



Quentin
SCARPA

2025 Rapport

VirtualBox et Wordpress

Installation VM et setup
Wordpress



[VirtualBox] _____	3
[Présentation] _____	3
[Installation machine] _____	3
[Paramétrage du disque] _____	6
[Wordpress] _____	7
[Présentation] _____	7
[Installer Apache, PHP et MariaDB] _____	8
[Installer Wordpress] _____	10
[Conclusion] _____	12

Vocabulaire utile

Une **machine virtuelle (VM)** est un logiciel qui simule un ordinateur complet à l'intérieur d'un autre système. Elle permet d'exécuter un système d'exploitation (comme Debian, Windows, etc.) dans un environnement isolé, tout en utilisant les ressources (CPU, RAM, disque) de l'ordinateur hôte.

Un **hyperviseur** est un logiciel (ou parfois un matériel dédié) qui permet de créer et gérer des **machines virtuelles**.

Un **CMS** (Content Management System, ou système de gestion de contenu) est un logiciel qui permet de créer, gérer et publier facilement du contenu sur un site web sans avoir besoin de coder.

Une **image disque** est un fichier qui contient une copie complète d'un support de stockage (disque dur, clé USB, CD/DVD, etc.), y compris ses données, sa structure et son système de fichiers. Elle sert à **sauvegarder, cloner** ou **installer** un système (ex. : fichiers ISO utilisés pour installer un OS).

[VirtualBox]

[Présentation]



VirtualBox est un logiciel gratuit de virtualisation développé par **Oracle**. Il permet de créer et d'exécuter des **machines virtuelles (VM)**.

Avec VirtualBox, il est possible d'installer un autre système d'exploitation (appelé "invité"), comme **Linux**, **Windows** ou **macOS**, sans affecter le système principal. Cela permet de tester, développer ou exécuter des applications dans un environnement totalement isolé.

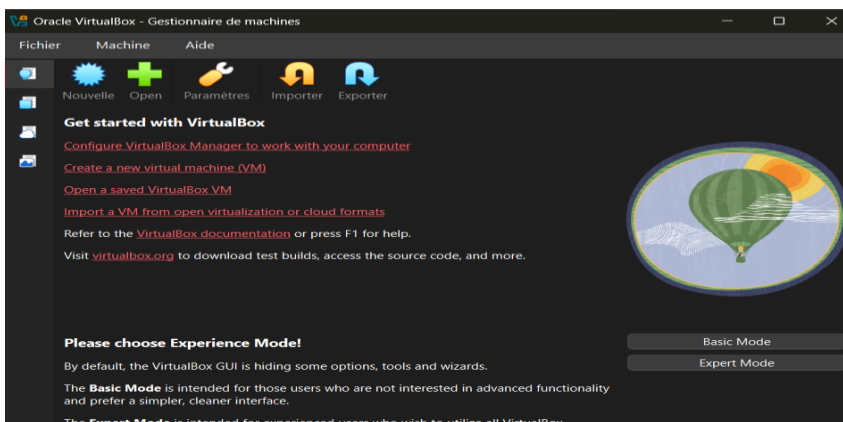
✓ Avantages :

- Gratuit et open source
- Multi-plateforme (Windows, macOS, Linux)
- Facile à installer et à utiliser
- Idéal pour les tests, les environnements de développement ou l'apprentissage

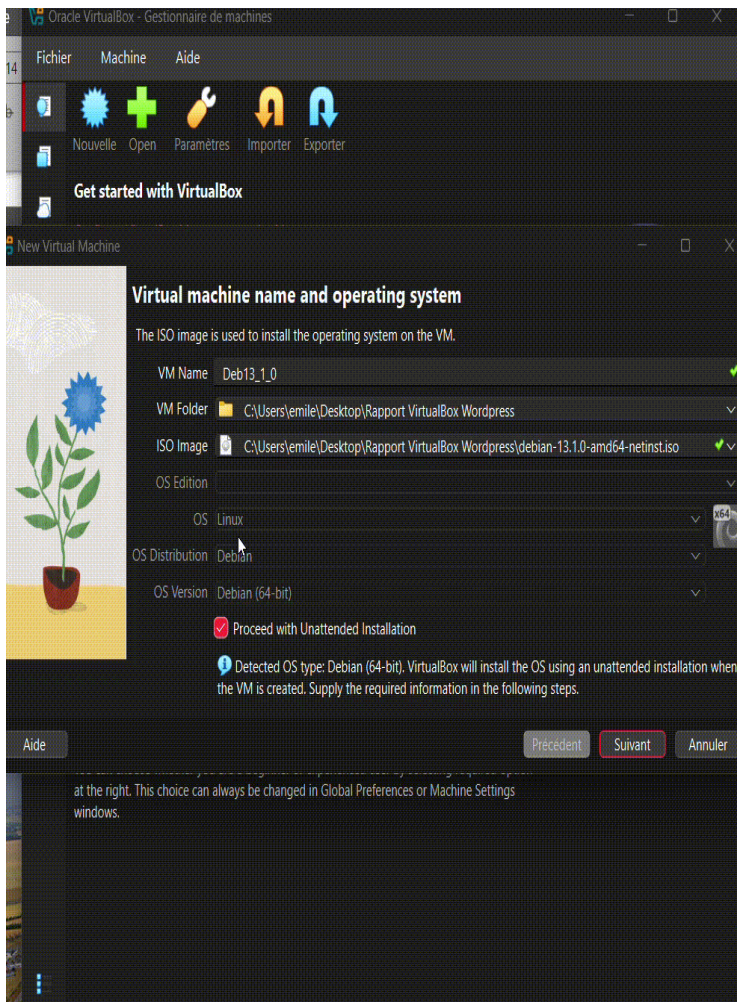
[Installation machine]

Pour créer une machine virtuelle on a besoin de **l'hyperviseur** mais on a aussi besoin d'une **image**, ici on utilisera Debian.

Interface de VirtualBox :



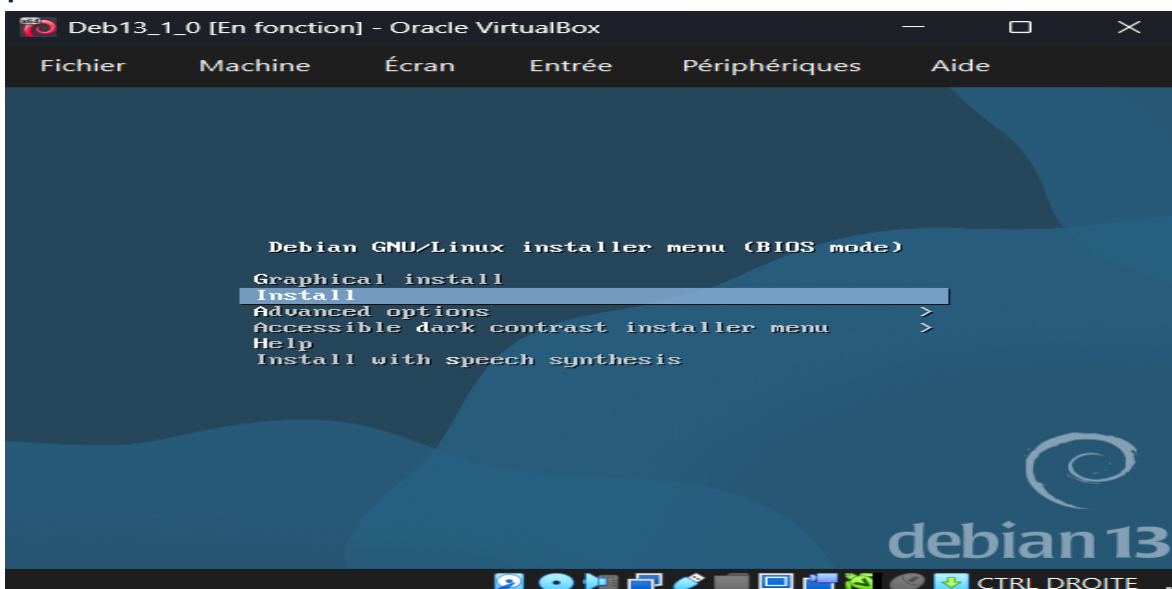
Ce qui nous intéresse c'est de créer une nouvelle machine.

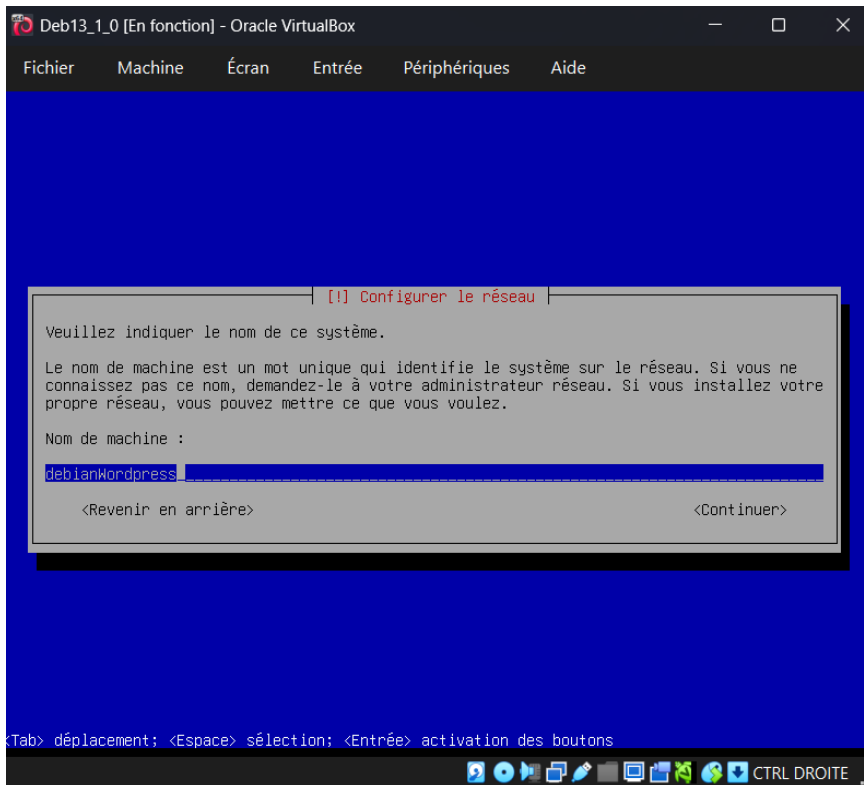


1. On choisit le nom de la machine et on décoche la case « proceed with unattended installation », une option qui permet d'automatiser un maximum l'installation d'une machine virtuelle

2. On règle ce que notre machine *hôte* va allouer à notre machine *invitée* en termes de ram et de cœurs de processeurs, enfin on crée un disque virtuel qui alloue dynamiquement l'espace c'est-à-dire qu'il ne pèse sur le disque physique que le poids nécessaire au stockage de données pour son environnement et non l'entièreté des 50 gigas que nous lui avons octroyés.

Ensuite on lance une installation, non graphique car pas nécessaire pour notre serveur wordpress.

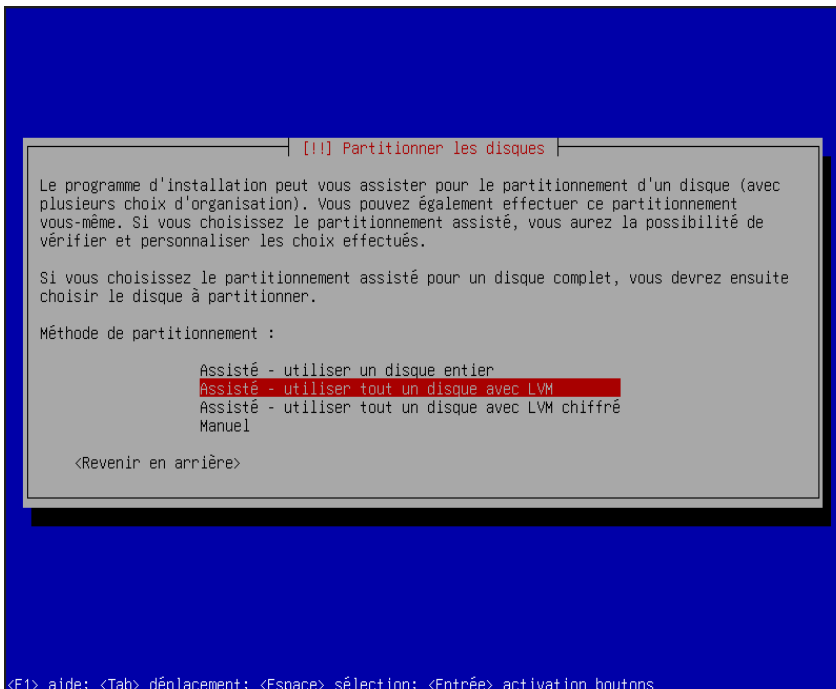




-Après avoir choisi le pack de langue et la région on se retrouve à choisir le nom de la machine, ici, *deblanKordpress* ainsi que le nom de domaine ici : *deblanKordpress.lan*

-Enfin on effectue la création du super user root et d'un utilisateur, ici *qscarpa*

[Paramétrage du disque]



Ici comme on peut le voir plusieurs choix s'offrent à nous :

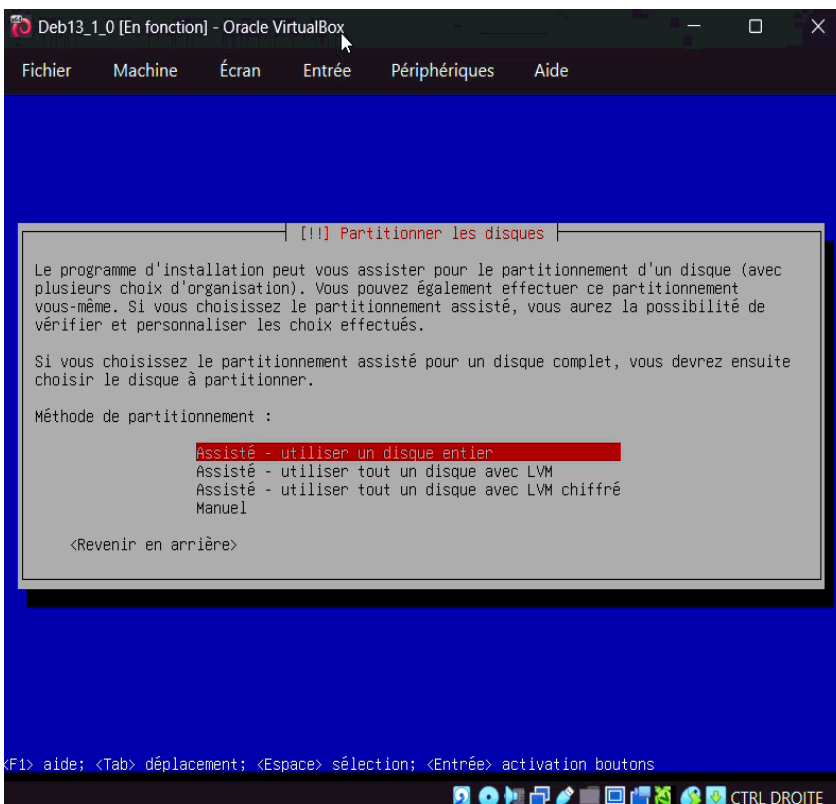
1. Utiliser tout le disque

Le système utilise l'intégralité du disque avec une

configuration automatique

-Avantages : Simple, rapide, aucune décision à prendre

-Inconvénients : Pas de personnalisation, potentiellement inefficace pour certains cas d'usage



Ou alors

2. Utiliser tout le disque avec LVM (Logical Volume Management)

Crée une structure de volumes logiques qui peut être redimensionnée facilement

-Avantages : Flexibilité pour modifier les partitions après installation, prise en charge du redimensionnement à chaud

-Inconvénients : Légèrement plus complexe, petite surcharge de performances

Idéal pour : Serveurs,

environnements où les besoins d'espace peuvent évoluer.

C'est la solution que nous allons choisir pour ce cas de figure car plus flexible avec une meilleure maîtrise sur nos fichiers.

Après l'installation de **Grub** sur le disque principal il est temps de passer à l'installation de Wordpress.

[Wordpress]

[Présentation]



WordPress est un **système de gestion de contenu (CMS) open-source, créé en 2003**. Il permet de créer et gérer facilement des sites web sans avoir besoin de coder.

✓ Avantages :

- . **Open-source et gratuit** : développé en PHP avec une base de données MySQL/MariaDB.
- . **Facile à utiliser** : interface intuitive pour publier des pages, articles et médias.
- . **Personnalisable** : plus de 60 000 plugins et des milliers de thèmes disponibles.
- . **Polyvalent** : utilisé pour des blogs, vitrines, boutiques en ligne (WooCommerce), portfolios, intranets, etc.



MariaDB est un système de gestion de bases de données relationnelles (SGBDR) open-source, créé en 2009 comme alternative à MySQL, suite à son rachat par Oracle.

✓ Avantages :

- . **Compatible MySQL** : MariaDB reprend les mêmes commandes et protocoles, ce qui facilite la migration.
- . **Performances améliorées** : meilleure rapidité et optimisation pour les grosses bases de données.
- . **Open-source et gratuit** : sous licence GPL, soutenu par une communauté active.

[Installer Apache, PHP et MariaDB]

Installer le serveur web Apache

- `sudo apt install apache2 -y`
- Apache est utilisé pour héberger les fichiers de WordPress et servir les pages au navigateur.

Installer MariaDB (SGBD)

- `sudo apt install mariadb-server mariadb-client -y`
- MariaDB permet de stocker les données du site (articles, utilisateurs, paramètres).

Installer PHP et les modules nécessaires

- `sudo apt install php libapache2-mod-php php-mysql php-xml php-gd php-curl php-mbstring unzip -y`
- PHP est le langage dans lequel WordPress est écrit. Les extensions ajoutent des fonctionnalités

Accéder à la console mariadb :

```
root@invitedDeb:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 11.8.3-MariaDB-0+deb13u1 from Debian -- Please help get to 10k
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)]>
```

Créez la base et l'utilisateur WordPress :

Création de la base de données

```
•CREATE DATABASE wordpress  
DEFAULT CHARACTER SET  
utf8mb4 COLLATE  
utf8mb4_unicode_ci;
```

Création d'un utilisateur

```
•CREATE USER  
'wpuser'@'localhost' IDENTIFIED  
BY 'mot_de_passe';
```

Octroiement des droits à l'utilisateur

```
•GRANT ALL PRIVILEGES ON  
wordpress.* TO  
'wpuser'@'localhost';
```

Enfin on fait un **FLUSH PRIVILEGES ;**

La commande **FLUSH** de MySQL est utilisée pour rafraîchir ou recharger divers caches et tables internes, garantissant que la base de données fonctionne efficacement et que tout changement est reflété immédiatement.

Enfin on peut sortir du terminal mariadb avec **EXIT ;**

[Installer Wordpress]

1. `sudo rm -rf /var/www/html/*`
 - `sudo` : exécute la commande avec les privilèges administrateur
 - `rm -rf` : supprime de manière récursive (`-r`) et forcée (`-f`, sans confirmation)
 - `/var/www/html/*` : cible tous les fichiers dans le répertoire web par défaut
 - **Effet** : vide complètement le dossier qui contient normalement ton site web
2. `sudo cp -r wordpress/* /var/www/html/`
 - `cp -r` : copie de manière récursive (inclut tous les sous-dossiers)
 - `wordpress/*` : source - tous les fichiers du dossier wordpress
 - `/var/www/html/` : destination - le répertoire web
 - **Effet** : copie tous les fichiers WordPress vers le dossier du serveur web
3. `sudo chown -R www-data:www-data /var/www/html/`
 - `chown -R` : change le propriétaire récursivement
 - `www-data:www-data` : définit l'utilisateur et le groupe (utilisé par Apache/Nginx sur Debian/Ubuntu)
 - **Effet** : donne la propriété des fichiers au serveur web pour qu'il puisse les lire et modifier
4. `sudo chmod -R 755 /var/www/html/`
 - `chmod -R` : change les permissions récursivement
 - `755` :
 - Propriétaire : lecture + écriture + exécution (7)
 - Groupe : lecture + exécution (5)
 - Autres : lecture + exécution (5)
 - **Effet** : définit les permissions appropriées (le propriétaire peut tout faire, les autres peuvent lire et accéder)

En résumé : cette séquence nettoie ton répertoire web, y installe WordPress, puis configure les bons propriétaires et permissions pour que le serveur web fonctionne correctement.

On peut ensuite créer notre fichier de configuration Wordpress à l'aide de ces commandes :

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

```
cp wp-config-sample.php wp-config.php
```

```
nano wp-config.php
```

Les valeurs que nous avons modifiées dans notre fichier sont les suivantes :

```
// ** Database settings - You can get this info from your web host
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'wpuser' );

/** Database password */
define( 'DB_PASSWORD', 'P@ssword1' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** 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
 * the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress
 *
 * You can change these at any point in time to invalidate all exist
```

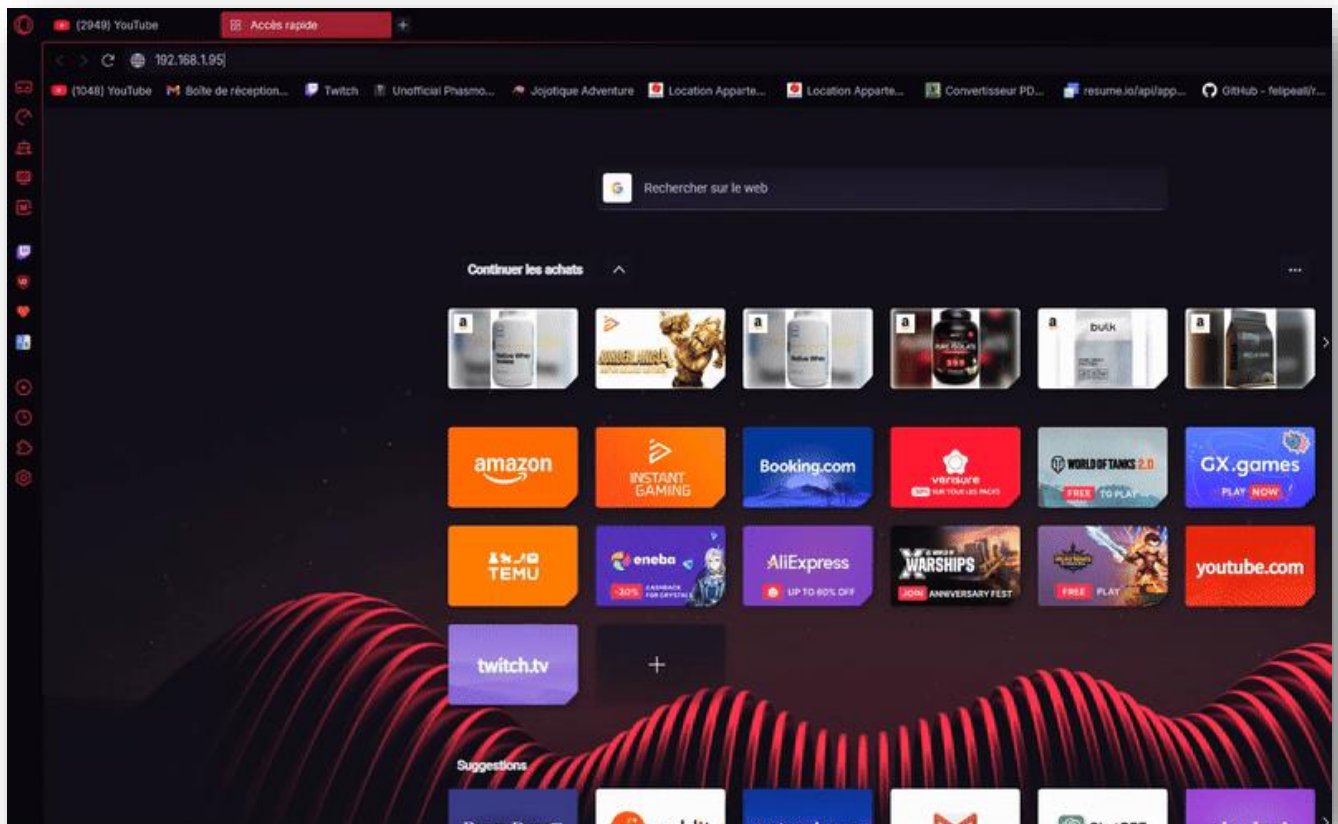
Pour accéder à notre site il nous faut l'adresse mail de la machine sur lequel la base de données se situe. Pour ce faire on fait un **ip a** où l'on obtient ce résultat :

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP>
    link/ether 08:00:27:c9:0d:5d brd ff:ff:ff:ff:ff:ff
    altname enx080027c90d5d
    inet 192.168.1.95/24 brd 192.168.1.255 scope eth0
        valid_lft 82583sec preferred_lft 71783sec
    inet6 fe80::fbd7:682b:5a9e:3c97/64 scope link
        valid_lft forever preferred_lft forever
root@invitedeb:/var/www/html#
```

L'adresse qui nous intéresse est celle encadrée.

Avec elle on peut se connecter sur notre navigateur à notre cms.

Résultat :



[Conclusion]

On peut grâce aux machines virtuelles héberger plusieurs sites de manière local grâce aux machines virtuelles.

Développement et tests

- Tester de nouveaux thèmes, plugins ou mises à jour sans risquer de casser votre site en production
- Développer de nouveaux sites dans un environnement isolé avant le déploiement
- Expérimenter avec différentes configurations PHP, MySQL ou versions de WordPress

Formation et apprentissage

- Apprendre WordPress sans frais d'hébergement
- Tester vos compétences en développement où même sans coder d'où l'utilité d'un CMS

Dans mon prochain document j'expliquerai le procédé de création de sites et comment mettre en place un portfolio simple.