

Déploiement

Introduction

Le déploiement est la mise en production d'un service informatique. Lorsque l'on développe un site internet sur un serveur local, il faut par la suite le déployer sur un serveur web.

La majeure partie des serveurs web sont réalisés sous le système d'exploitation Linux. C'est pourquoi on va étudier le déploiement d'une application sous Linux, via une machine virtuelle.

Virtualisation

Etant donné que le système d'exploitation actuel de la machine est Windows, il faut au préalable virtualiser un environnement Linux.

Installation d'un hyperviseur

Il est proposé d'installer l'hyperviseur l'outil VirtualBox développé par Oracle. Il s'agit d'un logiciel libre permettant de créer un environnement virtuel isolé sur la machine hôte.

Installation de GNU/Linux

Avant de pouvoir installer une machine virtuelle sous Linux, il faut télécharger le système d'exploitation. La distribution Debian est préconisée car stable. Il faut télécharger une version stable de Debian (pas la version testing), au format iso.

Écran d'accueil de VirtualBox v.6.1.

- Dans le logiciel VirtualBox, cliquez sur "Nouvelle" pour créer une nouvelle VM.- Dans la fenêtre "Créer une machine virtuelle" :

- Nom : choisir un nom,
- Dossier de la machine : le dossier où est enregistré le fichier iso
- Type : ici on choisit Linux
- Version : ici on choisit Debian
- Puis cliquez sur "suivant"

Ecran "Créer une machine virtuelle"

- On choisit ensuite la mémoire à allouer à la machine virtuelle,

- On sélectionne ensuite le type de fichier disque dur (ici un VDI),
- On choisit l'emplacement et la taille du fichier de disque dur (ici environ 10Go),
- On peut ensuite cliquer sur "Créer" pour lancer la création de la VM.
- Dans VirtualBox, cliquez sur "Configuration" pour modifier les paramètres de la VM.
- Dans VirtualBox, cliquez sur "Démarrer" pour lancer la VM.
- Dans la fenêtre "Choisissez le disque de démarrage", sélectionnez le fichier iso de la distribution Debian, puis cliquez sur "Démarrer".

Debian se lance alors dans une fenêtre. Il est alors proposé de définir les profils utilisateurs. Le profil "root" correspond à l'administrateur, le profil "user" est un utilisateur classique.

Dans la liste des logiciels à installer, sélectionnez "environnement de bureau Debian", "xfce" et "utilitaires usuels du système".

Xfce est un environnement de bureau léger pour les systèmes d'exploitation Linux. Il est rapide et léger, tout en étant visuellement attractif et facile d'utilisation. Il est recommandé d'utiliser Xfce pour une meilleure visualisation des tâches effectuées sur la VM.

Il est également nécessaire d'installer Grub, qui est le gestionnaire de démarrage de Linux.

Terminal de commandes

L'interpréteur de commandes par défaut sous Linux est Shell. Le terminal par défaut est Xterm.

Lignes de commande

- Afficher le répertoire en cours : pwd
- Changer de répertoire : cd "nom du répertoire"
- Revenir au répertoire précédent : cd -
- Lister les fichiers présents dans un répertoire : ls
- Lister les fichiers présents dans un répertoire avec leurs droits associés, sous forme de liste et en incluant les fichiers cachés : ls -al
- Créer un fichier : cat > "nom du fichier"
- Insérer du texte dans un fichier : echo "texte" >> "nom du fichier"
- Supprimer un fichier : rm "nom du fichier"
- Afficher le contenu d'un fichier : cat "nom du fichier" ou less "nom du fichier"
- Créer un répertoire : mkdir "nom du répertoire"
- Créer un lien symbolique : ln -s "dossier cible" "nom du lien"
- Supprimer un répertoire : rmdir "nom du répertoire"

- Copier un répertoire : cp "nom du fichier à copier" "nom de la copie"
- Renommer un répertoire : mv "nom du répertoire" "nouveau nom"
- Déplacer un répertoire : mv "répertoire source" "destination"
- Afficher le manuel de la commande "find" : man find (man "nom de commande")
- Chercher un fichier sur votre disque en se basant sur son nom : locate "nom du fichier" ou find / -name "nom du fichier"
- Chercher du texte dans un fichier : grep "mot à chercher" "fichier"
- Afficher le texte "Bonjour tout le monde" : echo "Bonjour tout le monde" (echo "texte")
- Afficher l'historique des commandes qui ont été tapées : history
- Afficher la version du système d'exploitation installée : uname
- Afficher la date et l'heure : date
- Afficher la durée depuis laquelle le système d'exploitation est allumé : uptime
- Rechercher les mises à jour disponibles pour le système : apt update
- Installer les nouvelles mises à jour disponibles depuis la dernière recherche : apt upgrade
- Se connecter en tant que superutilisateur : sudo -i
- Installer l'éditeur de texte "emacs" : sudo apt-get install emacs
- Connaître son/ses adresses ip : hostname -I

Configuration réseau

Protocole SSH

Pour pouvoir administrer la machine virtuelle, il faut se connecter à un serveur à l'aide du protocole SSH.

Après avoir récupéré l'adresse IP de votre VM, se connecter à l'aide du protocole ssh.

L'IP de la machine virtuelle est 10.0.2.15, ce qui nous provoquera des request timeout après plusieurs tests, afin de contourner cela nous suivons les instructions pour créer un nouveau "VirtualBox Host-Only Ethernet Adapter", nous activons le serveur DHCP, puis dans la partie Réseau de la Configuration de votre VM, nous activons, l'Adapter 2 et on choisit un mode d'accès réseau "Réseau privé hôte" en sélectionnant l'interface récemment créée.

Cela ne suffit pas et l'IP reste inchangé, nous suivons donc le reste des instructions en ajoutant les lignes suivantes dans le fichier :

/etc/network/interfaces afin de demander une adresse ip au serveur dhcp :

```
auto enp0s3
```

```
iface enp0s3 inet dhcp
```

Toujours aucun changement d'IP malgré un restart systemctl sshd et une redémarrage de la VM.

Après plusieurs recherches sur le net, je décide de rajouter un accès par pont pour l'adapter l avec la carte réseau de ma machine physique, puis je change le port de 22 à 2222 comme demandé via la commande :

...

```
sudo vim /etc/ssh/ssh_config
```

...

Avec Vim préalablement installé. #nanosucks

Enfin je rajoute le port 2222 en port autorisé pour la connexion via la commande :

...

```
sudo /sbin/iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 2222 -j ACCEPT.
```

...

Miracle, le changement d'IP est effectif et je peux enfin me connecter en ssh à ma VM.

Installation du serveur web

Afin d'avoir un serveur web, il est nécessaire d'installer différents services.

Sous Windows, WAMP et XAMPP sont les principaux logiciels qui permettent d'avoir un site

web complet. Sous Mac, il y a MAMP.

Pour Linux il y a LAMP, qui est en fait un ensemble de logiciels à installer séparément.

Les différents services à installer sont :

- Apache
- La dernière version de php disponible
- Mariadb server
- Phpmyadmin
- Les différents modules/connecteurs

Pour cela je suis le tutoriel <https://doc.ubuntu-fr.org/lamp> et exécute les lignes de commandes.

PHPmyadmin

- connexion de la VM à internet via un réseau NAT pour l'adapter 1;
- connexion à la machine hôte à la VM via un réseau hôte privée pour l'adapter 2;
- connexion à mariadb SQL via sudo mysql, puis création de l'utilisateur 'user';

- connexion à phpmyadmin via l'IP de ma VM soit : 192.168.209.3/phpmyadmin et localhost/phpmyadmin, connection à phpmyadmin avec utilisateur 'user'.
- Rajout de l'Alias : /pma dans /etc/apache2/conf-available/phpmyadmin.conf pour accéder à phpmyadmin via localhost/pma;

Déploiement du site

- Création de la base de donnée API, git clone du projet API dans /var/www/api après avoir rajouté la clé public ssh de la VM sur mon compte github.
- Changement du document root dans /etc/apache2/sites-available/000-default.conf en '/var/www/api';
- Installation de composer via le site officiel doc car il manque le fichier composer.phar via le gestionnaire de paquet debian ce qui provoque des erreurs pour les commandes suivantes;
- Exécution de "composer install" pour installer les différents modules nécessaires à Symfony et API platform;
- Modification du fichier .env en rajoutant le préfix mariadb devant la version sur la ligne mysql (nous étions sur postgres) ;
- Modification de la version php sur apache2 pour que soit compatible avec notre API (j'avais installé la version 7.4 au lieu de 8.0);

...

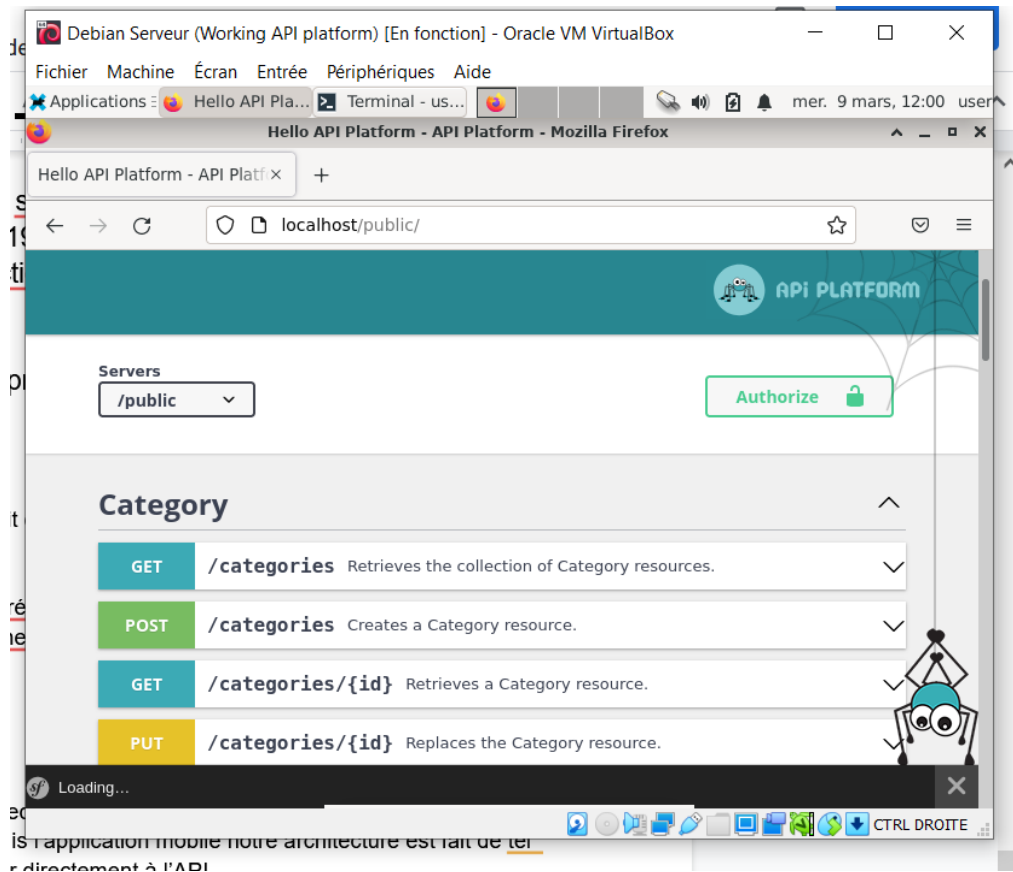
```
sudo a2dismod php7.4
```

```
sudo a2enmod php8.0
```

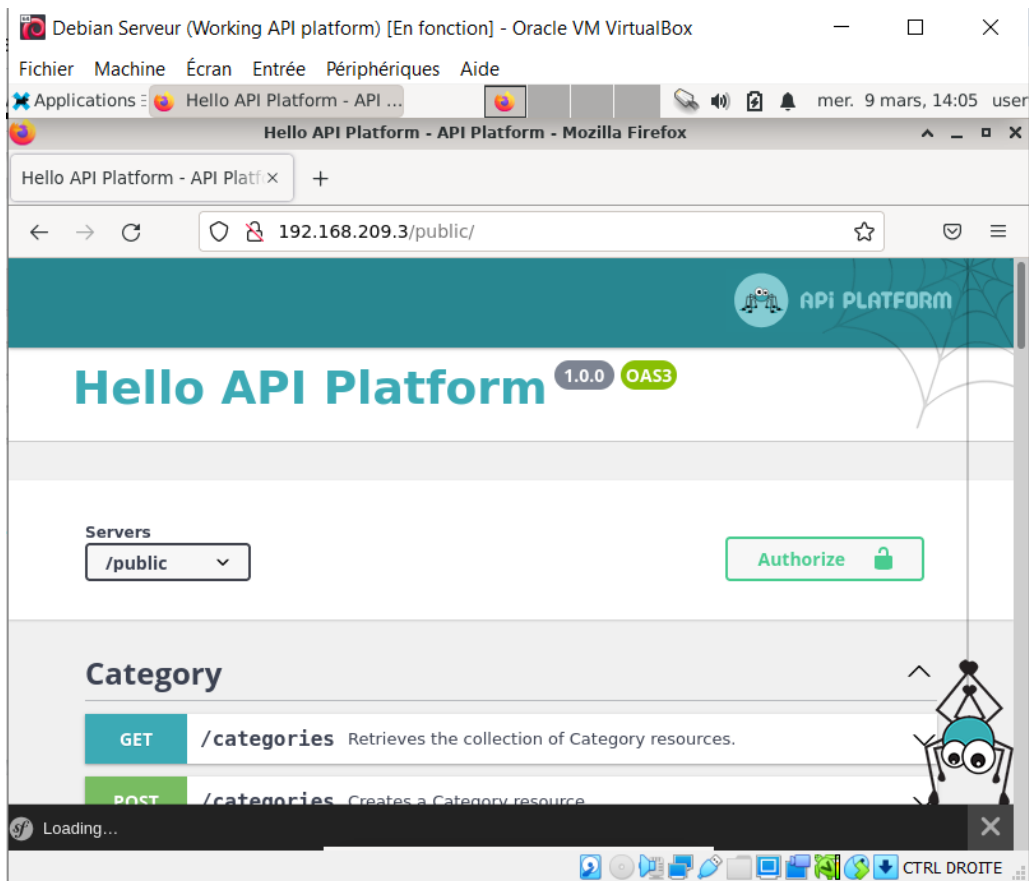
```
sudo service apache2 restart
```

...

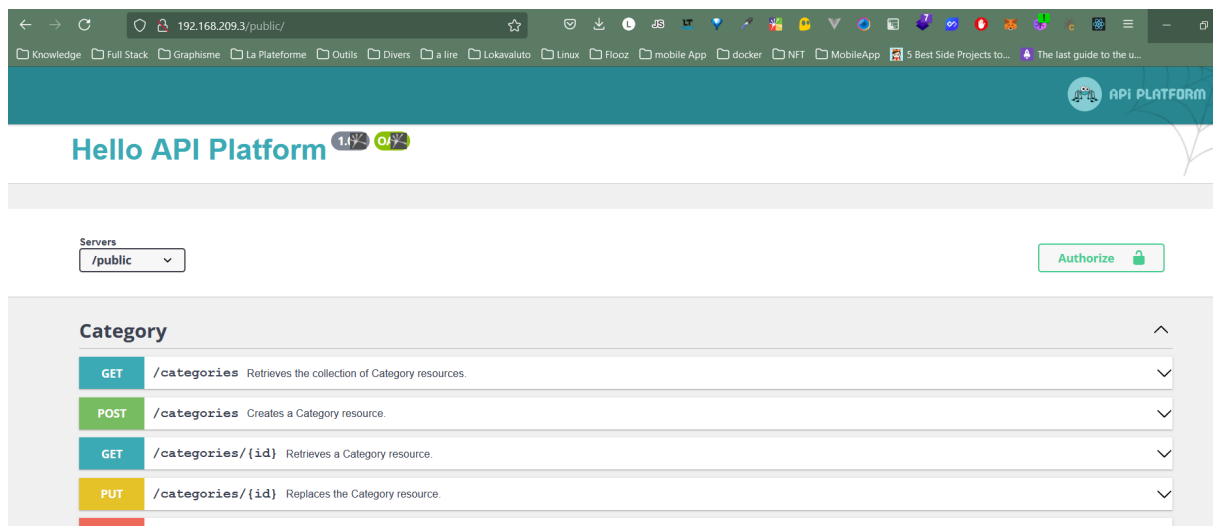
- Commande symfony pour faire la migration en prenant soins d'effacer le versionning de la migration précédente qui était en environnement postgres;
- Victoire totale directement sur la VM en localhost :



- Pareil avec l'adresse IP de la VM:



- Même chose côté machine hôte :



- Après connexion au wifi de l'école avec mon téléphone, impossible de me connecter à l'IP de la VM : NSURL error domain;
- Notre application mobile est déjà configuré pour accéder à notre API hébergé sur heroku, je me permets donc de passer cette étape puisque déjà réalisé dans un

environnement de développement cloud sur le lien suivant :

<https://intense-springs-77079.herokuapp.com/>

Répétition des opérations avec une connexion avec bridge

Il est toujours possible de se connecter en SSH à la VM et à API platform, cependant nous n'avons plus accès à internet sur la vm :

...

```
user@debian-serveur-vm:~$ host security.debian.org
```

```
;; connection timed out; no servers could be reached
```

...

Cela est dû au système de portail captif Alcasar, en effet, via un partage de connexion sur mon téléphone j'ai accès à internet sur la VM.

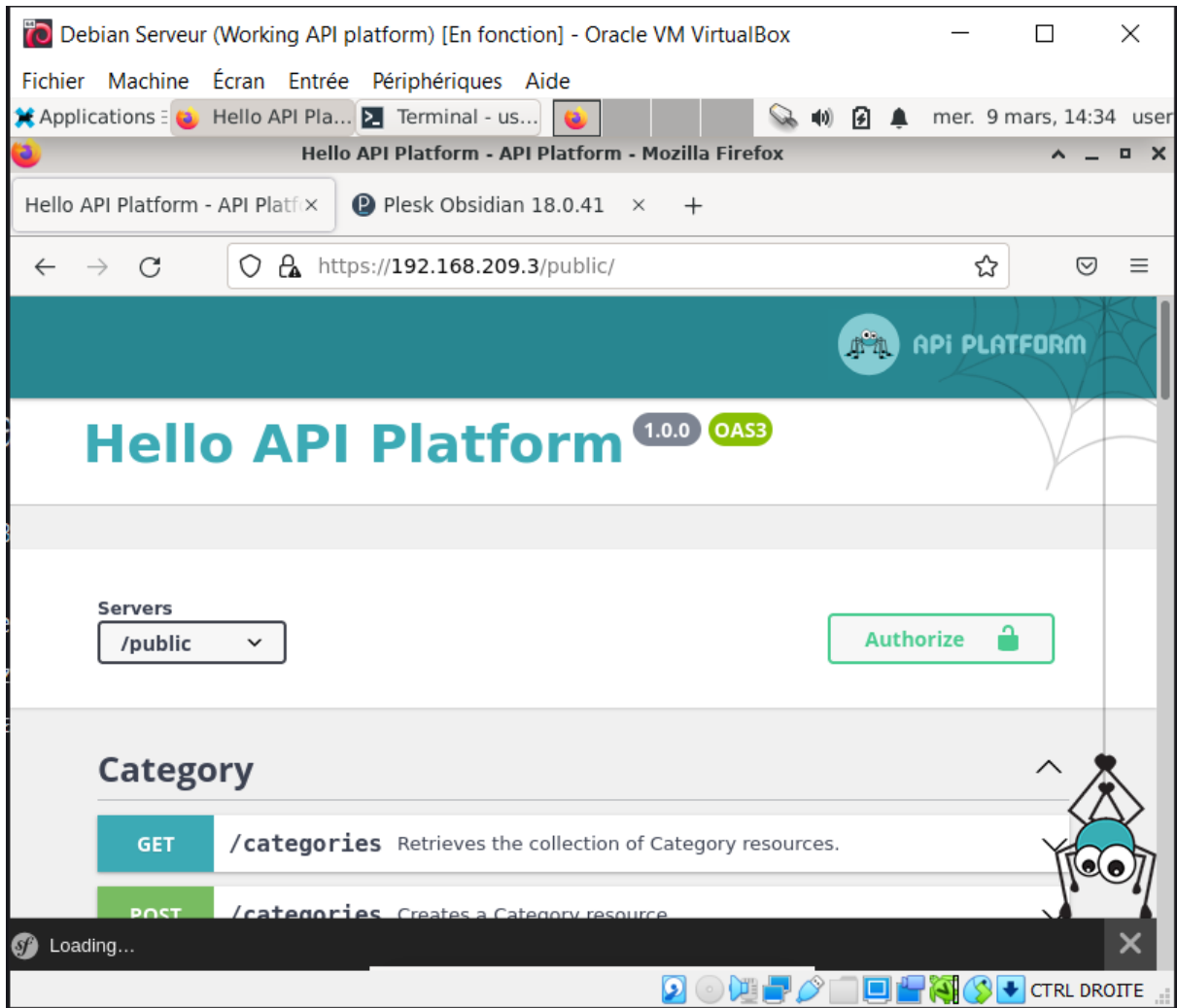
Pour le reste, Api platform est toujours accessible sur la VM et la machine hôte.

Sécurité : installation du certificat ssl

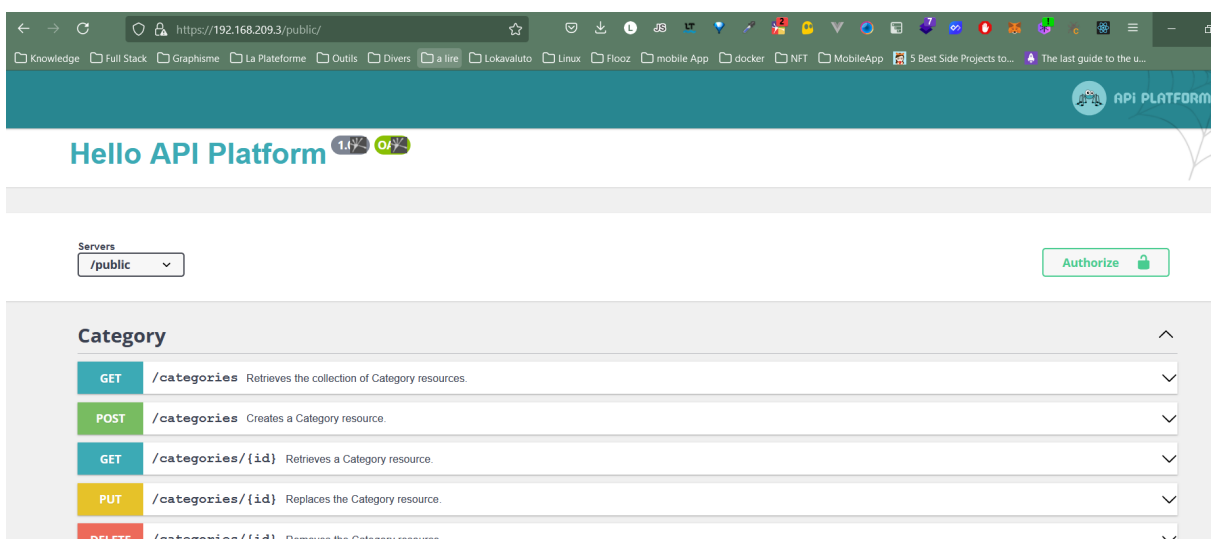
Après avoir suivi ce tutoriel à la commande près hormis la partie firewall :

<https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-20-04-fr>

voici le résultat côté VM :



et côté machine hôte :



A quoi sert un certificat SSL ? Un certificat SSL est un fichier de données qui lie une clé cryptographique aux informations d'une organisation. Installé sur un serveur, le certificat active le cadenas et le protocole « https », afin d'assurer une connexion sécurisée entre le serveur web et le navigateur.

Diagramme de déploiement en UML

Android Application Deployment Diagram

