

Rapport De Test Détaillé

Virtualisation:	1
Machine Virtuel clients Windows:	2
Machine Virtuel Serveur Ubuntu:	2
Serveur DHCP:	2
Architecture réseau	2
Wordpress et Lamp:	3
LAMP	3
A. Installer Apache	3
B - Installer PHP	4
C. Installer MySQL/MariaDB	4
Wordpress	5
A. L'archive d'installation de WordPress	5
B – Créer une base de données propre à WordPress	6
C. Décompresser l'archive WordPress à la racine du site	7

Virtualisation:

Pour créer nos machines virtuelles clients/serveur nous avons utilisé le logiciel "Oracle VirtualBox" ainsi qu'une image ISO qui correspond à l'OS de la machine virtuel (exemple: On va prendre un fichier iso windows pour une machine virtuel windows).

Ci-dessous vous trouverez la configuration que nous avons utiliser pour la création de nos machines:

Machine Virtuel clients Windows:

Mémoire Vive: 4096 Mo

Processeurs: 4 CPU

Stockage: 50 Go

Machine Virtuel Serveur Ubuntu:

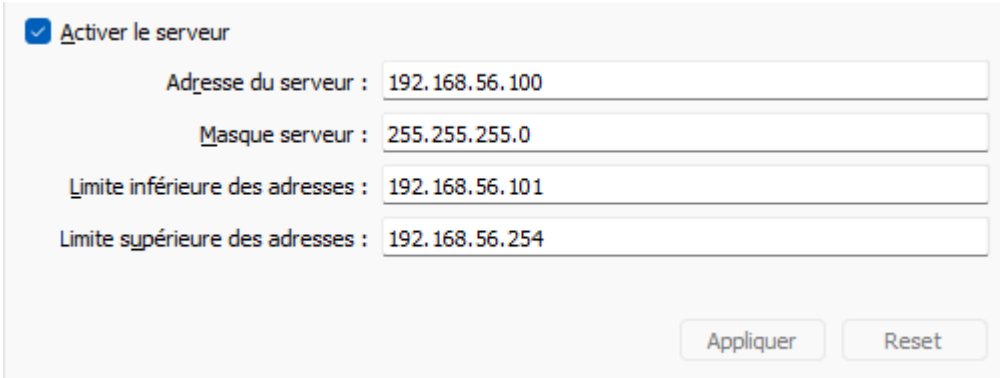
Mémoire Vive: 4353 MB

Processeurs: 5 CPU

Stockage: 25 Go

Serveur DHCP:

Configuration du serveur DHCP

A screenshot of a DHCP server configuration window. At the top, there is a checkbox labeled 'Activer le serveur' which is checked. Below this, there are four text input fields: 'Adresse du serveur' with the value '192.168.56.100', 'Masque serveur' with '255.255.255.0', 'Limite inférieure des adresses' with '192.168.56.101', and 'Limite supérieure des adresses' with '192.168.56.254'. At the bottom right, there are two buttons: 'Appliquer' and 'Reset'.

Architecture réseau

Concernant l'architecture réseau nous avons utiliser le logiciel "Cisco Packet Tracer" à l'aide de ce dernier nous avons pu schématiser notre infrastructure réseau de manière détaillée le schéma se fait en fonction de la configuration de votre infrastructure une fois le schéma hardware terminer on peut configurer les machines une par une en y ajoutant les adresses Ip tout en se référant à notre infrastructures (exemples : les adresses IP doivent être identiques à celle des machines de notre infrastructure).

Wordpress et Lamp:

Voici la procédure effectuer pour LAMP ET Wordpress:

LAMP

A. Installer Apache

On commence par mettre à jour le cache des paquets :

```
sudo apt-get update
```

Ensuite, on installe le paquet "apache2" afin d'obtenir la dernière version d'Apache

```
sudo apt-get install -y apache2
```

Pour qu'Apache démarre automatiquement en même temps que Linux, nous avons saisi la commande ci-dessous:

```
sudo systemctl enable apache2
```

Suite à l'installation du paquet, le serveur Apache démarre directement. On accède donc à sa page par défaut. Pour cela, il suffit de récupérer l'adresse IP du serveur :

```
ip address
```

Puis, à l'aide d'une machine équipée d'un navigateur, on peut accéder à notre serveur Apache :

```
http://10.0.2.15
```

Commande : **module utilisé pour la réécriture d'URL :**

```
sudo a2enmod rewrite
```

la commande **"a2enmod"** qui sert à activer un module. A l'inverse, la commande **"a2dismod"** sert à désactiver un module.

Activons trois autres modules :

- **"deflate"** pour la gestion de la compression, notamment en gzip, pour utiliser la mise en cache des pages sur votre site
- **"headers"** afin de pouvoir agir sur les en-têtes HTTP
- **"ssl"** pour gérer les certificats SSL et donc l'utilisation du protocole HTTPS

```
sudo a2enmod deflate
sudo a2enmod headers
sudo a2enmod ssl
```

Après avoir activé ou désactivé un module, ou modifié la configuration d'Apache, il faut redémarrer le service apache2 :

```
sudo systemctl restart apache2
```

Où se situent la configuration d'Apache et des sites dans tout ça ?

Le fichier de configuration d'Apache 2 est le suivant :

taper : `sudo etc/apache2/apache2.conf`

B - Installer PHP

taper : `sudo apt-get install -y php`

nous avons installer quelques paquets supplémentaires pour compléter l'installation de PHP sur notre serveur. Par exemple, pour permettre les interactions entre PHP et notre instance MariaDB.

```
sudo apt-get install -y php-pdo php-mysql php-zip php-gd php-mbstring
php-curl php-xml php-pear
```

C. Installer MySQL/MariaDB

Pour installer MariaDB, voici la commande à exécuter :

```
sudo apt-get install -y mariadb-server
```

Suite à l'installation, nous avons exécuter le script "**mariadb-secure-installation**" afin de sécuriser un minimum notre installation de MariaDB:

```
sudo mariadb-secure-installation
```

En résumé, nous allons pouvoir définir un mot de passe pour le compte "root" de MariaDB, empêcher les connexions distantes sur votre instance à l'aide du compte "root", empêcher les connexions anonymes et supprimer la base de test.

Nous avons redémarrer le service MariaDB:

```
systemctl restart mariadb
```

Notre serveur Lamp est donc bien installé.

Wordpress

A. L'archive d'installation de WordPress

Nous nous sommes positionner dans le dossier tmp pour récupérer l'archive zip d'installation de wordpress

première commande : `cd /tmp`

deuxième commande : `sudo
wget https://wordpress.org/latest.zip`

B – Créer une base de données propre à WordPress

Tout d'abord nous nous sommes connectés sur notre instance MariaDB (root)

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

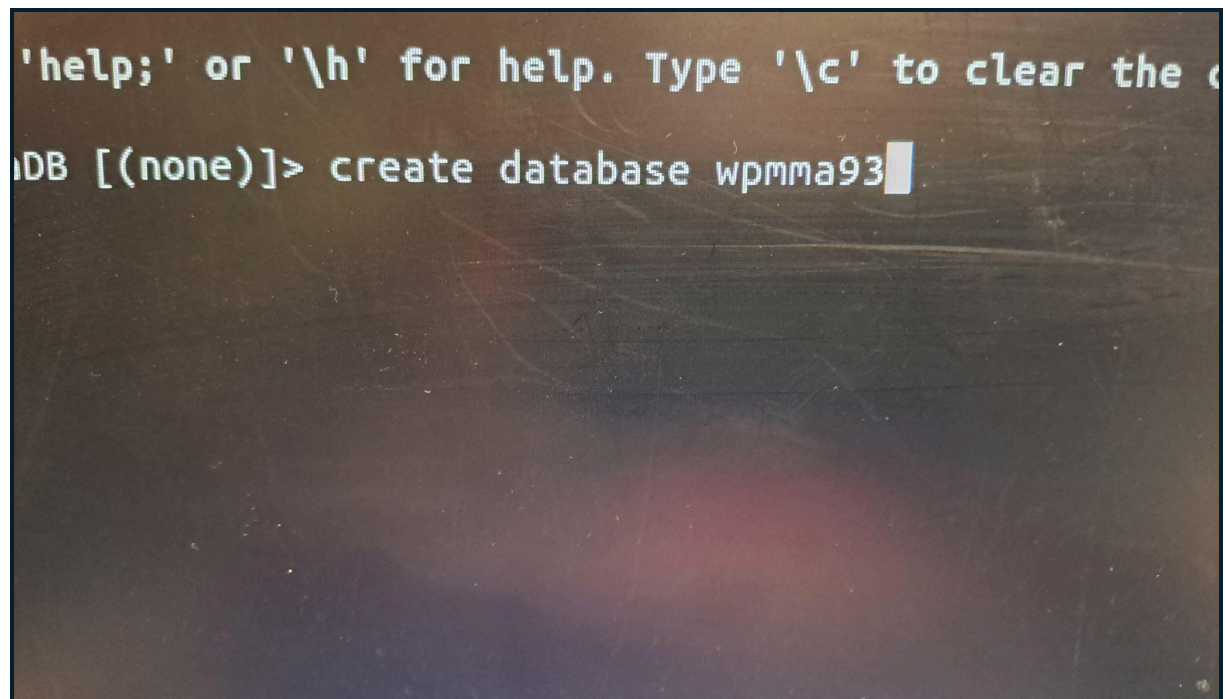
on insérer le mdp :1234

Nous sommes donc connectés sur la console.

Première étape : la création de la base de données.

Avec la commande :

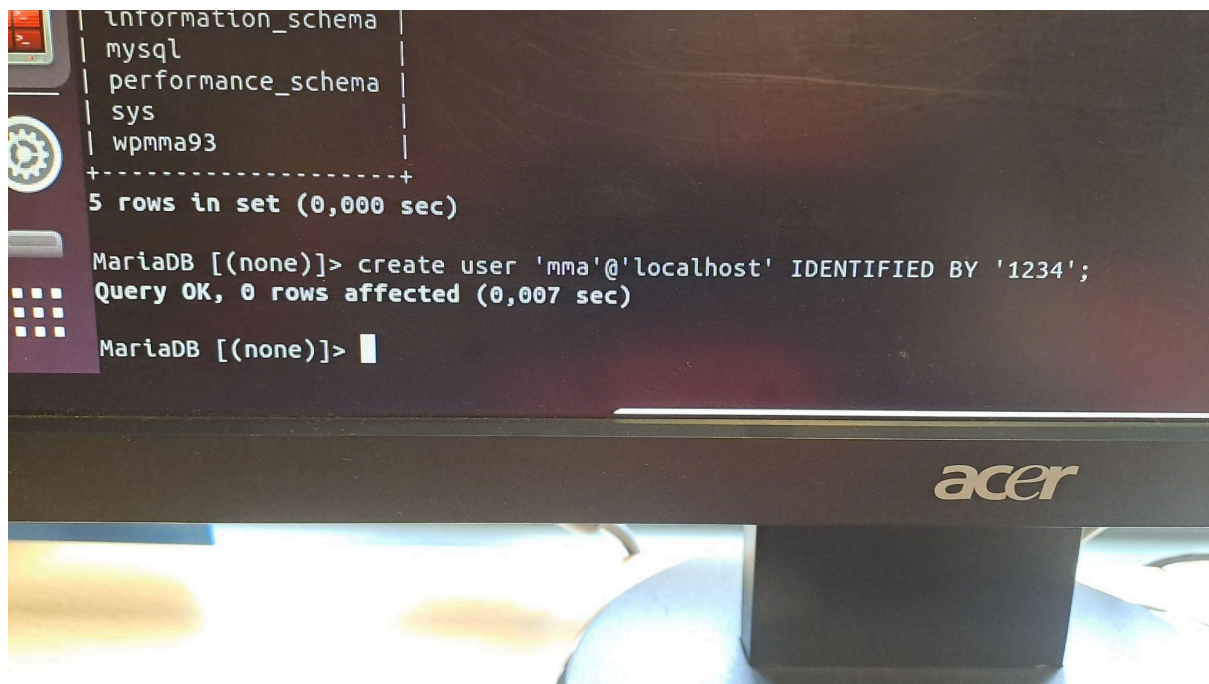
```
CREATE DATABASE wpmma93;
```



Deuxième étape : créer l'utilisateur qui sera administrateur de la base de données WordPress.

Avec la commande :

```
CREATE USER 'mma'@'localhost' IDENTIFIED BY '1234';
```



Troisième étape : donner tous les droits à l'utilisateur "*mma*" sur la base de données WordPress.

Avec la commande :

```
GRANT ALL PRIVILEGES ON wpmma93.* TO mma@localhost;
```

la commande suivante pour actualiser les droits et activer les nouveaux privilèges sur notre base de données :

```
FLUSH PRIVILEGES;
```

Quittez la console MariaDB / MySQL :

```
exit
```

C. Décompresser l'archive WordPress à la racine du site

on installe le paquet « zip » sur notre serveur pour pouvoir décompresser l'archive de WordPress :

```
sudo apt-get update  
sudo apt-get install zip
```

On décompresser l'archive dans `/var/www/html` grâce à la commande suivante (en étant positionné dans le dossier où l'on a téléchargé le fichier latest.zip) :

```
sudo unzip latest.zip -d /var/www/html
```

On se déplace dans le dossier `/var/www/html` :

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

Ensuite, on tape la commande ci-dessous pour déplacer tout le contenu du dossier "wordpress" à la racine de notre site :

```
sudo mv wordpress/* /var/www/html/
```

Enfin, on termine en donnant les droits à l'utilisateur `www-data`:

à l'aide de cette commande :

```
sudo chown -R www-data:www-data /var/www/html/
```

On se dirige dans le navigateur en tapant : `10.0.2.15/wp-admin/setup-config.php`

On se connecte en insérant: nom de la base de donnée, identifiant, mdp, adresse de la base de donnée.

On insère les infos nécessaires : titre du site:, identifiant, mdp, email.

Puis on clique sur installer Wordpress



Bienvenue

Bienvenue dans la très célèbre installation en 5 minutes de WordPress ! Vous n'avez qu'à remplir les informations demandées ci-dessous et vous serez prêt à utiliser la plus extensible et puissante plateforme de publication de contenu au monde.

Informations nécessaires

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

Titre du site

MmaActu

Identifiant

mma

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

.....

Show

Forte

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

Votre e-mail

mmaactu@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.

Installer WordPress

On supprimer le fichier "*wp-config-sample.php*", car il n'a plus d'intérêt (nous avons notre fichier *wp-config.php* définitif).

```
sudo rm /var/www/html/wp-config-sample.php
```

Ensuite, pour **appliquer des droits très restrictifs sur le fichier "wp-config.php" pour le basculer en lecture seule** seulement pour Apache. Indispensable pour des raisons de sécurité.

```
sudo chmod 400 /var/www/html/wp-config.php
```

Nous voici donc l'interface Wordpress.

