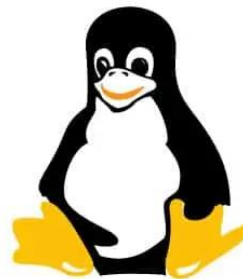


Installation, D'un Serveur web Linux



1.	Prérequis.....	3
2.	Qu'est-ce qu'un serveur web.....	4
3.	Installation serveur web	4
4.	Installation interpréteur PHP	5
1.	Ajout d'un dépôt Sury PHP	5
2.	Installation	5
3.	Vérification de l'installation	5
5.	Configuration d'Apache avec PHP	5
6.	Installation serveur de base de données	6
7.	Installation PHPMyAdmin.....	7

Situation d'entreprise :

Une startup voulait héberger en interne son site web vitrine pour en garder le contrôle total et limiter les coûts liés aux services cloud. J'ai été chargée d'installer un serveur web sous Linux (Debian), avec Apache, afin de permettre l'hébergement local du site. J'ai également configuré les droits d'accès, sécurisé l'hébergement avec un pare-feu, et testé l'accessibilité du site en local et depuis l'extérieur.

1. Prérequis

Nous devons avoir une machine Linux, avec une IP fixe et un nom de machine qui permet de l'identifier facilement, ainsi que son réseau fonctionnel.

2. Qu'est-ce qu'un serveur web

Un serveur web est un serveur qui nous permet d'afficher des pages aux formats HTML statique, depuis un client web (Navigateur). Il est possible d'ajouter un interpréteur PHP, qui nous permet d'avoir des pages dynamiques.

3. Installation serveur web

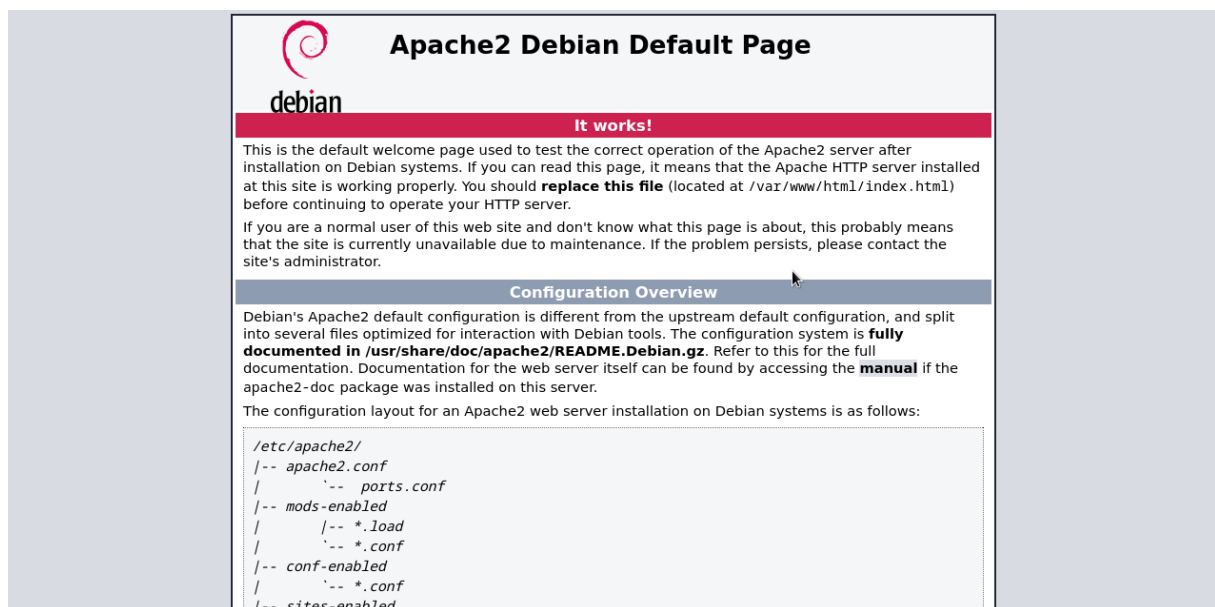
Pour installer le serveur web, nous allons devoir installer 1 paquet, nous devons effectuer cette commande

```
Apt-get install apache2
```

Emplacement du contenu du site : /var/www/html/

Nous pouvons avoir accès à notre site à l'adresse IP de notre serveur web. `http:// <Ip du serveur>/`

Cette page doit s'afficher, c'est la page créée par défaut sur tout serveur web apache



Cette page nous permet de certifier que notre serveur apache est fonctionnel

4. Installation interpréteur PHP

Un interpréteur PHP, nous permet de créer des pages dynamiques afin de pouvoir ajouter du contenu ou effectuer des actions plus poussées que peut supporter le HTML.

1. Ajout d'un dépôt Sury PHP

```
sudo apt install -y apt-transport-https lsb-release ca-certificates wget
```

Permet d'installer les outils nécessaires

```
wget -O /etc/apt/trusted.gpg.d/php.gpg https://packages.sury.org/php/apt.gpg
```

Permet de télécharger la clé GPG pour le dépôt

```
echo "deb https://packages.sury.org/php/ $(lsb_release -sc) main" | sudo  
tee/etc/apt/sources.list.d.php.list
```

Ajoutez le dépôt à votre fichier des sources

```
sudo apt update
```

Mettez à jour la liste des paquets

2. Installation

```
sudo apt install -y php8.3 libapache2-mod-php8.3 php8.3-cli php8.3-common php8.3-mysql
```

Installez la version souhaitée de PHP ici le 8.3

3. Vérification de l'installation

```
Php -v
```

Affiche la version de php

5. Configuration d'Apache avec PHP

```
sudo a2enmod php8.3
```

Activez le module PHP pour Apache

```
sudo systemctl restart apache2
```

Permet de redémarrer Apache pour appliquer les changements

Afin de vérifier que notre serveur PHP8.3 est fonctionnel, on crée un fichier qui retourne les informations du serveur PHP.

```
Nano /var/www/html/info.php
```

Fichier modifié "/var/www/html/info.php"

```
< ?php  
Phpinfo () ;  
?>
```

Une fois cette page créer, on se rendre sur cette page : http://<IP_du_serveur>/info.php

6. Installation serveur de base de données

Les base de données sont très utiles, elles permettent de stocker des informations pouvant être interrogées grâce à un langage comme le "SQL", et est accessible depuis le réseau.

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

Notre serveur est installé, mais n'est pas configuré. Pour cela, nous devons mettre un mot de passe à notre base de données. De base elle n'a pas de mot de passe.

```
mysql mysql -u root
```

Permet de rentrer en console dans notre serveur de base de données en utilisateur "root" et se positionner dans la base "mysql"

Nous allons effectuer les commandes SQL suivantes dans cette console.

```
UPDATE user SET password=PASSWORD('nouveau mot de passe') WHERE user="root";  
  
grant all on *.* to root@'%' identified by "nouveau mot de passe";  
  
FLUSH PRIVILEGES;  
  
exit
```

Afin d'autoriser les connexions ou requêtes externe à la machine, nous devons autoriser les connexions

```
nano /etc/mysql/my.cnf
```

Nous devons redémarrer le service afin d'avoir les modifications appliquées

```
/etc/init.d/mysql restart
```

Afin de vérifier que notre serveur accepte bien la connexion, nous pouvons nous connecter avec cette commande depuis une autre machine linux.

```
Mysql -h <IP_Serveur_Base_De_Données> -u root -p
```

Si la connexion s'effectue, alors il est possible de se connecter depuis un autre serveur vers la base de données. Utile pour les applications web ou pour la redondance de MySQL.

En cas d'échec utiliser cette commande

```
update user set plugin="" where User='root';
```

7. Installation PHPMyAdmin

PHPMyAdmin est une application web qui nous permet d'avoir une interface de gestion pour notre base de données, ce qui est plutôt pratique pour voir le contenu d'une base de données ou pour ajouter un utilisateur, ou bien exporter ou restaurer une base de données. Nous allons aller télécharger l'applications web depuis leur site officiel. Mais avant nous devons installer deux paquets. Lien téléchargement : <https://www.phpmyadmin.net/>

```
apt-get install wget unzip
```

Puis nous allons télécharger le fichier

```
wget https://files.phpmyadmin.net/phpMyAdmin/4.7.9/phpMyAdmin-4.7.9-all-languages.zip
```

Maintenant que notre fichier est téléchargé, nous allons le décompresser

```
unzip phpMyAdmin-4.7.9-all-languages.zip
```

Une fois extrait, nous pouvons supprimer l'archive

```
rm phpMyAdmin-4.7.9-all-languages.zip
```

Nous allons renommer notre dossier phpMyAdmin et nous allons le mettre directement dans le site

```
mv phpMyAdmin-4.7.9-all-languages /var/www/html/phpmyadmin
```

Notre interface est en place. Nous pouvons aller sur la page http://<IP_Serveur>/phpmyadmin