

DEBIAN

Le travail demander etait de mettre en place la distribution Debian en tant que serveur sur une machine virtuel (Virtual-Box) puis il fallait mettre en place un serveur LAMP (Linux , Apache , Mariadb et Php)

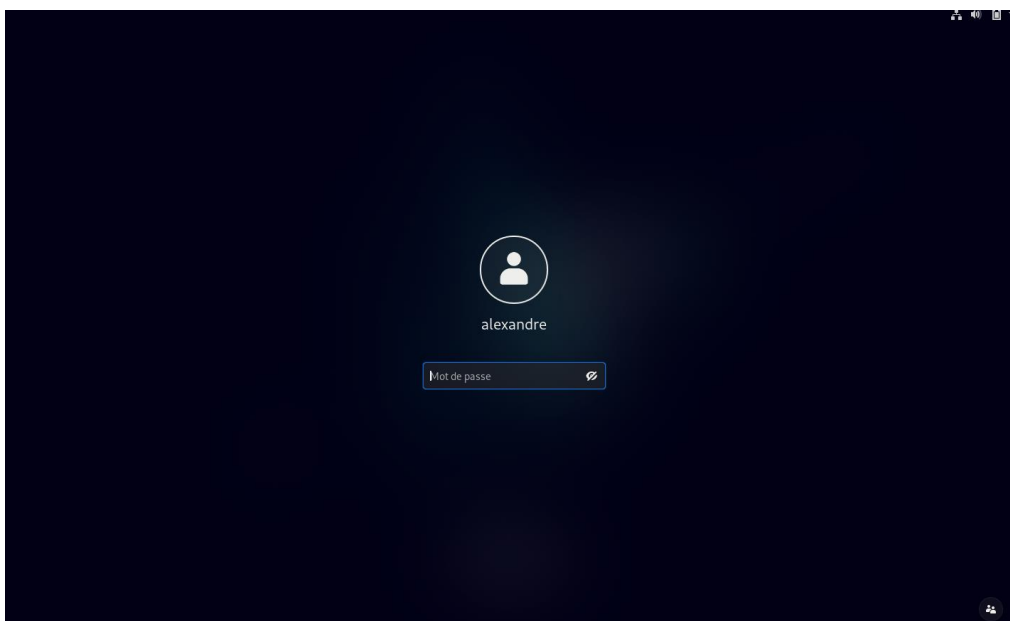
La première choses a mettre en place a etait la machine virtuel Debian avec une configuration spéciale :

- Notre nom
- 4G de RAM
- 80G pour le disque dur
- Utiliser VDI
- Utiliser L'Allocation dynamique (pour le stockage sur le disque dur physique)
- Sur l'écran Taille et emplacement du fichier cliquez sur crée
- Selectionnez Debian 11

Ensuite , nous avons mis les configurations adéquates il va falloir configurer l'image en suivant les différents paramètres

- Installation graphique
- Selectionnez la france
- Le nom de la machine est Debian
- Pas de nom de domaine
- Crée un mdp pour le superutilisateur (root) puis confirmer
- Mettre mon nom en minuscule
- Utiliser le disque entier
- Disque a partitionner
- Ne pas faire l'analyse d'autres supports d'installation
- Miroir = Deb.debian.org
- Gardez seulement Serveur SSH / Utilisateur usuels du système et le programme GNOME
- Installez GRUB
- Choisir la ligne du HARDDISK_

Puis on peut commencer à se connecter en tant que root



La première chose à faire quand on est dans le terminal est de mettre à jour en utilisant (apt-get update) puis faire (apt-get upgrade) pour obtenir la dernière version .

Une fois que on a mis nos paquets à jour il va falloir installer Vim qui est un éditeur de texte extrêmement personnalisable (ajout d'extensions, modification du fichier de configuration , ...)

Une fois qu'il est installé on va devoir installer le serveur Apache2 en utilisant les commandes suivantes :

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

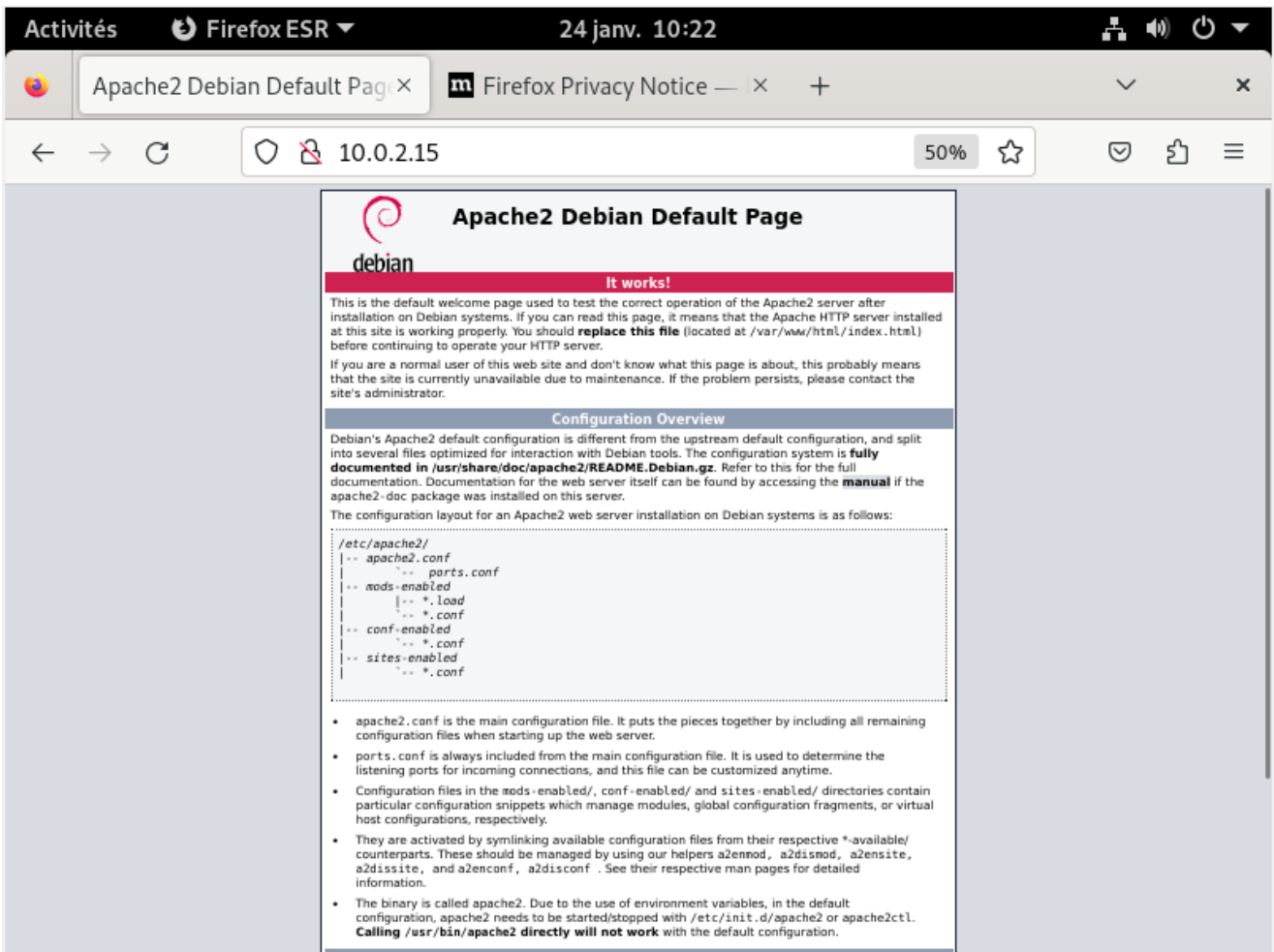
```
sudo systemctl enable apache2
```

Après avoir taper ces commandes ont devras obtenir un certains resultat :

```
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.  
Executing: /lib/systemd/systemd-sysv-install enable apache2
```

Une fois que le serveur apache a etait installer ont va chercher notre ip que l'on va mettre sur firefox par exemple (ip adresse) et sa nous montrera notre serveur apache

`http://10.0.2.15/`



Un serveur Apacheest un serveur web open-sources multiplateforme qui sert a ecouter les requêtes émises par les navigateurs, de chercher la page demandée et la renvoyer .

Ilva falloir regarder si la version de debian est bonne, activier les différents modules d'Apache pour faire tourner des site internet

```
sudo apache2ctl -v
```

```
sudo a2enmod rewrite
```

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

Deflate = Format de compression de données sans perte

Headers = Il contient les données présentes au début de ce fichier/ paquet

SSL = Protocole de sécurité qui crée un lien chiffré entre serveur web et navigateur

Dans un seconds temps il faut installer PHP :

Comme toujours ils faut installer les paquets php :

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

cette commande va installer certains paquets

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

```
libapache2-mod-php7.4 libsodium23 php-common php7.4 php7.4-cli php7.4-common
php7.4-json php7.4-opcache php7.4-readline
```

Si tout les est bien installer ont obtient ce resultat des différent paquets pout php

Ensuite, pour que php et mariadb puisse communiquer ensamble il faut installer d'autre paquet pour qu'il communique

Sur cette commande ont va installer tout les paquet pour que l'interaction en php et mariadb soit au point ont peut re-trouver :

```
-php-pdo
-php-mysql
-php-zip
-php-gd ...
```

qui sont tous notez les une après les autres.

Une fois que tout est installez ont va crée un fichier phpinfo.php en faisant :

```
sudo nano/var/www/html/phpinfo.php
```

Et dans le fichiers que ont vien de crée il faudra mettre du code php comme

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

une fois que ont fait sa ont va revenir chercher notre adresse ip et mettre http ://X.X.X.X/phpinfo.php

Ce qui va nous donne une page php comme sur la page suivante :

PHP (officiellement, ce sigle est un acronyme récursif pour PHP Hypertext Preprocessor) est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web

System	Linux Debian 5.10.0-27-amd64 #1 SMP Debian 5.10.205-2 (2023-12-31) x86_64
Build Date	Jun 9 2023 16:51:37
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gd.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

Et pour terminez il va valoir faire les installation de MariaDB ont va utiliser la commande suivante :

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

ceci va installez installez mariaDB et il va falloir obtenir la version de mariaDB en faisant mariadb -V. Après cela ont va ce connecter a notre instance en utilisant :

```
sudo mariadb -u root -p
```

```
show databases ;
```

si sa fonctionne alors ont devrait logiquement obtenir le resultat ci-dessous :

MariaDB Server est un système de gestion de base de données relationnelle open source. C'est l'un des serveurs de bases de données les plus populaires au monde

```
root@Debian:/home/alexandre# mariadb -V
mariadb Ver 15.1 Distrib 10.5.21-MariaDB, for debian-linux-gnu (x86_64) using
EditLine wrapper
root@Debian:/home/alexandre# sudo mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 30
Server version: 10.5.21-MariaDB-0+deb11u1 Debian 11

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)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0,001 sec)

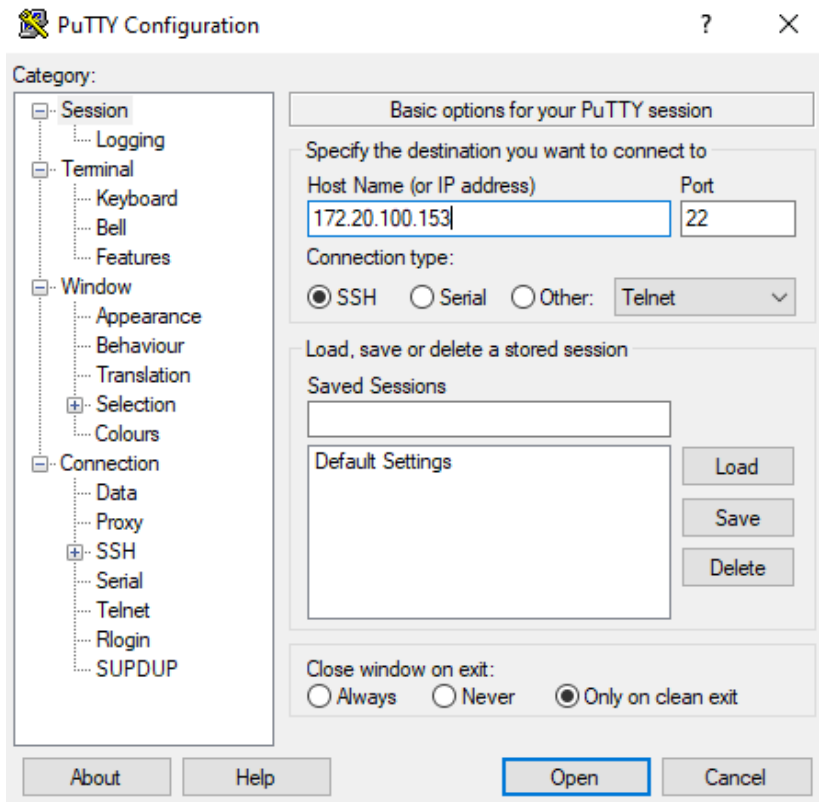
MariaDB [(none)]> █
```

Sur cette image ont peut voir nos bases de données crée comme :

```
-mysql
-information_shema
-performance_schema
```

Puis comme ont a fait avec php et apache ont va devoir redémarrer le service afin que les différentes modification soit appliquer

nsuite, nous avons essayer de ce connecter a notre serveur en utilisant putty :



Si sa ce connecte alors il suffira de mettre son nom et son mdp et la magie opère ont pourra directement taper des commande sur putty et plus dans le terminal :

