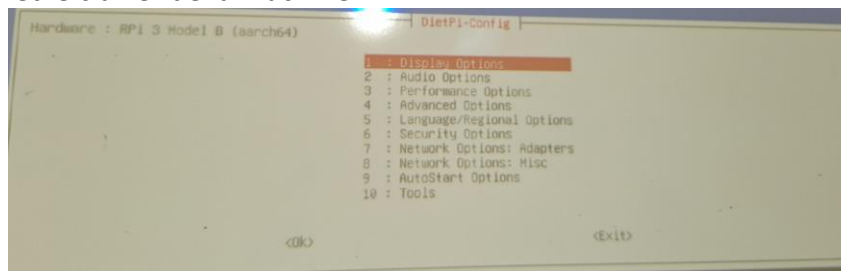
for a good installation.

The complete installation and configuration will last 1 hour 30 mins to 2 hours. An internet connection is mandatory. After plugging the SD card in the correct slot, connecting the computer to a keyboard and a monitor, you are ready to start.

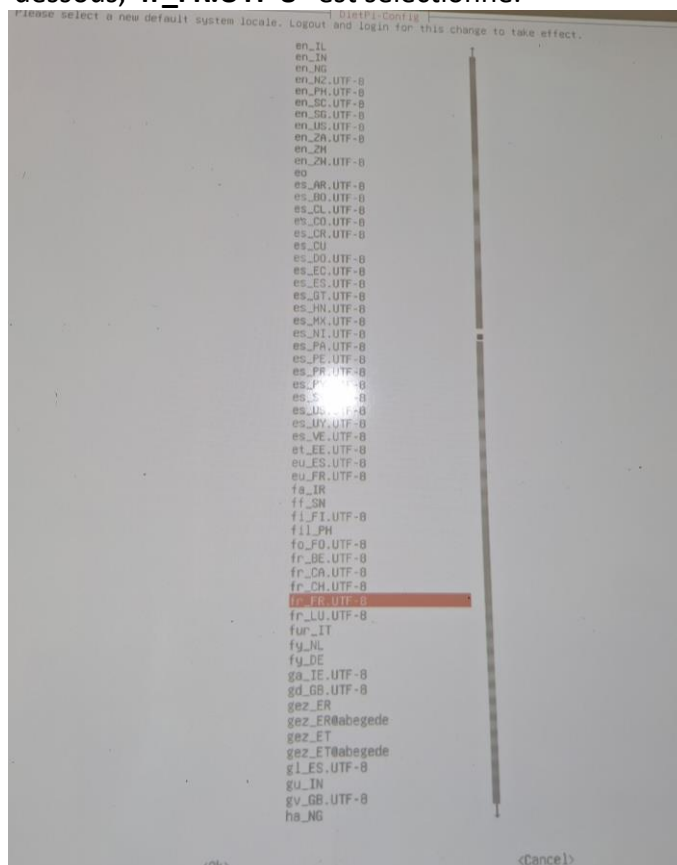
3. Installation de DietPi

3.1 Configuration de la langue et de la région

- Après la phase d'initialisation, la phase de configuration peut commencer.
- Sélectionnez l'option **"Langage/Région"** pour configurer l'heure et le clavier de la machine.



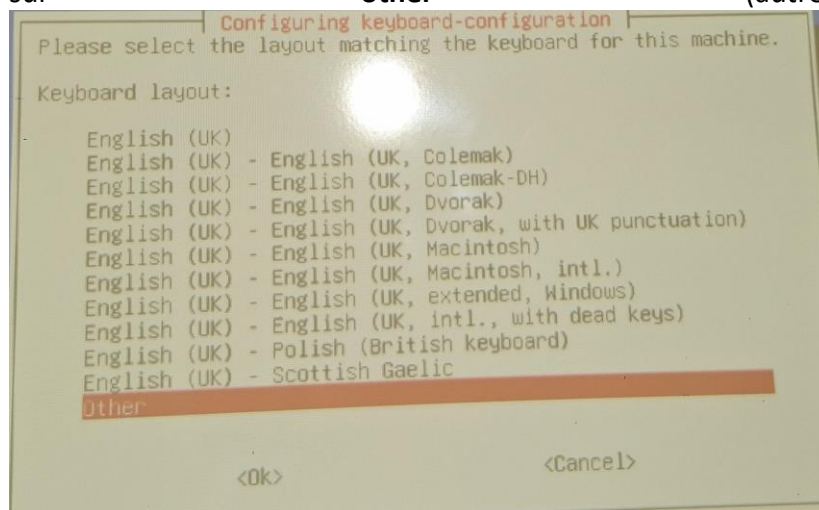
- Dans **"Locale"**, choisissez le système local qui vous convient. Ci-dessous, **"fr_FR.UTF-8"** est sélectionné.



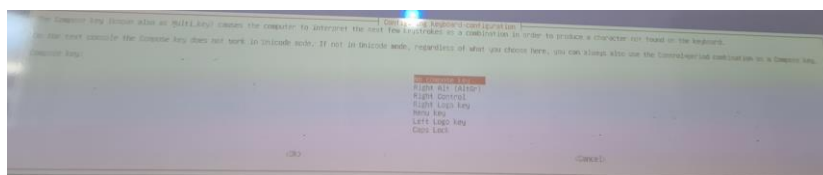
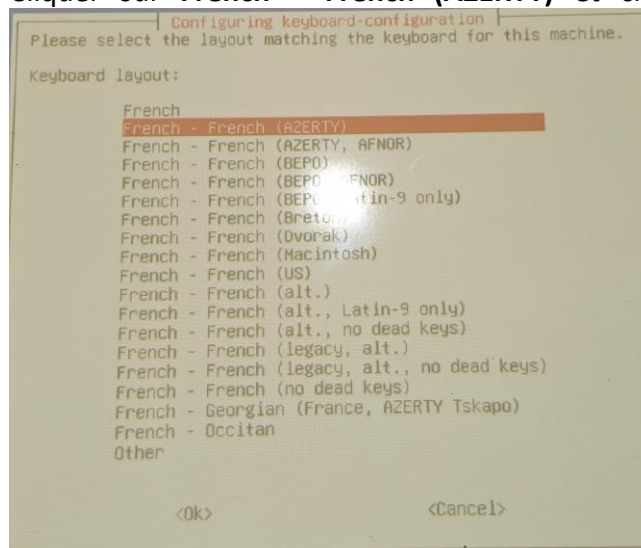
- Dans **"TimeZone"**, choisissez la zone qui vous concerne. Ci-dessous,



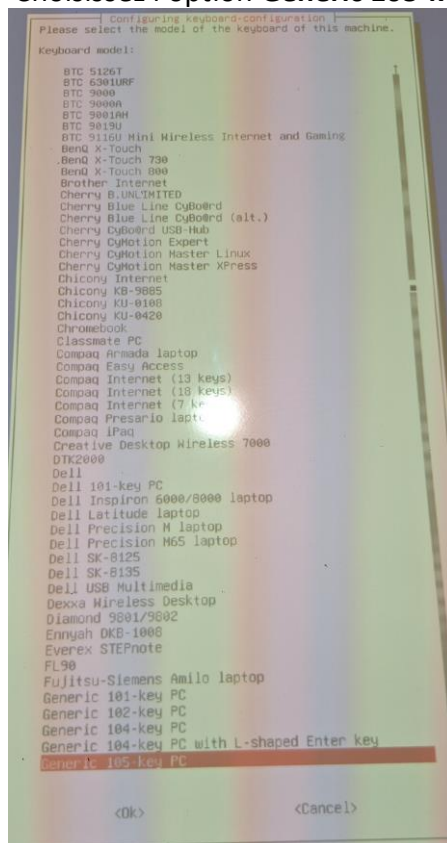
- Pour trouver la clavier Francais AZERTY, il vous suffit de cliquer sur **other** (autre)



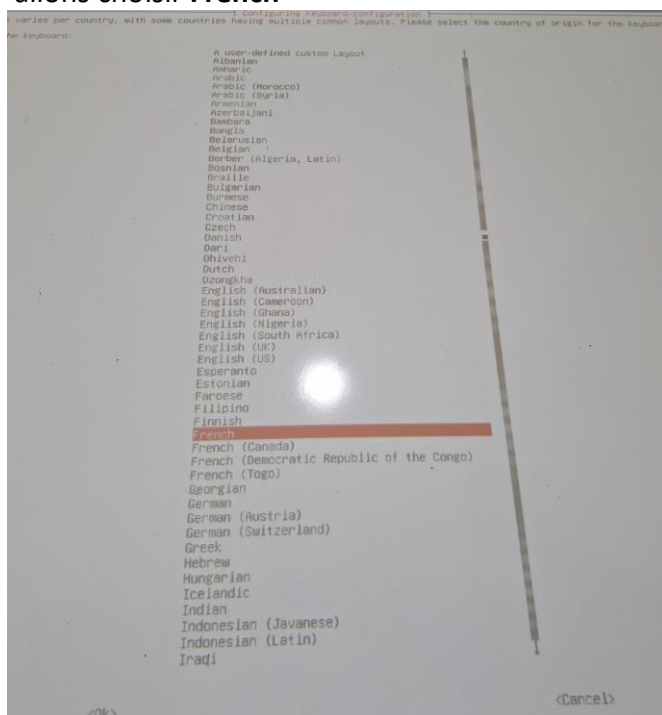
- Cliquer sur **French – French (AZERTY)** et cliquer sur **OK**.



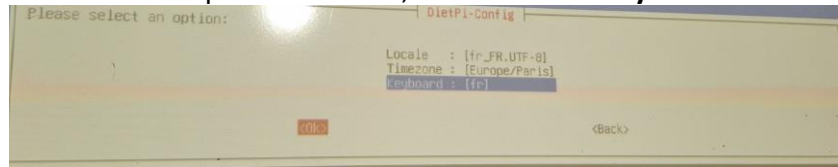
- Choisissez l'option **Generic 105-key PC**



- Choisissez votre pays pour le clavier que vous utilisez, ici nous allons choisir **French**

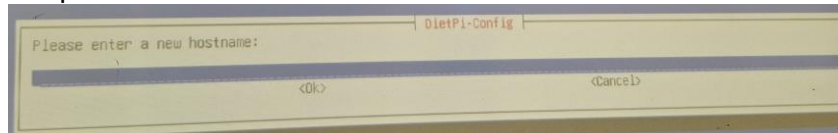


- Choisissez l'option désirée, ici donc **keyboard - fr.**

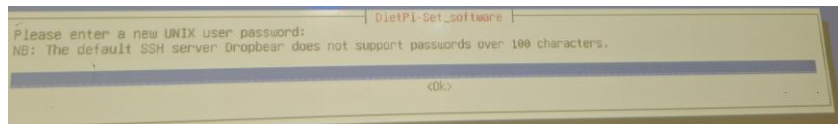


3.2 Définir les logins

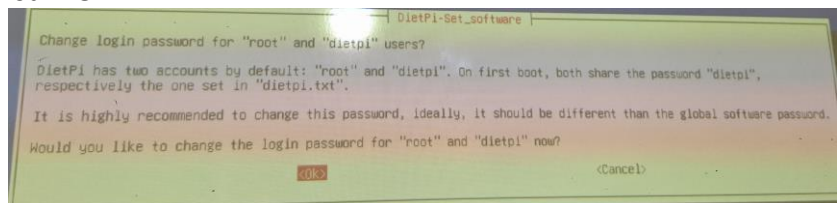
- Choisissez le nom d'utilisateur que vous souhaitez et cliquez sur OK pour le définir.



- À présent, il est nécessaire de saisir un mot de passe pour protéger votre compte UNIX, qui ne doit pas dépasser 100 caractères.

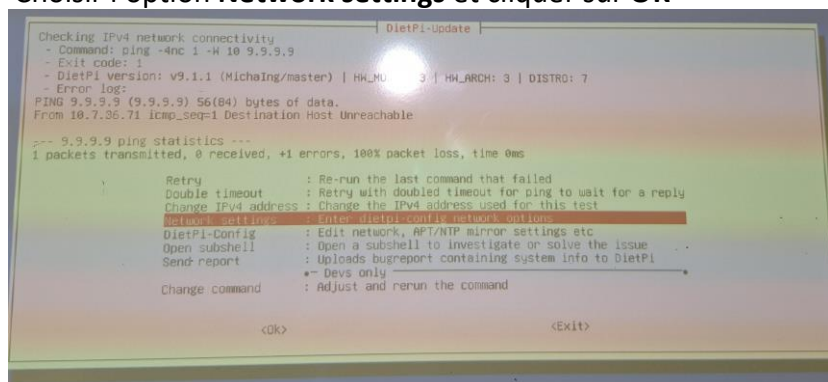


- Si tout cela a été modifié et que vous en êtes satisfait, cliquez sur "OK".

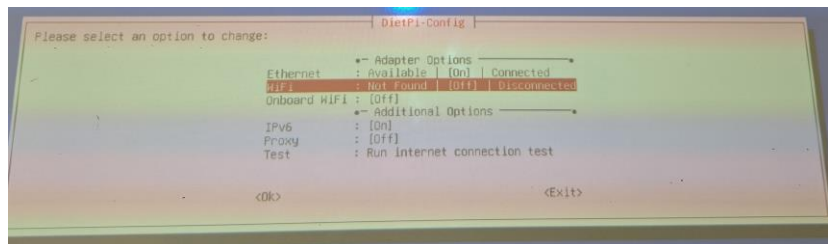


3.3 Configuration réseau

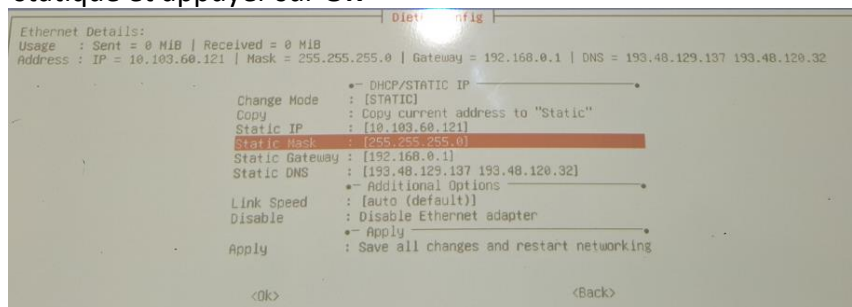
- Choisir l'option **Network settings** et cliquer sur **OK**



- Pour vous connecter paramétrer la configuration Ethernet, sélectionnez le menu « Ethernet »



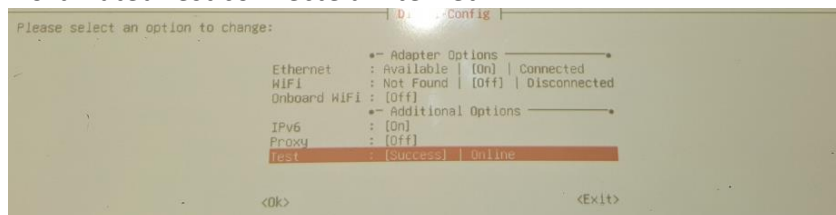
- Choisissez l'option **Static Mask** pour modifier l'adresse Ip statique et appuyer sur **OK**



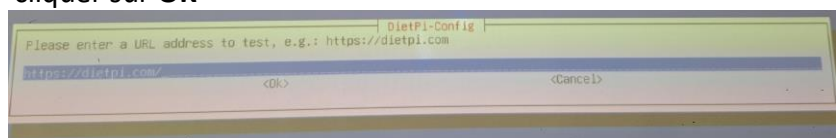
- Veuillez entrer le nouveau **adresse IP** que vous allez utiliser durant toute l'installation et cliquer sur **OK**



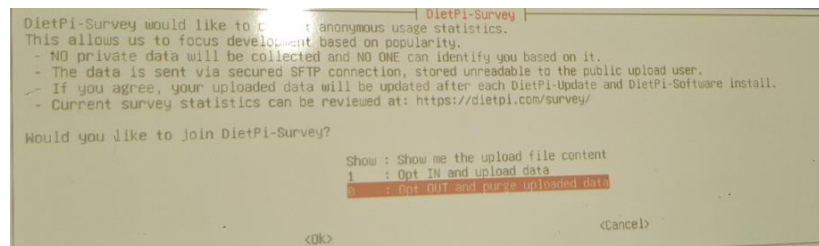
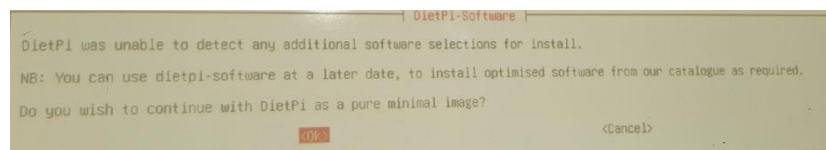
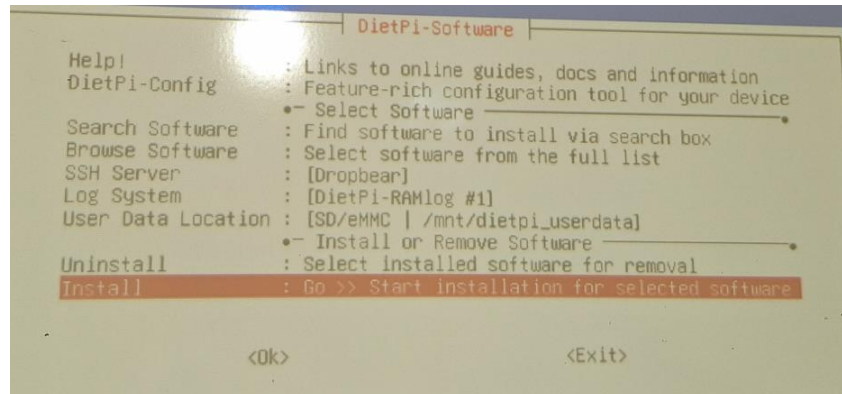
- Choisissez l'option **Test** et cliquer sur **OK**. Si le test est réussi, l'ordinateur est connecté à internet



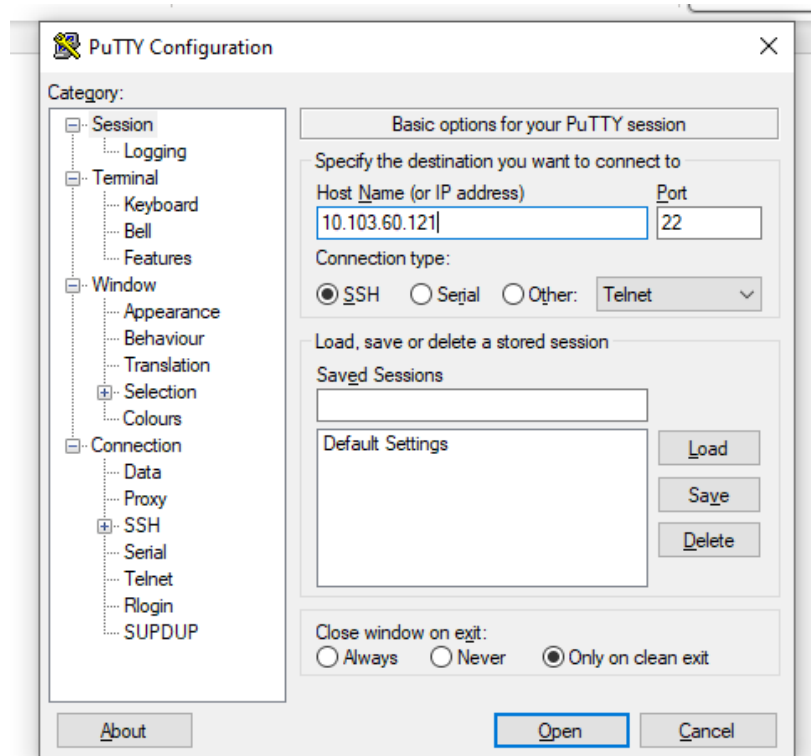
- Veuillez entrer le lien pour tester ici <https://dietpi.com> et cliquer sur **OK**



- Lancez l'installation



- Vous pouvez désormais vous connecter à Putty.



- Pour finaliser l'installation de DietPi, entrez **sudo apt-get update**.

```
dietpi@RpiSaeDGT:~$ sudo apt-get update
Atteint :1 https://deb.debian.org/debian bookworm InRelease
Atteint :2 https://deb.debian.org/debian bookworm-updates InRelease
Atteint :3 https://deb.debian.org/debian-security bookworm-security InRelease
Atteint :4 https://deb.debian.org/debian bookworm-backports InRelease
Atteint :5 https://archive.raspberrypi.com/debian bookworm InRelease
[?]ecture des listes de paquets... 0%
```

- Puis **sudo apt-get upgrade**.

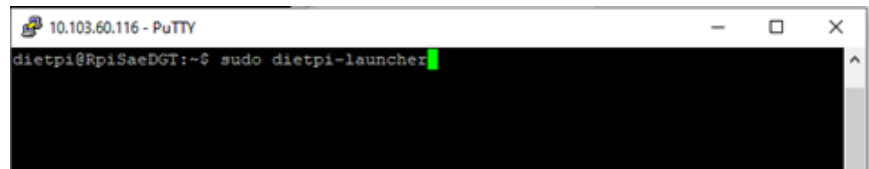
10.103.60.116 - PuTTY

```
dietpi@RpiSaeDGT:~$ sudo apt-get upgrade
[?]ecture des listes de paquets... 0%
```

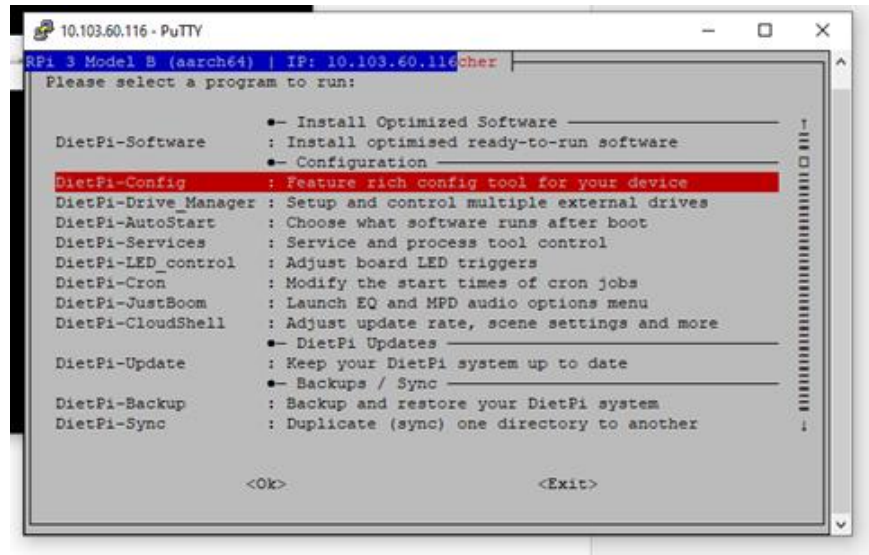
4. Installation de Nginx

4.1 Configuration

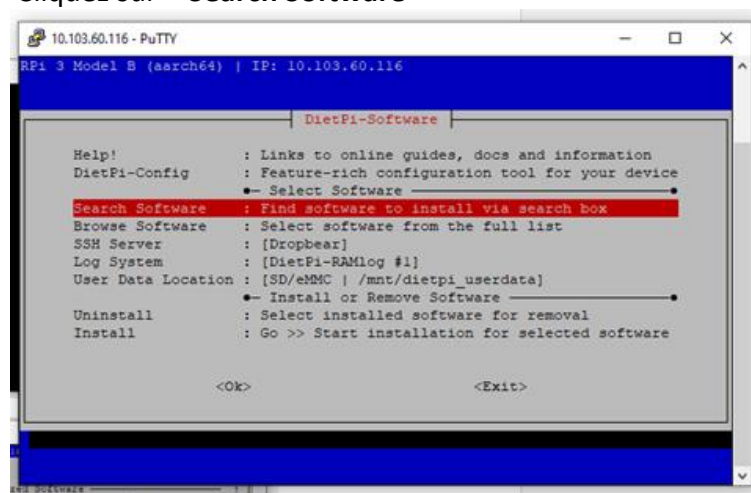
- Afin de consulter l'interface graphique du lanceur DietPi, veuillez saisir cette instruction : **sudo dietpi-laucher** sur Putty



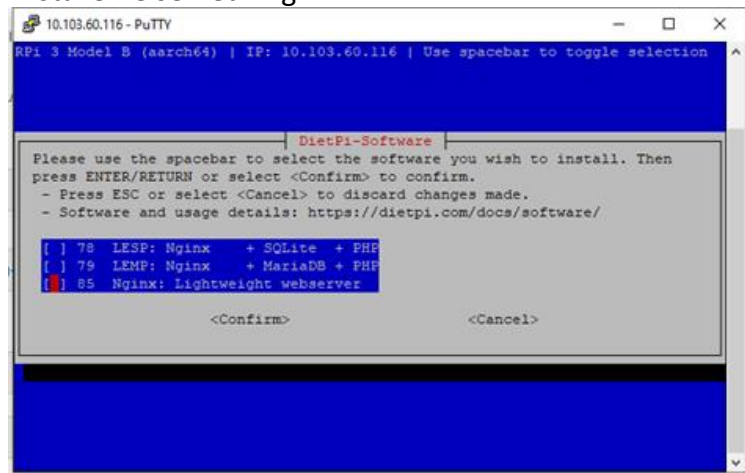
- Veuillez sélectionner un programme à exécuter (Ici nous choisissons **Dietpi – Config**)



- Cliquez sur « **Search Software** »

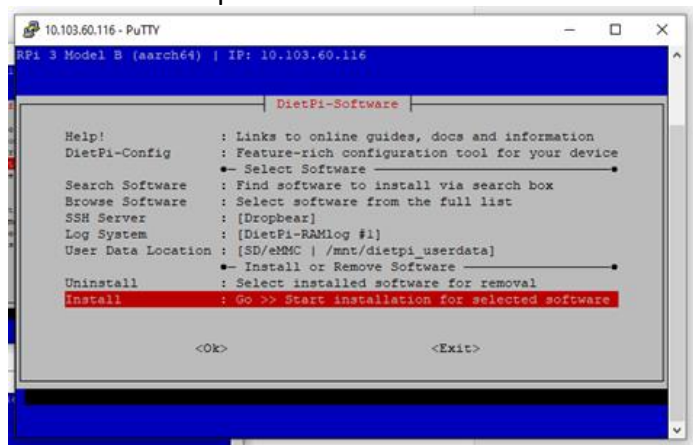


- Veuillez sélectionner le serveur que vous souhaitez installer en utilisant la touche "**espace**" de votre clavier. Ici nous souhaitons installer le serveur Nginx.

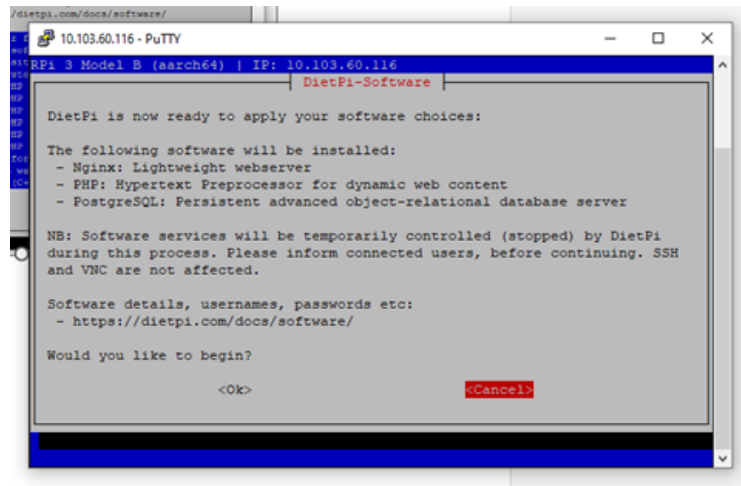


- Répéter la recherche pour l'installation de PHP

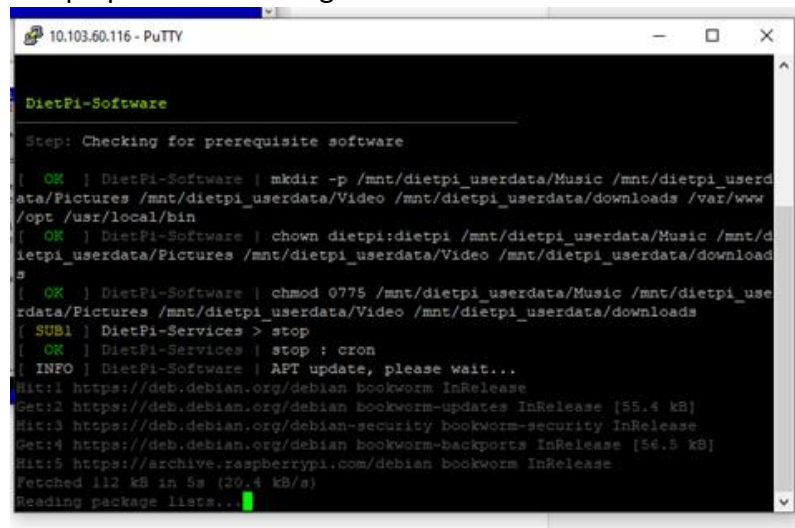
- Sélectionnez l'option **Install** afin de débiter l'installation.



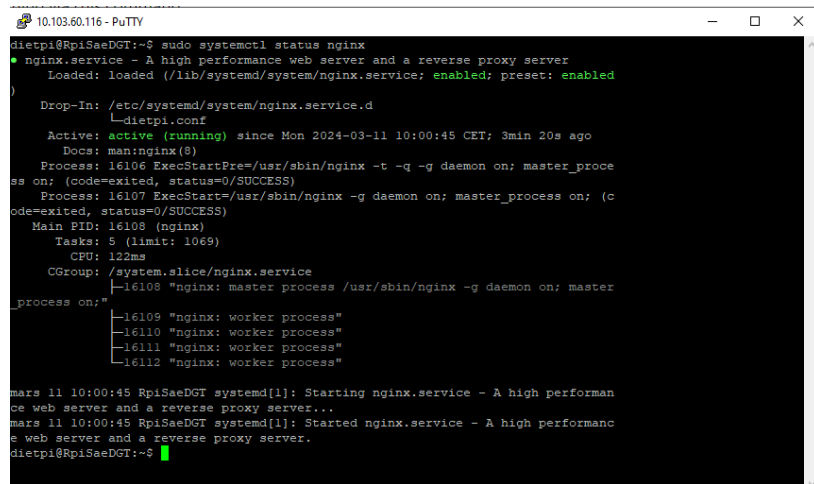
- Cliquer sur **OK** pour commencer l'utilisation



- Les paquets se téléchargent.



- Utilisez la commande **sudo systemctl status nginx** afin de contrôler le bon fonctionnement du serveur NGINX



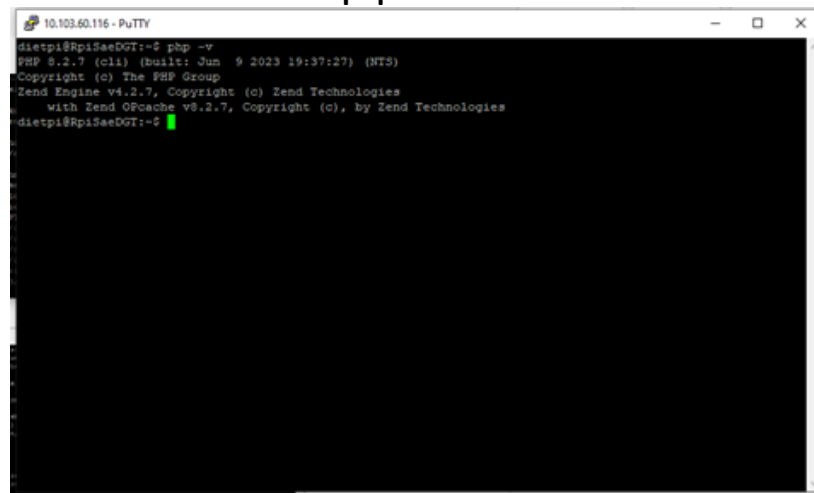
```

10.103.60.116 - PuTTY
dietpi@RpiSaeDGT:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Drop-In: /etc/systemd/system/nginx.service.d
            └─dietpi.conf
   Active: active (running) since Mon 2024-03-11 10:00:45 CET; 3min 20s ago
     Docs: man:nginx(8)
   Process: 16106 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 16107 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 16108 (nginx)
    Tasks: 5 (limit: 1069)
       CPU: 122ms
   CGroup: /system.slice/nginx.service
           └─16108 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
               └─16109 "nginx: worker process"
                   └─16110 "nginx: worker process"
                       └─16111 "nginx: worker process"
                           └─16112 "nginx: worker process"

mars 11 10:00:45 RpiSaeDGT systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
mars 11 10:00:45 RpiSaeDGT systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
dietpi@RpiSaeDGT:~$

```

- Veuillez écrire cette commande pour vérifier l'installation de PHP voici la commande : **php-v**



```

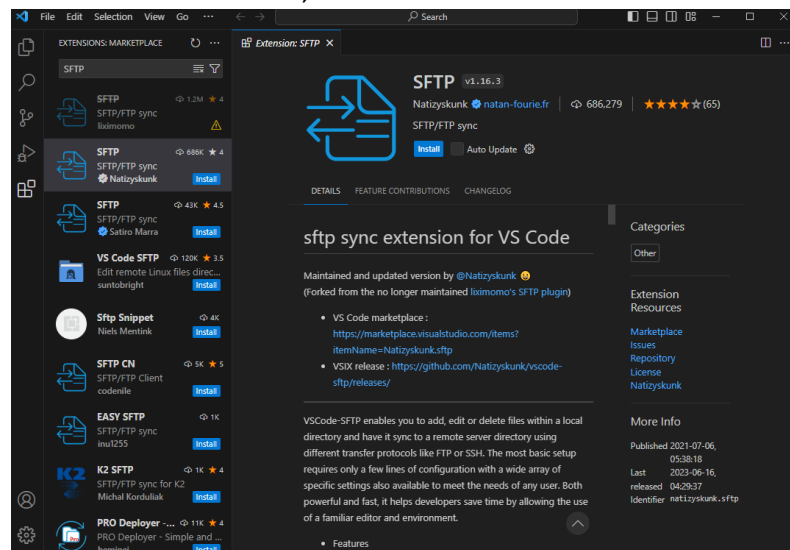
10.103.60.116 - PuTTY
dietpi@RpiSaeDGT:~$ php -v
PHP 8.2.7 (cli) (built: Jun  9 2023 19:37:27) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.2.7, Copyright (c) Zend Technologies
    with Zend OPcache v8.2.7, Copyright (c), by Zend Technologies
dietpi@RpiSaeDGT:~$

```

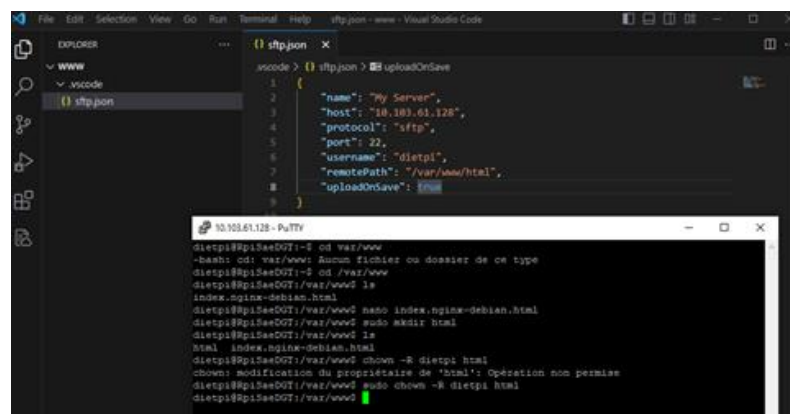
4.2 Mise en place du serveur distant et vérification NGINX

- Sur votre ordinateur Windows, créez un répertoire **www**.

- Sur Visual Studio Code, installez l'extension SFTP



- Placez-vous sur votre répertoire, appuyez sur **shift+ctrl+p** puis entrez **SFTP : Config**.
- Changez :
 - L'adresse hôte
 - Le nom
 - RemotePath en **/var/www/html**
 - UploadOnSave en **true**
 - Le port (SSH) s'il est modifié (voir partie 6 – Sécurisation)



- Les fichiers configurés ou créés dans ce répertoire seront copiés sur votre RaspberryPi, au chemin RemotePath spécifié

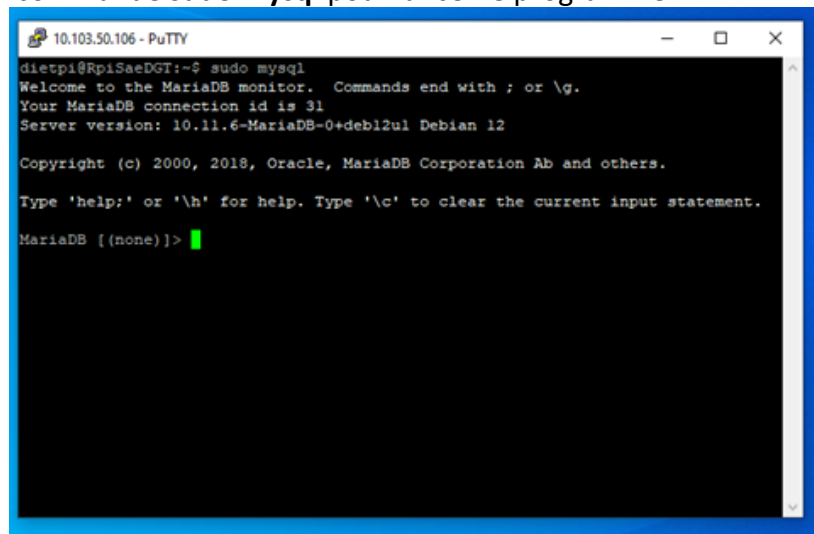
- Pour vérifier l'accès au serveur NGINX depuis un ordinateur situé sur le même réseau local, rendez vous sur un navigateur et entrez l'adresse IP statique de votre RaspberryPi



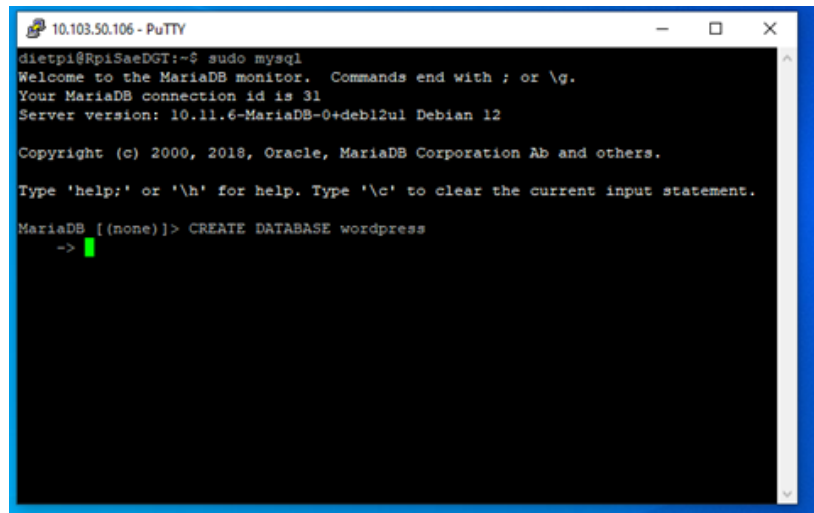
5. Installation de MySQL

5.1 MySQL

- MySQL est un système de base de données, nécessaire au bon fonctionnement de WordPress. Après l'avoir installé en utilisant la commande **sudo apt-get install mariadb**, entrez la commande **sudo mysql** pour lancer le programme.



- Ecrivez **CREATE DATABASE wordpress** pour commencer à créer votre base nécessaire au site WordPress



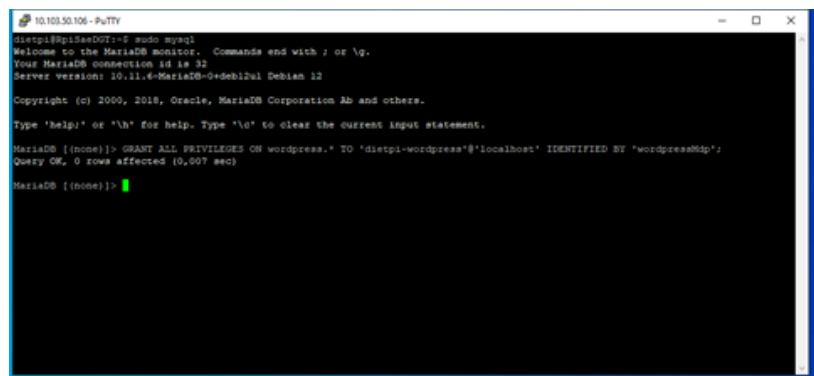
```
10.103.50.106 - PuTTY
dietpi@RpiSaeDGT:~$ sudo mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE wordpress
->
```

- Ecrivez **GRANT ALL PRIVILEGES ON wordpress.* TO 'dietpi-wordpress'@'localhost' IDENTIFIED BY 'wordpressMdp';**



```
10.103.50.106 - PuTTY
dietpi@RpiSaeDGT:~$ sudo mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> GRANT ALL PRIVILEGES ON wordpress.* TO 'dietpi-wordpress'@'localhost' IDENTIFIED BY 'wordpressMdp';
Query OK, 0 rows affected (0.007 sec)

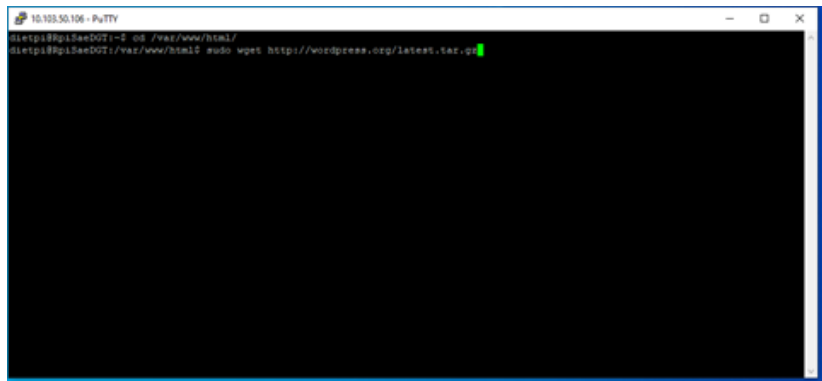
MariaDB [(none)]>
```

- Ecrivez **Sudo systemctl restart nginx** pour redémarrer le serveur nginx

5.2 Installation de WordPress

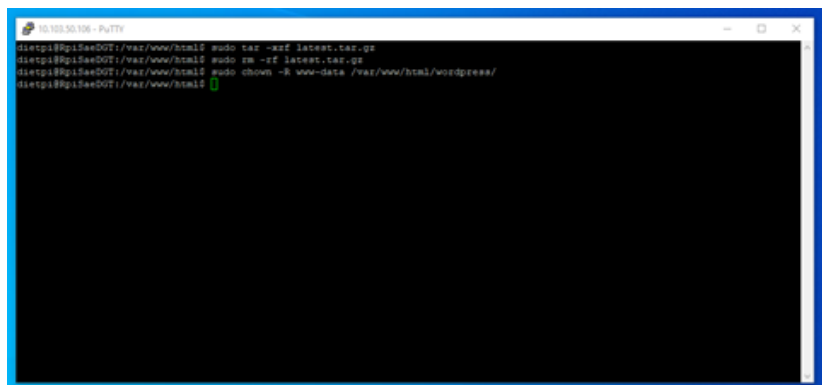
- Entrez **cd /var/www/html/**

- Entrez par la suite :
sudo wget <https://wordpress.org/latest.tar.gz>
afin de télécharger la dernière version de WordPress



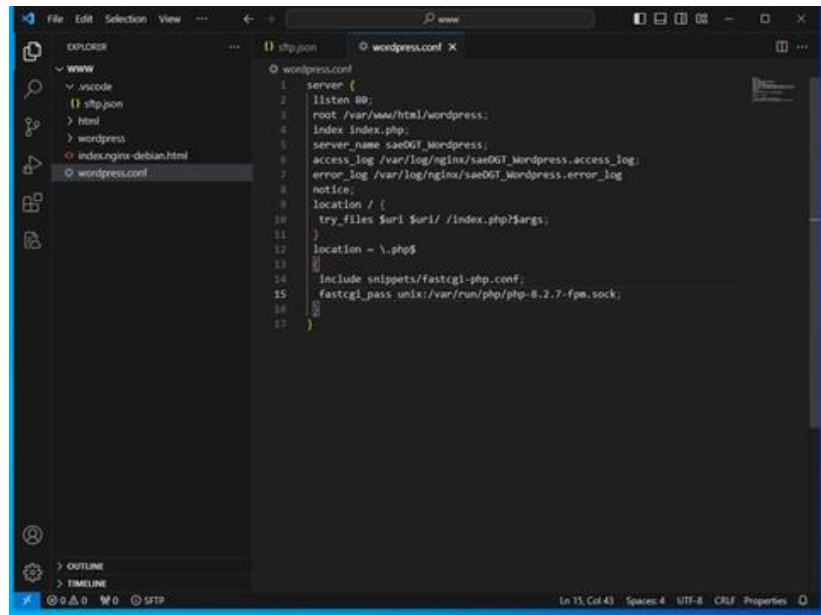
```
10.103.50.106 - PuTTY
dietpi@dietpi:~$ cd /var/www/html/
dietpi@dietpi:~/var/www/html$ sudo wget https://wordpress.org/latest.tar.gz
```

- Commencez par écrire cette commande **sudo tar -xzf latest.tar.gz**
- Puis **sudo rm -rf latest.tar.gz**
- Terminé par écrire ça : **sudo chown -R www-data /var/www/html/wordpress/**



```
10.103.50.106 - PuTTY
dietpi@dietpi:~/var/www/html$ sudo tar -xzf latest.tar.gz
dietpi@dietpi:~/var/www/html$ sudo rm -rf latest.tar.gz
dietpi@dietpi:~/var/www/html$ sudo chown -R www-data /var/www/html/wordpress/
dietpi@dietpi:~/var/www/html$
```

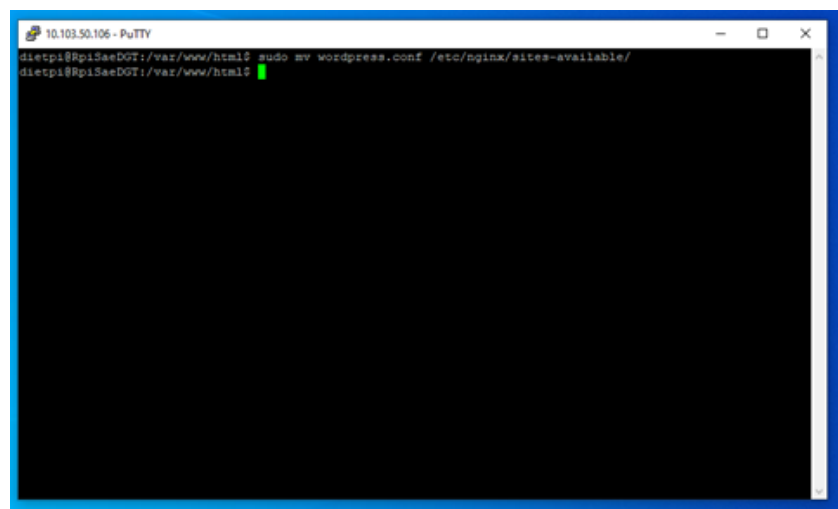
- Grâce au partage SFTP sur Visual Studio Code, vous pouvez **configurer** votre site WordPress comme ceci en créant un fichier **wordpress.conf**:



The screenshot shows the Visual Studio Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a directory structure with files like .vscode, .htp.json, html, wordpress, index.nginx-debian.html, and wordpress.conf. The code editor displays the contents of the wordpress.conf file, which is an Nginx configuration snippet for WordPress. The configuration includes a server block listening on port 80, with the root directory set to /var/www/html/wordpress. It also includes a location block for /index.php that uses the try_files directive to serve the index.php file or redirect to /index.php/\$args. The configuration includes the fastcgi_pass directive to connect to the PHP-FPM socket.

```
1 server {
2     listen 80;
3     root /var/www/html/wordpress;
4     index index.php;
5     server_name sae0GT_Wordpress;
6     access_log /var/log/nginx/sae0GT_Wordpress.access_log;
7     error_log /var/log/nginx/sae0GT_Wordpress.error_log;
8     notice;
9     location / {
10         try_files $uri $uri/ /index.php?$args;
11     }
12     location ~ \.php$
13     {
14         include snippets/fastcgi-php.conf;
15         fastcgi_pass unix:/var/run/php/php-8.2.7-fpm.sock;
16     }
17 }
```

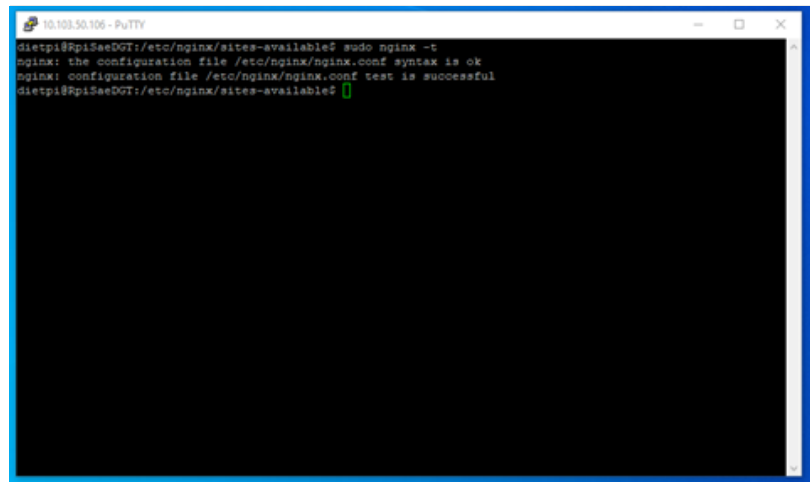
- Écrivez à présent **sudo mv wordpress.conf /etc/nginx/sites-available/**



The screenshot shows a terminal window with the command `sudo mv wordpress.conf /etc/nginx/sites-available/` being executed. The prompt is `dietspi@RpiSae0GT: /var/www/html$`. The command is successful, and the prompt returns to `dietspi@RpiSae0GT: /var/www/html$`.

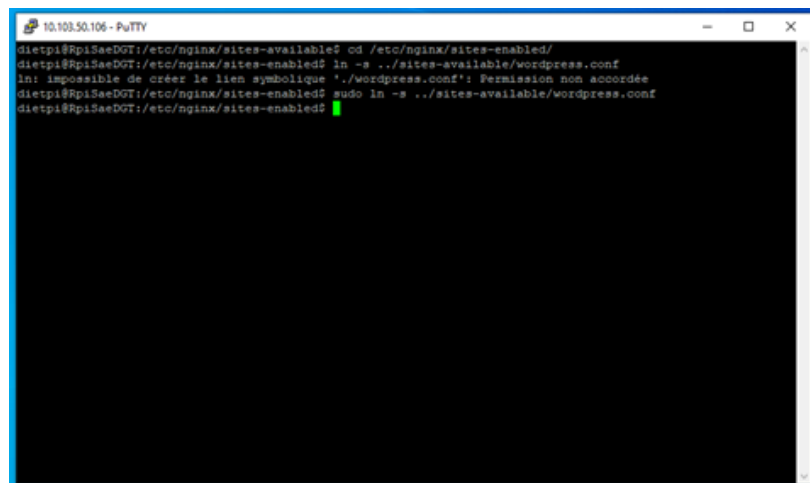
```
dietspi@RpiSae0GT: /var/www/html$ sudo mv wordpress.conf /etc/nginx/sites-available/
dietspi@RpiSae0GT: /var/www/html$
```

- Déplacez-vous au sein de ce répertoire, puis entrez **sudo nginx -t** afin de vérifier la conformité du fichier .conf



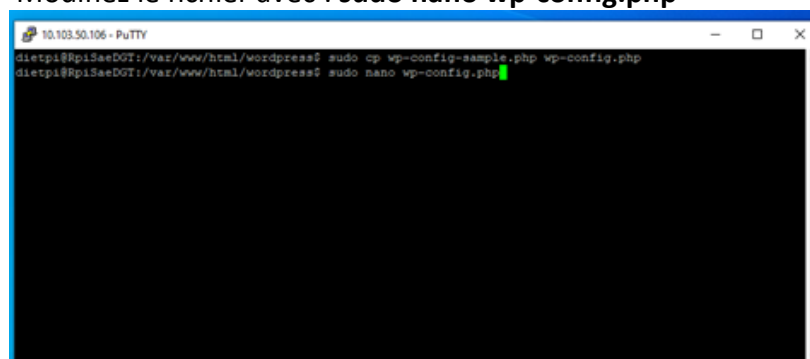
```
10.103.50.106 - PuTTY
dietpi@RpiSaeDGT:/etc/nginx/sites-available$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
dietpi@RpiSaeDGT:/etc/nginx/sites-available$
```

- Cette série de commande permet de créer un lien entre deux répertoires permettant le déploiement du site WordPress 1:
**cd /etc/nginx/sites-enabled/
sudo ln -s ../sites-available/wordpress.conf**



```
10.103.50.106 - PuTTY
dietpi@RpiSaeDGT:/etc/nginx/sites-available$ cd /etc/nginx/sites-enabled/
dietpi@RpiSaeDGT:/etc/nginx/sites-enabled$ ln -s ../sites-available/wordpress.conf
ln: impossible de créer le lien symbolique '../wordpress.conf': Permission non accordée
dietpi@RpiSaeDGT:/etc/nginx/sites-enabled$ sudo ln -s ../sites-available/wordpress.conf
dietpi@RpiSaeDGT:/etc/nginx/sites-enabled$
```

- Retournez dans le répertoire /var/www/html/wordpress puis entrez la commande suivante :
sudo cp wp-config-sample.php wp-config.php
- Modifiez le fichier avec : **sudo nano wp-config.php**

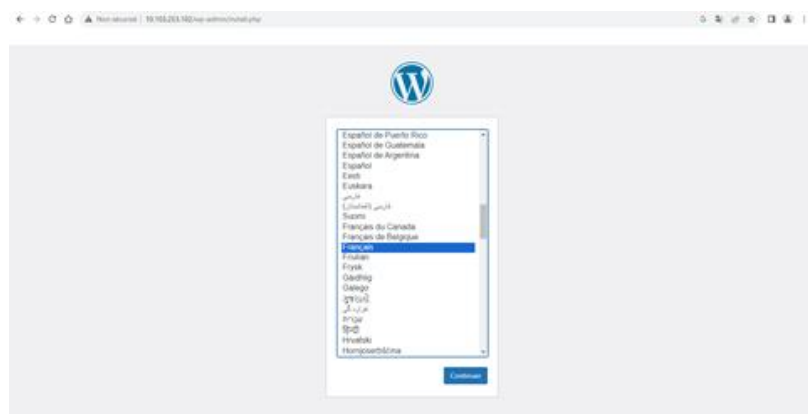


```
10.103.50.106 - PuTTY
dietpi@RpiSaeDGT:/var/www/html/wordpress$ sudo cp wp-config-sample.php wp-config.php
dietpi@RpiSaeDGT:/var/www/html/wordpress$ sudo nano wp-config.php
```


- Éditez ce fichier avec les informations fournies lors de la création de la base de données MySQL :

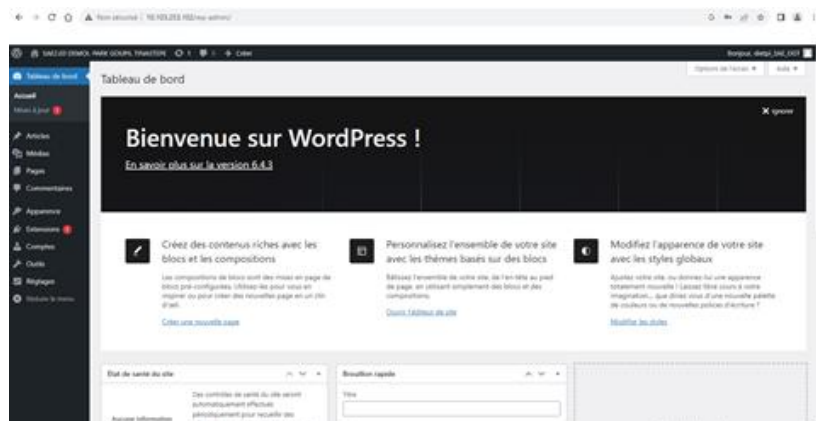
```
10.101.50.106 - PuTTY
diepi@pi540DT:/etc/nginx/sites-available$ sudo systemctl restart nginx
diepi@pi540DT:/etc/nginx/sites-available$ cd /var/www/html/
diepi@pi540DT:/var/www/html$ sudo nano wp-config.php
diepi@pi540DT:/var/www/html$ ls
html index.nginx-debian.html wordpress
diepi@pi540DT:/var/www/html$ cd wordpress/
diepi@pi540DT:/var/www/html/wordpress$ ls
index.php readme.html wp-admin wp-comments-post.php wp-config-sample.php wp-cron.php wp-links-opml.php wp
license.txt wp-activate.php wp-blog-header.php wp-config.php wp-content wp-includes wp-load.php wp
diepi@pi540DT:/var/www/html/wordpress$ nano wp-config.php
GNU nano 2.2 wp-config.php
define( 'DB_USER', 'diepi-wordpress' );
/** Database password */
define( 'DB_PASSWORD', 'wordpressMq' );
/** Database hostname */
define( 'DB_HOST', 'localhost:3306' );
/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );
/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );
/**#@+
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
```

- Redémarrez NGINX, puis connectez-vous en utilisant l'adresse IP statique de l'ordinateur dans la barre de recherche URL
- Choisissez la langue



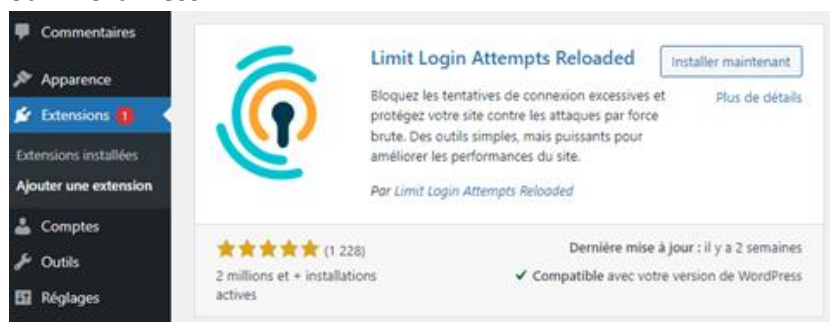
- Afin d'accéder au site, veuillez fournir votre identifiant mot de passe WordPress ainsi qu'un nom de site, par exemple.

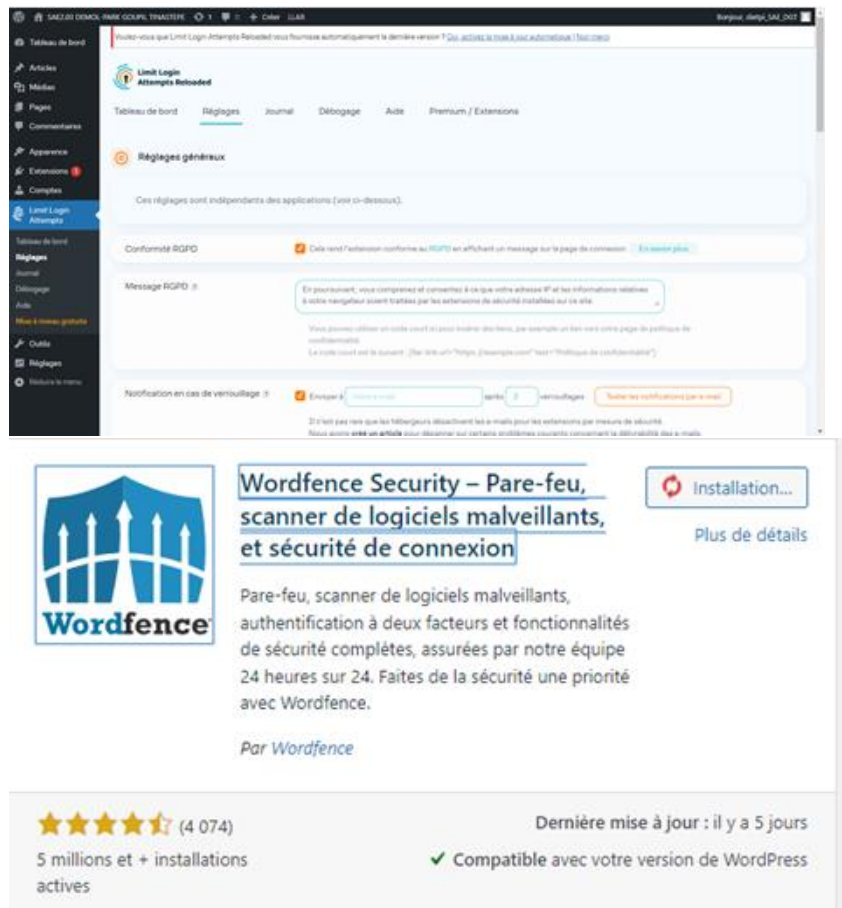
- Maintenant, vous avez la possibilité de créer votre site web sur WordPress.



6. Sécurisation

Sur WordPress





SAE2.03 DEMOL-PARK GOUPIL TINASTEPE 1 Color LAB

Limit Login Attempts Reloaded

Tableau de bord Réglages Journal Débogage Aide Premium / Extensions

Réglages généraux

Ces réglages sont indépendants des applications (voir ci-dessous).

Conformité RGPD ☒ Cela rend l'extension conforme au RGPD en affichant un message sur la page de connexion. [En savoir plus](#)

Message RGPD :


En poursuivant, vous acceptez et consentez à ce que votre adresse IP et les informations relatives à votre navigateur soient traitées par les extensions de sécurité installées sur ce site.

Vous pouvez utiliser un code court ci-dessous pour insérer des liens, par exemple un lien vers votre page de politique de confidentialité. Le code court est le suivant : `[<code>[?src=wp-https://example.com/?src=WordPress de confidentialité]]</code>`

Notification en cas de verrouillage :

Envoyer à : nombre : verrouillages. [Configurer les notifications par e-mail](#)

Il n'est pas rare que les utilisateurs bloquent les e-mails pour les extensions par mesure de sécurité. Nous avons écrit un article pour discuter sur certains problèmes courants concernant la délivrance des e-mails.



Wordfence Security – Pare-feu, scanner de logiciels malveillants, et sécurité de connexion

Pare-feu, scanner de logiciels malveillants, authentification à deux facteurs et fonctionnalités de sécurité complètes, assurées par notre équipe 24 heures sur 24. Faites de la sécurité une priorité avec Wordfence.

Par Wordfence

Installation...

[Plus de détails](#)

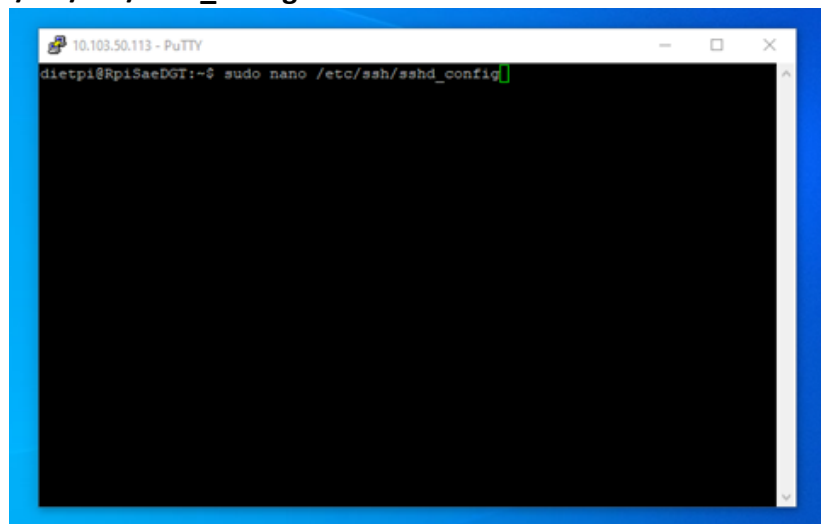
★★★★★ (4 074)

5 millions et + installations actives

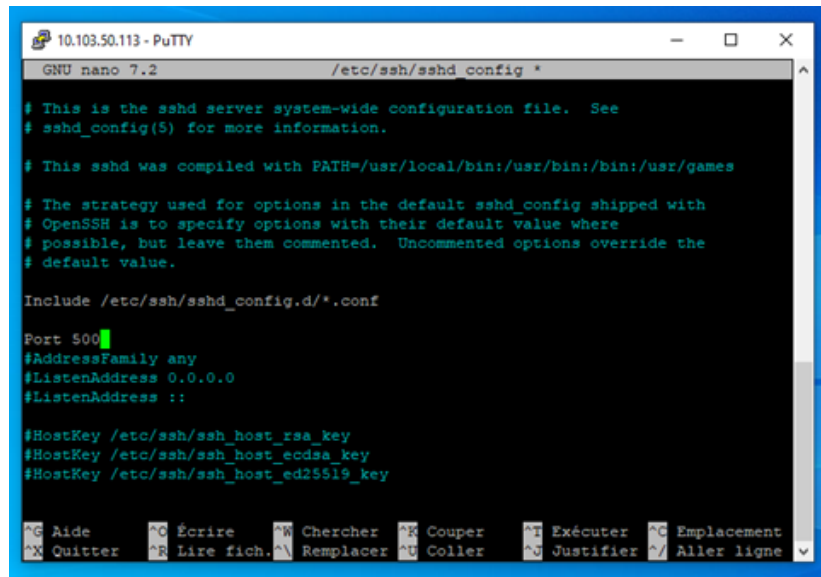
Dernière mise à jour : il y a 5 jours

✓ Compatible avec votre version de WordPress

- Sur Putty, entrez la commande **sudo nano /etc/ssh/sshd_config**



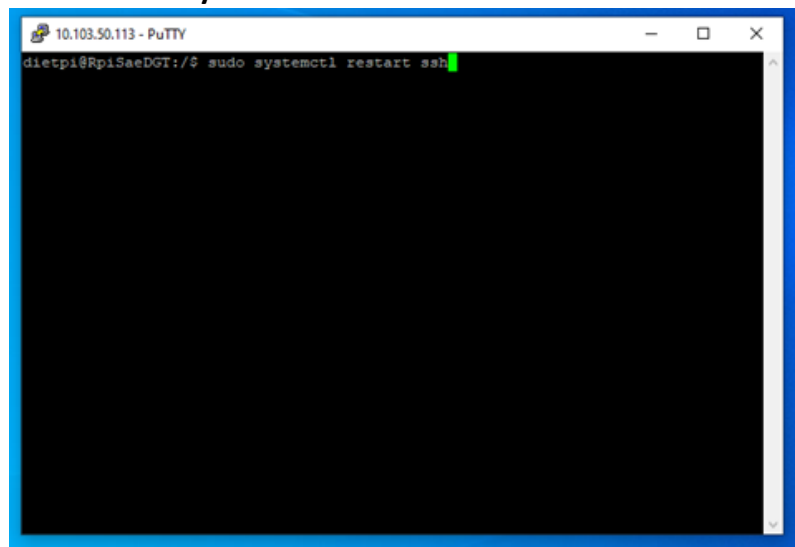
```
10.103.50.113 - PuTTY
dietpi@RpiSaeDGT:~$ sudo nano /etc/ssh/sshd_config
```



A terminal window titled "10.103.50.113 - PuTTY" showing the contents of the `/etc/ssh/sshd_config` file. The editor is GNU nano 7.2. The file contains configuration for the SSH daemon, including comments about the system-wide configuration, compilation path, and strategy. It includes a line `Include /etc/ssh/sshd_config.d/*.conf` and sets `Port 500`. Other options like `#AddressFamily any`, `#ListenAddress 0.0.0.0`, and host keys are also visible. A status bar at the bottom shows various keyboard shortcuts.

```
GNU nano 7.2 /etc/ssh/sshd_config *
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.
Include /etc/ssh/sshd_config.d/*.conf
Port 500
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
^G Aide ^O Écrire ^W Chercher ^K Couper ^T Exécuter ^C Emplacement
^X Quitter ^R Lire fich. ^\ Remplacer ^U Coller ^J Justifier ^_/ Aller ligne
```

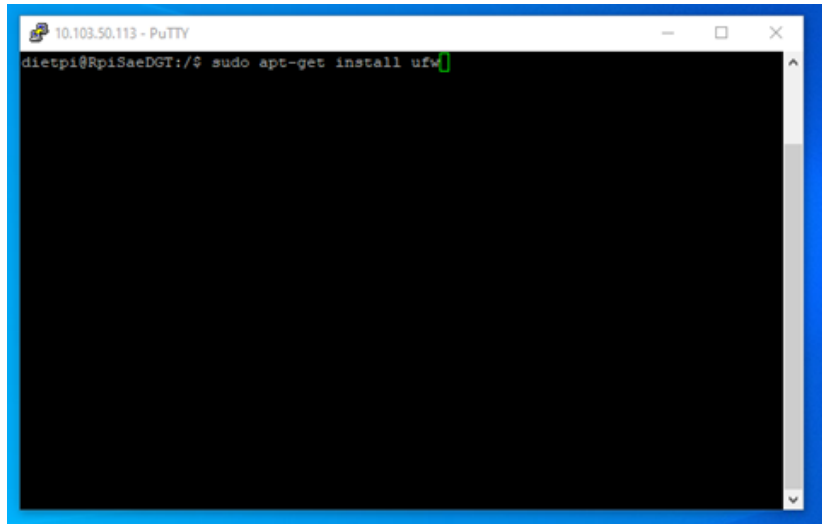
- Écrivez **sudo systemctl restart ssh**



A terminal window titled "10.103.50.113 - PuTTY" showing the command `sudo systemctl restart ssh` being entered at the prompt `dietpi@RpiSaeDGT:/$`. The cursor is at the end of the command.

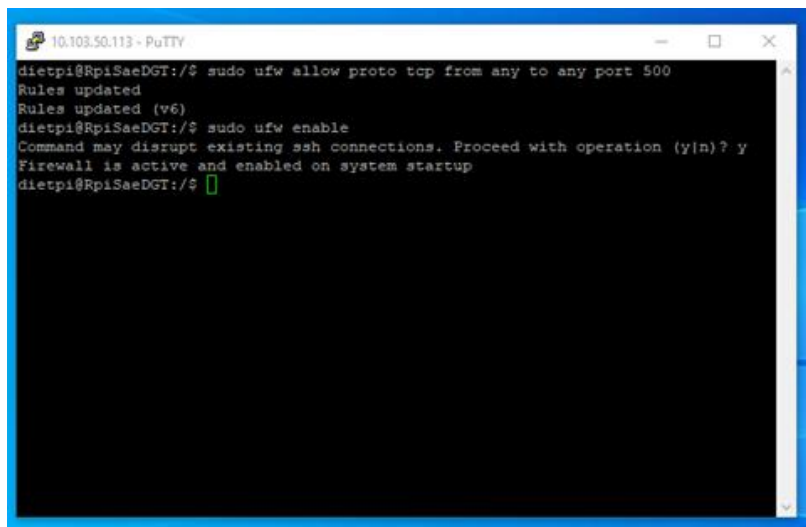
```
10.103.50.113 - PuTTY
dietpi@RpiSaeDGT:/$ sudo systemctl restart ssh
```

- Écrivez **sudo apt-get install ufw**

A terminal window titled '10.103.50.113 - PuTTY' with a black background and white text. The prompt is 'dietpi@RpiSaeDGT:/\$' and the command 'sudo apt-get install ufw' is being entered, with a green cursor at the end of the line.

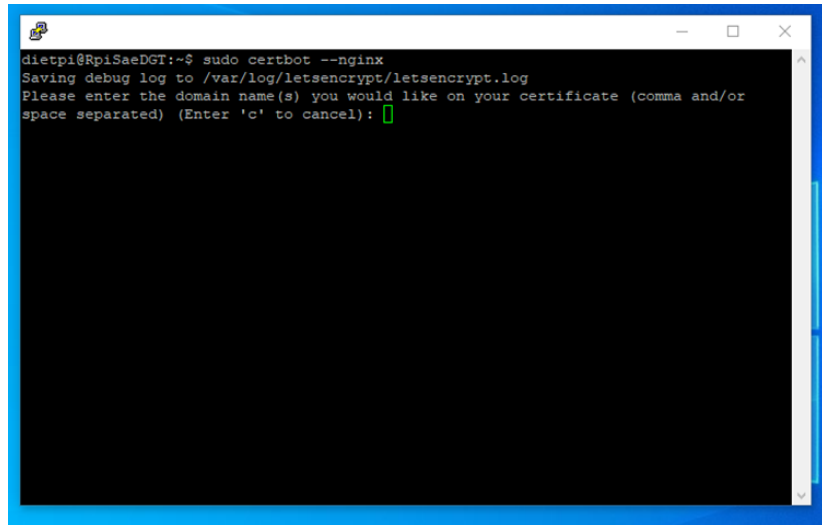
```
10.103.50.113 - PuTTY
dietpi@RpiSaeDGT:/$ sudo apt-get install ufw
```

- Écrivez **sudo ufw allow proto tcp from any to any port 500**
- Puis **Sudo ufw enable**

A terminal window titled '10.103.50.113 - PuTTY' with a black background and white text. It shows the execution of two commands: 'sudo ufw allow proto tcp from any to any port 500' and 'sudo ufw enable'. The output shows 'Rules updated' and a confirmation prompt for enabling the firewall, which is answered with 'y'.

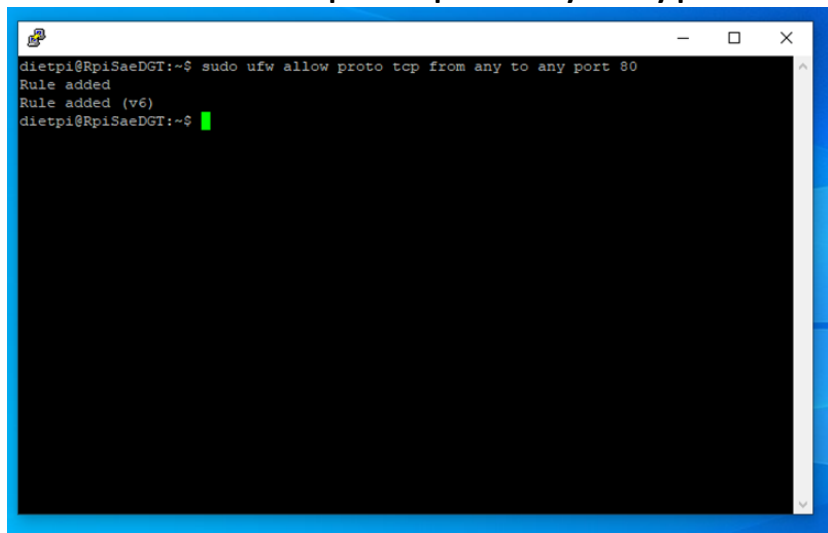
```
10.103.50.113 - PuTTY
dietpi@RpiSaeDGT:/$ sudo ufw allow proto tcp from any to any port 500
Rules updated
Rules updated (v6)
dietpi@RpiSaeDGT:/$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y/n)? y
Firewall is active and enabled on system startup
dietpi@RpiSaeDGT:/$
```

- Écrivez **sudo certbot --nginx**



```
dietpi@RpiSaeDGT:~$ sudo certbot --nginx
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Please enter the domain name(s) you would like on your certificate (comma and/or
space separated) (Enter 'c' to cancel):
```

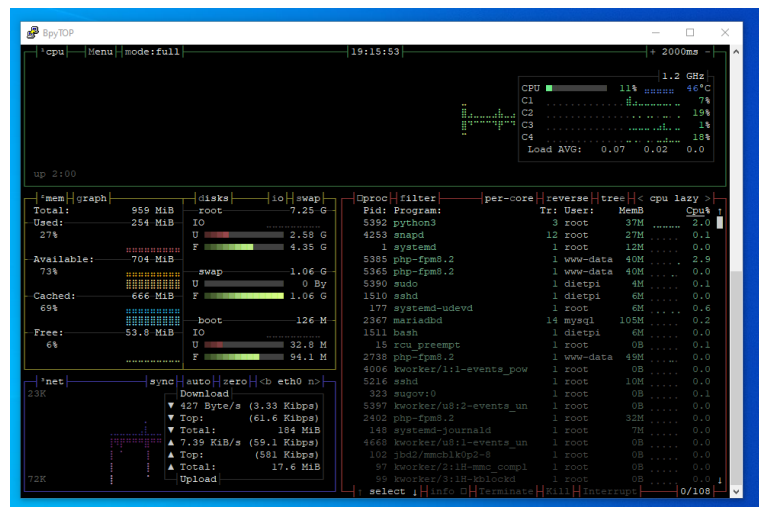
- Écrivez **sudo ufw allow proto tcp from any to any port 80**



```
dietpi@RpiSaeDGT:~$ sudo ufw allow proto tcp from any to any port 80
Rule added
Rule added (v6)
dietpi@RpiSaeDGT:~$
```

7. Testing

7.1. Vérification de la connexion sur WordPress



8. Reference Materials (error messages, troubleshooting, specific configurations, etc.)

8.1 Erreur possible avec le téléchargement de DietPI:

- ◆ Erreur de téléchargement ou de vérification possible
- ◆ Problème de comptabilité possible qui se repère par des erreurs de démarrage ou de plantages lors de l'installation.
- ◆ Erreur dû à un problème de carte SD défectueuse ou des autorisations manquantes.
- ◆ Problème de connexion réseau

8.2 Erreur possible avec l'installation de nginx:

- ◆ Message d'erreur si le port par défaut est déjà utilisé par un autre service
- ◆ Erreur possible lors de la configuration du fichier Nginx
- ◆ Message d'erreur possible si les permissions sont insuffisantes

9. Glossary

Raspberry Pi: The Raspberry Pi is a single board nano computer with an ARM processor about the size of a credit card.

Diet Pi: Diet Pi is a Debian-based Linux distribution developed for single-board computers like Raspberry Pi.

Nginx: Nginx open source or nginx is a free web server software

SSH: SSH stands for Secure Shell, which refers to both a communication protocol and a computer program.

WordPress: WordPress is an open-source content management system (CMS), written in PHP, that makes it easy to create and manage a website.

PHP: PHP is a general-purpose and open-source scripting language, specifically designed for web application development.

Operating System (OS): An operating system, or OS, defines a set of programs that is responsible for establishing a relationship between different hardware resources, applications, and the user.

PuTTY: PuTTY is a terminal emulator and client for SSH, Telnet, rlogin, and raw TCP protocols.

Visual Studio Code: Visual Studio Code is an extensible code editor developed by Microsoft for Windows, Linux, and macOS.

SFTP: SFTP (Secure File Transfer Protocol) is a file transfer protocol that leverages a set of utilities that provide secure access to a remote computer to provide secure communications.

10. Index

D Diet pi, 2 N	Nginx, 3 O OS, 4 P PHP, 5
--------------------------	---

PuTTY, 5

R

Raspberry Pi, 5

S

SFTP, 6

SSH, 7

V

Visual Studio Code, 6

W

Wordpress, 8