**LOG3000**

**Processus du génie logiciel**

**Hiver 2018**

TP4

Infrastructure en tant que code

Présenté par :

Taleb Souli (1770491)

David Tremblay (1748125)

02 Avril 2018

École Polytechnique de Montréal

Département de Génie Informatique et Logiciel

**Questions d’analyses**

**Q1.Sous quelle distribution de linux roule le site? Donnez le “uname -a” complet.**

La distribution de Linux sur laquelle roule le site est la Linux photon-machine version 4.4.41.ph1-esx.

****

**Q2. Comment faire pour effacer un volume dans docker ?**

La commande suivante nou permet de supprimer un volume d’un conteneur docker:

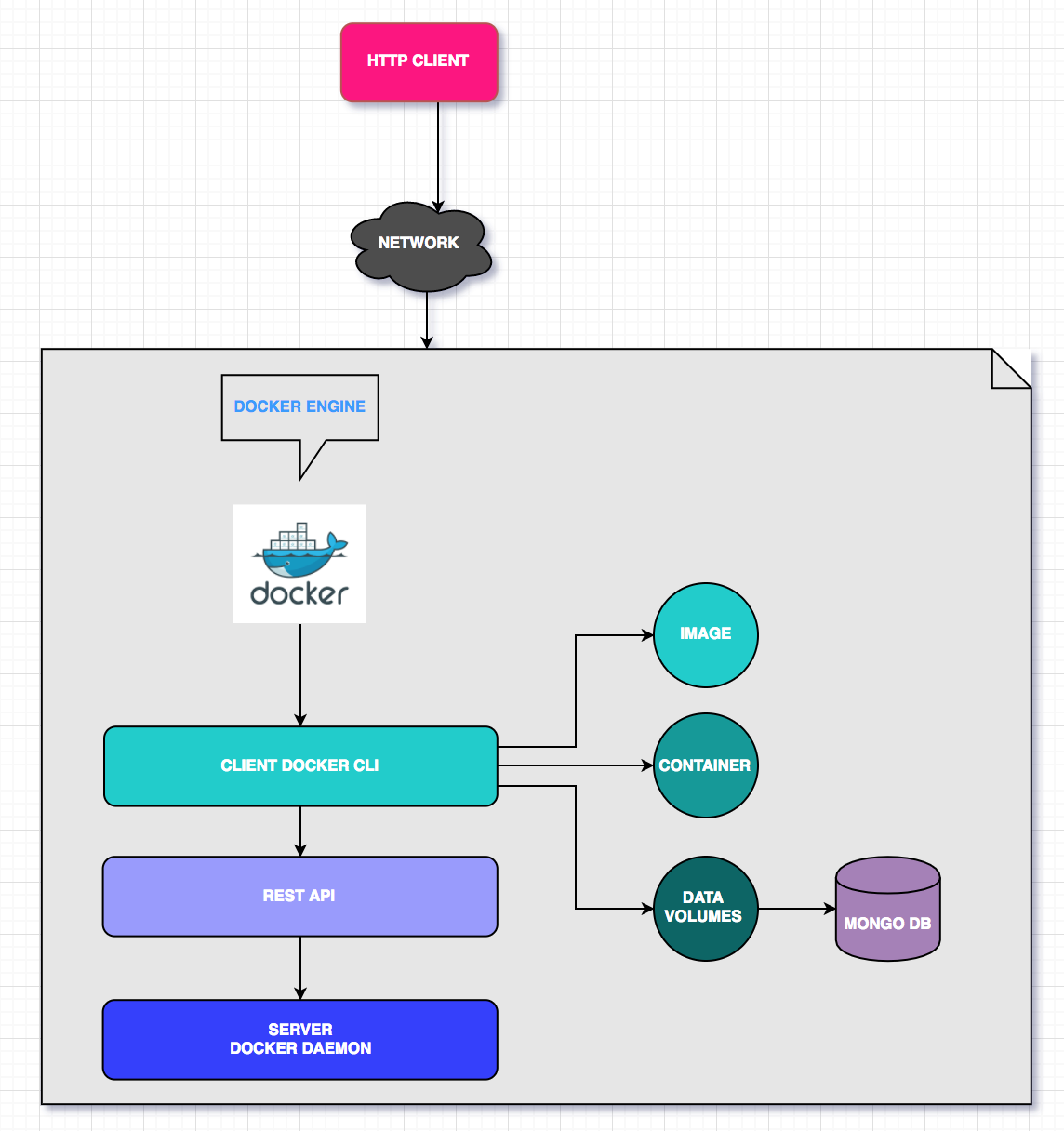
**docker volume rm [OPTIONS] VOLUME [VOLUME...]**

On pourrait ajouter cette option à la commande:

**--force , -f : Pour force l’effacement d’un ou de plusieurs volumes.**

**Questions de réflexion**

**Q1. À l’aide d’un diagramme UML, expliquer le chemin que font vos requêtes http au travers les conteneurs lorsque vous visitez le site.**



**Q2. Décrivez les avantages de déployer des applications avec un système d’IaC tel que Docker.**

*LÉGÈRETÉ DES CONTENEURS*

Les conteneurs sont de moins grande taille qu’une machine virtuelle (seulement quelques Mo). Aussi, les conteneurs n’ont pas de couche OS et l’espace mémoire requise est donc diminuée.

*RAPIDITÉ ET FACILITÉ DE DÉPLOIEMENT DES APPLICATION*

Il suffit que d’une ou deux commandes pour exécuter un conteneur Docker et celui-ci démarre très rapidement.

*AUTOMATISATION DES DÉPLOIEMENT*

L’automatisation du déploiement permet la reproduction d’environnements. Cela permet d’obtenir des copies identiques sur plusieurs machines afin d'uniformiser les tests par exemple. Cela permet aussi de tout simplement redéployer une application qui ne marche plus plutôt que de la réparer et ainsi sauver du temps et de l’argent.

*PORTABILITÉ*

Docker permet d’utiliser une application sur presque n’importe quel poste et offre donc une très grande portabilité.

*ORIENTATION MICRO-SERVICES*

Docker s’applique très bien à la philosophie micro-services (une façon de coder de plus en plus populaire). En effet, il est facile de séparer une application en micro-services qui seront tous séparés dans des conteneurs. Ainsi, chaque conteneur aura une tâche précise qui lui sera attribuée selon une architecture micro-services.

**Q3. Quel est l’avantage du fichier docker-compose.yml vis-à-vis l’utilisation de simples Dockerfiles? Utilisez un tableau pour comparer les fonctionnalités principales de chacun et expliquez les avantages et désavantages entre les deux solutions.**

Un docker-compose est un outil pour définir et exécuter des applications Docker multi-conteneur. Il permet de définir les services qui composent une application dans docker-compose.yml afin qu'ils puissent être exécutés ensemble dans un environnement isolé. Il permet aussi d’obtenir une application en cours d'exécution en une seule commande en exécutant “docker-compose up”. De plus, un docker-compose peut contenir un répartiteur de charge qui permet de lancer plusieurs instance du conteneur d’une image. Un Dockerfile est un simple fichier texte contenant les commandes qu'un utilisateur peut appeler pour assembler une image.

|  |  |
| --- | --- |
| **Dockerfile** | **docker-compose** |
| composé de commandes (ADD, CMD, EXPOSE, FROM, etc.) | composé de services eux-mêmes composés d’images, volumes, ports, etc. |
| docker build | docker-compose up pour démarrer l’application |
| Défini l’environnement de l’application | Défini les services de l’application |
| Utile pour créer une image personnalisée | Plusieurs environnements isolés sur un même hôte |
| Commence toujours par un FROM qui définit l’image parent | Persistance des données des volumes |
| Permet à Docker de créer automatiquement des images | Seulement les volumes modifiés sont recréés |

**Q4. Quelle est la différence entre un volume et un “bind mount” dans docker?**

Un bind mount est tout simplement un bind d‘un répertoire (ou fichier) du client à l’intérieur du conteneur Docker. Par contre, un volume est créé à l'exécution d’un containeur et permet de facilement gérer la persistance des données puisqu’on peut choisir les données qu’on désire sauvegarder. Les volumes sont entreposés dans un dossier ‘’volume’’ tandis que les bind mount peuvent être entreposés ailleurs. Les volumes sont portables entre les images mais pas les mount.

**Question de rétroaction**

**Q1. Combien de temps avez-vous passé sur le travail pratique, en heures-personnes, en sachant que deux personnes travaillant pendant trois heures correspondent à six heures-personnes. Est-ce que l'effort demandé pour ce laboratoire est adéquat ?**

Nous avons passé environ 12 heures sur ce travail pratique. Il était intéressant de comprendre le concept de docker, une nouvelle technologie qui nous sera utile dans notre parcours d'ingénierie logicielle. La charge de travail était légèrement supérieure aux TPs précédents mais était tout de même adéquate.

**Q2. Quelles difficultés avez vous rencontré lors de ce laboratoire? Important →**

Nous avons eu une erreur lors de l'exécution de la commande *“docker-compose up --build”*

**Erreur :**

/usr/local/bin/docker-compose: line 1: syntax error near unexpected token 'newline'  
/usr/local/bin/docker-compose: line: '<!DOCTYPE html>'

Ce qui nous a amené à remplacer 'uname -s' et 'uname -m'. Ceux-ci sont censés récupérer des informations sur le noyau. Si on copiez et publiez la commande pour curl et l'exécuter, cela téléchargere la page html et non l'exécutable dont nous avons besoin.

**Commande remplacée:**



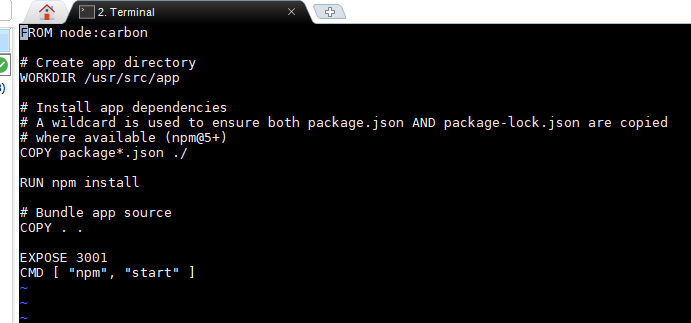
**Commande Utilisée**



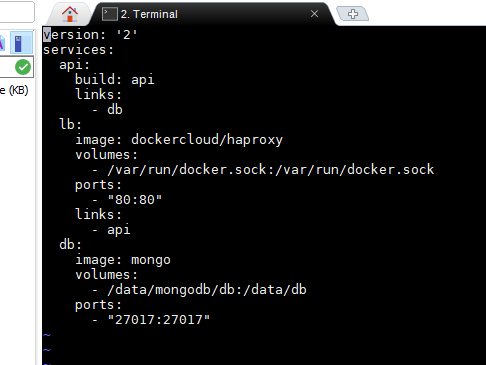
Aussi, nous n'arrivions pas à travailler dans Moba Xterm puisque la connexion à la VM ne fonctionnait pas. Nous avons donc dû tout recommencer la création de la VM. Il pourrait être pertinent d’indiquer le contenu de la ligne 33 à modifier dans la VM pour les TPs futurs puisque le contenu de cette ligne la première fois n’était pas ce qu’il devait être mais nous ne pouvions pas savoir que l’erreur était là puisque nous ne savions pas ce qu’il devait y avoir initialement (PermitRootLogin yes).

**ANNEXE**

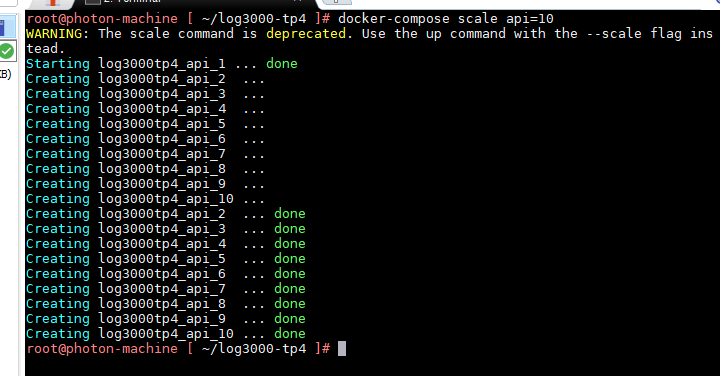
* **Dockerfile:**

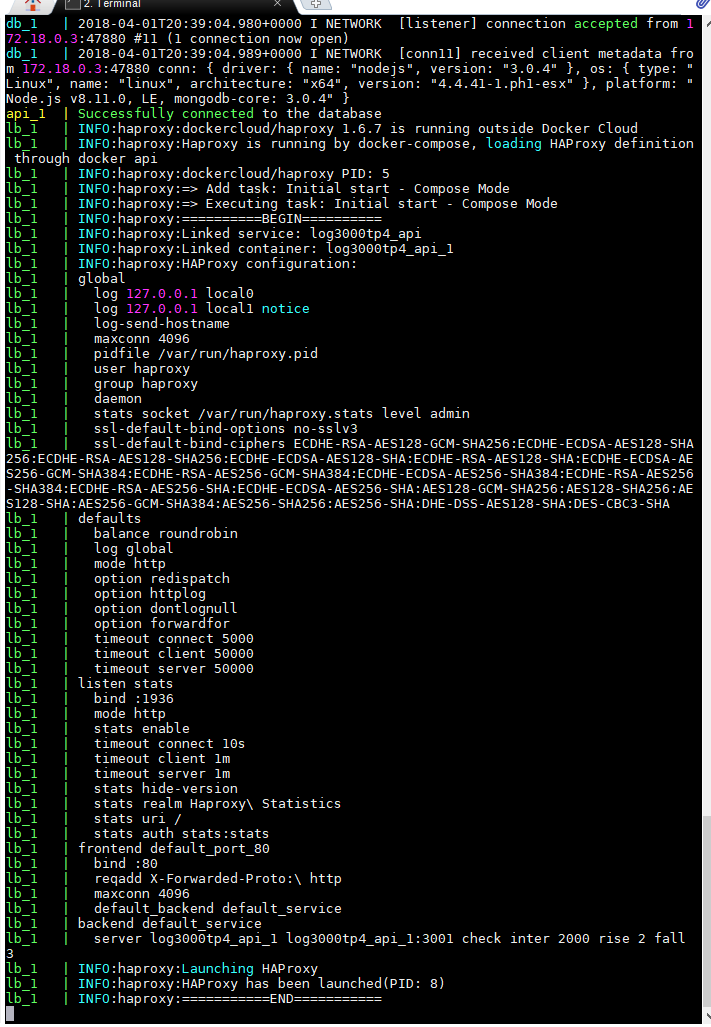
****

* **docker-compose.yml:**

****

* **Exécuter docker compose:**

****

****

**RÉFÉRENCES**

<https://actu.alfa-safety.fr/devops/conteneur-docker-fonctionnement-et-avantages-pour-heberger-ses-applications/>

<https://nodejs.org/en/docs/guides/nodejs-docker-webapp/>