

# READ ME

Fonctionnalités utilisées :

- Mysql
- PhpMyAdmin
- Apache2
- Rancher

Ports utilisés :

- Rancher : 8080
- Apache : 3000
- phpMyAdmin : 5000

## Mise en place d'une architecture applicative

### Lancer le script "init\_eval.sh"

Une fois le script exécuté, vous avez désormais créé quatre conteneurs : rancher, apache, mysql et phpmyadmin.

> Vous devez maintenant réalisé une dizaine d'étape avant de pouvoir accéder à votre rancher et vos conteneurs en mode graphique.

### ETAPE 1

Exécuter la commande **docker ps** dans un terminal

| CONTAINER ID  | IMAGE                          | COMMAND                | CREATED        | STATUS        | PORTS                            | NAMES                            |
|---------------|--------------------------------|------------------------|----------------|---------------|----------------------------------|----------------------------------|
| 10960c95f68d  | rancher/healthcheck:v0.3.1     | "/.r/r /rancher-en..." | 18 minutes ago | Up 18 minutes |                                  | r-healthcheck-healthcheck-1-ed69 |
| 4871a8be29d   | rancher/net:v0.11.2            | "/rancher-entrypoi..." | 18 minutes ago | Up 18 minutes |                                  | r-ipsec-ipsec-router-1-80c88edd  |
| 924418906260  | rancher/net:holder             | "/.r/r /rancher-en..." | 18 minutes ago | Up 18 minutes |                                  | r-ipsec-ipsec-1-e6801e0e         |
| 246186b32b83  | rancher/dns:v0.15.1            | "/rancher-entrypoi..." | 18 minutes ago | Up 18 minutes |                                  | r-network-services-metadata-dns- |
| 1-8b202551    | rancher/net:v0.11.2            | "/rancher-entrypoi..." | 19 minutes ago | Up 19 minutes |                                  | r-ipsec-ipsec-cni-driver-1-073a5 |
| e51fff5c6df9  | rancher/metadata:v0.9.1        | "/rancher-entrypoi..." | 19 minutes ago | Up 19 minutes |                                  | r-network-services-metadata-1-55 |
| 85ccfd756361  | rancher/scheduler:v0.8.0       | "/.r/r /rancher-en..." | 19 minutes ago | Up 19 minutes |                                  | r.scheduler.scheduler-1-c6b49c0b |
| 2e39f9        | rancher/network-manager:v0.7.1 | "/rancher-entrypoi..." | 19 minutes ago | Up 19 minutes |                                  | r-network-services-network-manag |
| 30d0a2100d04  | rancher/agent:v1.2.2           | "/run.sh run"          | 20 minutes ago | Up 20 minutes | 3306/tcp, 0.0.0.0:8080->8080/tcp | rancher-agent                    |
| d4d24b17ee2e  | rancher/server                 | "/usr/bin/entry /u..." | 23 minutes ago | Up 23 minutes | 0.0.0.0:3000->80/tcp             | eloquent_goldstine               |
| er-1-04c8c742 | apache                         | "/bin/sh -c 'cron ..." | 25 minutes ago | Up 25 minutes | 0.0.0.0:4000->80/tcp             | apache                           |
| e2ff4b7c2491  | phpmyadmin/phpmyadmin          | "/run.sh phpmyadmin"   | 25 minutes ago | Up 25 minutes | 3306/tcp                         | myadmin                          |
| 3c6b0b6c90b1  | mysql:latest                   | "docker-entrypoint..." | 25 minutes ago | Up 25 minutes |                                  | mysql                            |

Récupérer le nom (colonne NAMES) du conteneur **rancher/server** (colonne IMAGE)

- ici "eloquent\_goldstine"

### ETAPE 2

Exécuter la commande **docker inspect nom\_du\_conteneur | grep -i ip** dans un terminal

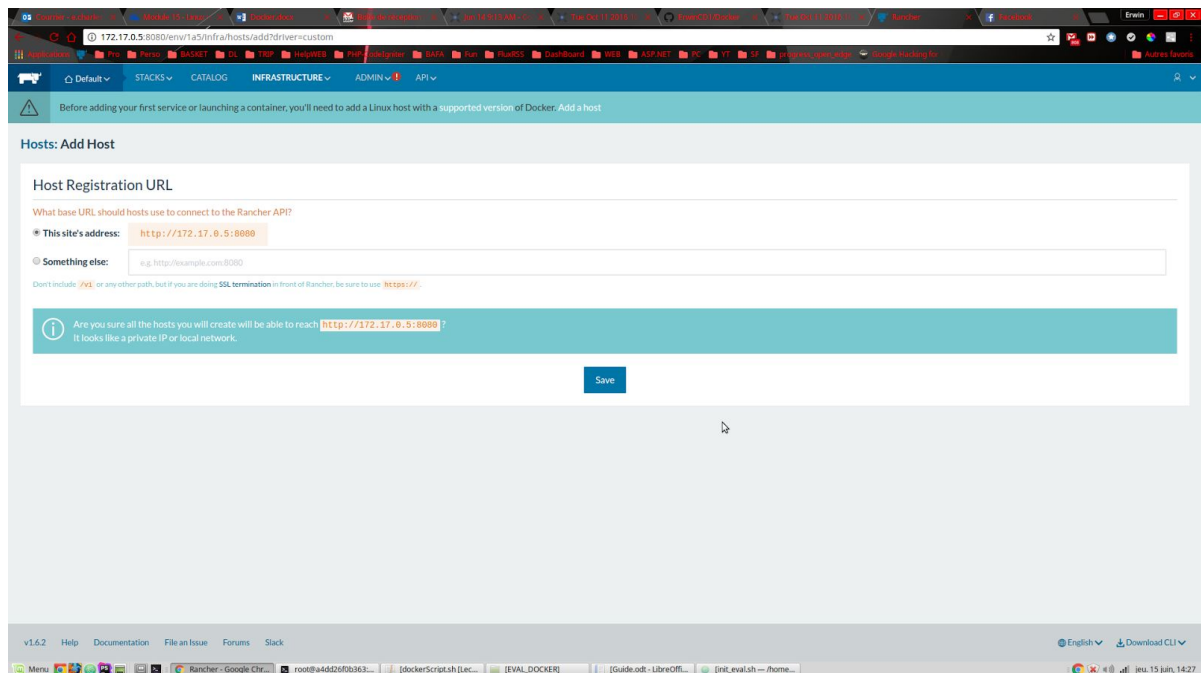
```
apprenant@poste-51 ~/Bureau/EVAL_DOCKER $ docker inspect determined_gates | grep -i ip
    "HostIp": "",
    "IpMode": "",
    "DEFAULT_CATTLE_RANCHER_COMPOSE_WINDOWS_URL=https://releases.rancher.com/compose/v0.12.5/rancher-compose-windows-386-v0.12.5.zip",
    "DEFAULT_CATTLE_RANCHER_CLI_WINDOWS_URL=https://releases.rancher.com/cli/v0.6.1/rancher-windows-386-v0.6.1.zip",
    "LinkLocalIPv6Address": "",
    "LinkLocalIPv6PrefixLen": 0,
    "HostIp": "0.0.0.0",
    "SecondaryIPAddresses": null,
    "SecondaryIPv6Addresses": null,
    "GlobalIPv6Address": "",
    "GlobalIPv6PrefixLen": 0,
    "IPAddress": "172.17.0.5",
    "IPPrefixLen": 16,
    "IPv6Gateway": "",
    "IPAMConfig": null,
    "IPAddress": "172.17.0.5",
    "IPPrefixLen": 16,
    "IPv6Gateway": "",
    "GlobalIPv6Address": "",
    "GlobalIPv6PrefixLen": 0,
```

Récupérer l'adresse IP dans la réponse de cette commande : **IPv6Gateway > IPAddress**

- ici 172.17.0.5

### ETAPE 3

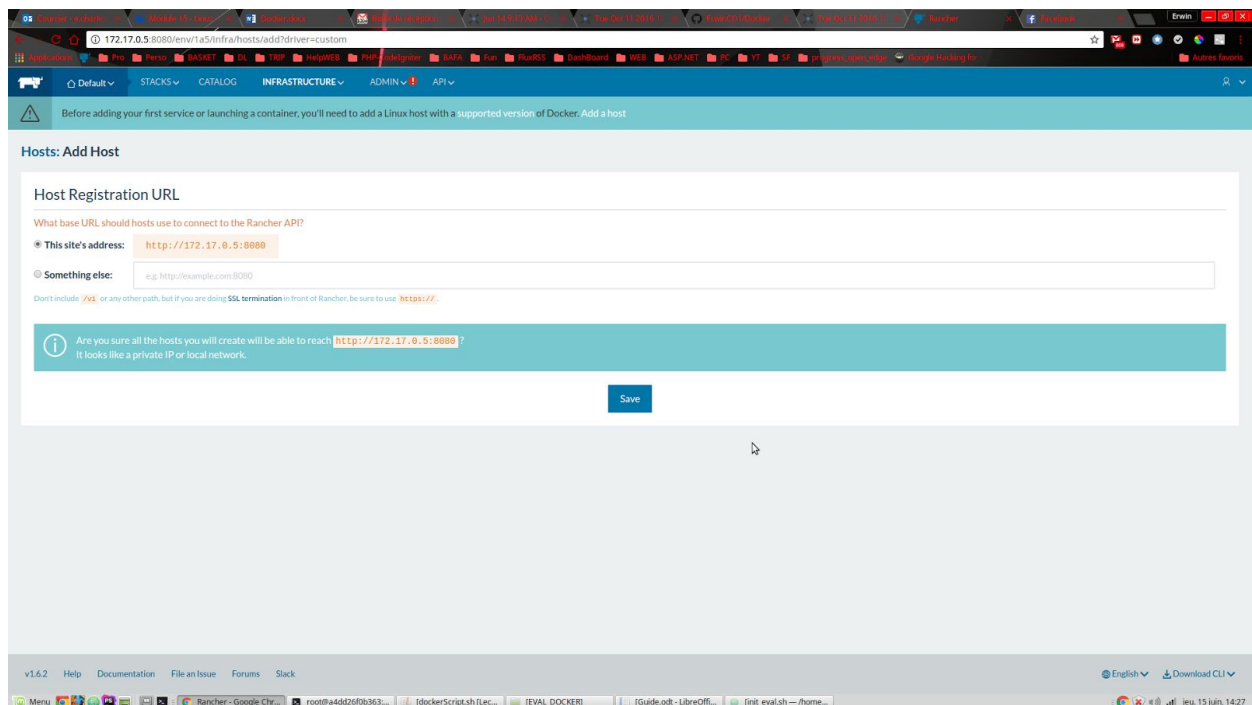
Aller sur la navigateur et entrer l'adresse IP ainsi que le port (**172.17.0.5:8080**) dans la barre d'URL. Vous devez attendre quelques minutes le temps que Rancher se charge. Après quelques minutes d'attente, vous devez arriver sur cette page.



#### ETAPE 4

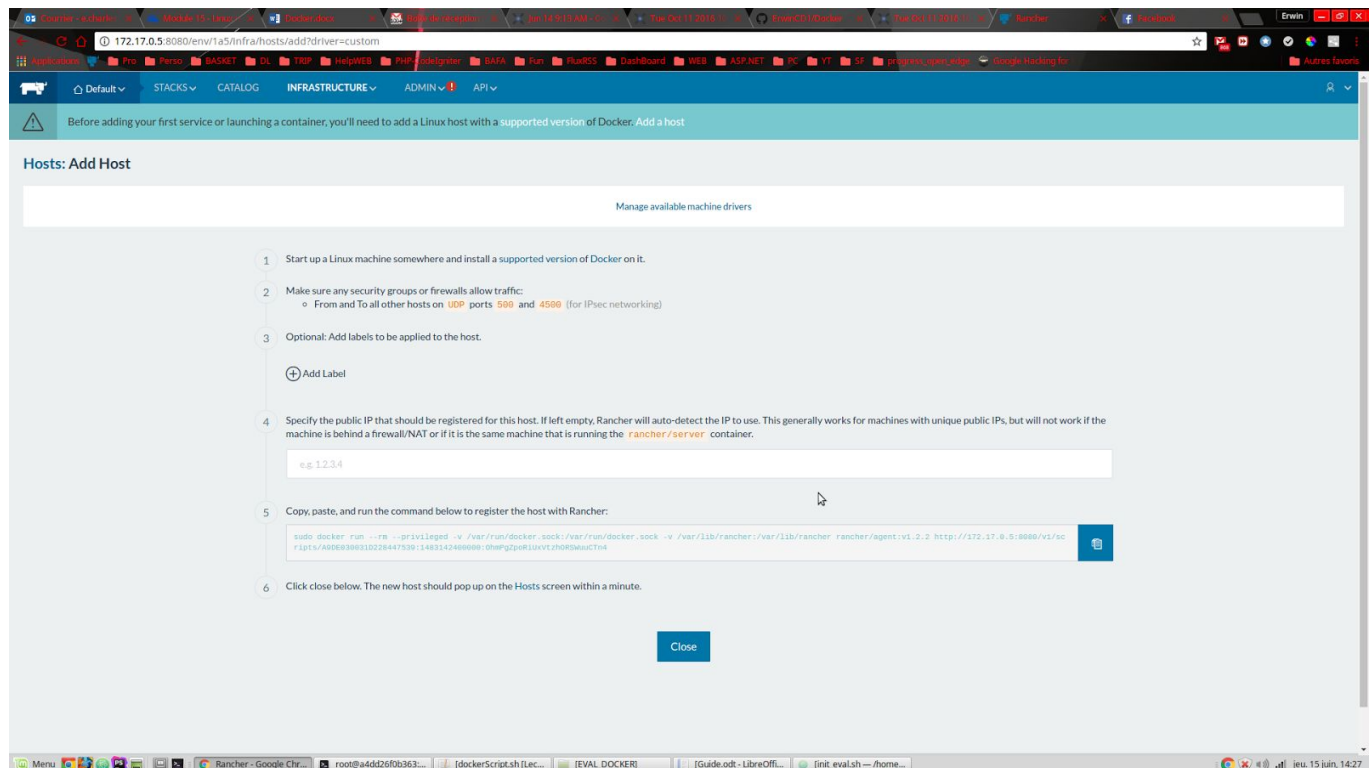
Cliquez sur ajouter un hôte “Add Host” dans la barre juste en dessous du menu en bleu (panneau attention).

Vous atterrissez sur cette page. Cliquer directement sur le bouton bleu “SAVE” en bas de la page. Cette page permettra d'utiliser l'adresse IP de rancher comme celle d'un nouvel hôte pour la connexion.



#### ETAPE 5

Une fois le bouton cliqué, cette page devrait s'afficher.



## ETAPE 6

Copier la commande qui est apparu dans le 5.

## ETAPE 7

Coller la commande dans un terminal puis exécuté la.

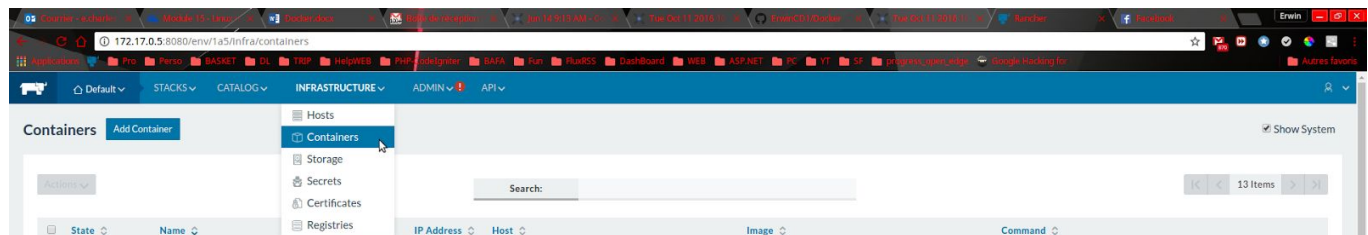
```
apprenant@pote:~$ docker run --rm --privileged -v /var/run/docker.sock:/var/run/docker.sock -v /var/lib/rancher:/var/lib/rancher rancher/agent:v1.2.2 http://172.17.0.5:8080/v1/scripts/A90E8306310228447539:148314240000:0hnp2d8uixvtzh0R5WuCTn4
Unable to find image 'rancher/agent:v1.2.2' locally
v1.2.2: Pulling from rancher/agent
b3e1c725a85f: Pull complete
4daad8bde31: Pull complete
63fe8c006a8: Pull complete
4a70713c486f: Pull complete
bd842a2105a8: Pull complete
3f7d6fd71888: Pull complete
16914729ef83: Pull complete
b9c91512c230: Pull complete
Digest: sha256:cfb7fc0d1146680ca26e3bc8cd09cc7a0274eff99f5b84246c33642a99f81da
Status: Downloaded newer image for rancher/agent:v1.2.2
INFO: Running Agent Registration Process, CATTLE_URL=http://172.17.0.5:8080/v1
INFO: Attempting to connect to: http://172.17.0.5:8080/v1
INFO: http://172.17.0.5:8080/v1 is accessible
INFO: Inspecting host capabilities
INFO: Boot2Docker: false
INFO: Host Writable: true
INFO: Token: xxxxxxxx
INFO: Running registration
INFO: Printing Environment
INFO: ENV: CATTLE_ACCESS_KEY=CE473F94B7166596BB09
INFO: ENV: CATTLE_HOME=/var/lib/cattle
INFO: ENV: CATTLE_REGISTRATION_ACCESS_KEY=registrationToken
INFO: ENV: CATTLE_REGISTRATION_SECRET_KEY=xxxxxxx
INFO: ENV: CATTLE_SECRET_KEY=xxxxxxx
INFO: ENV: CATTLE_URL=http://172.17.0.5:8080/v1
INFO: ENV: DETECTED CATTLE_AGENT_IP=172.17.0.6
INFO: ENV: RANCHER_AGENT_IMAGE=rancher/agent:v1.2.2
INFO: Launched Rancher Agent: 3bb9800e60a52d34e47141fc1b7b48c1670734b3967b5ef5a5cc1507e58e47
```

## ETAPE 8

Une fois la commande terminée, retournez sur le navigateur et rafraîchissez la page (F5).

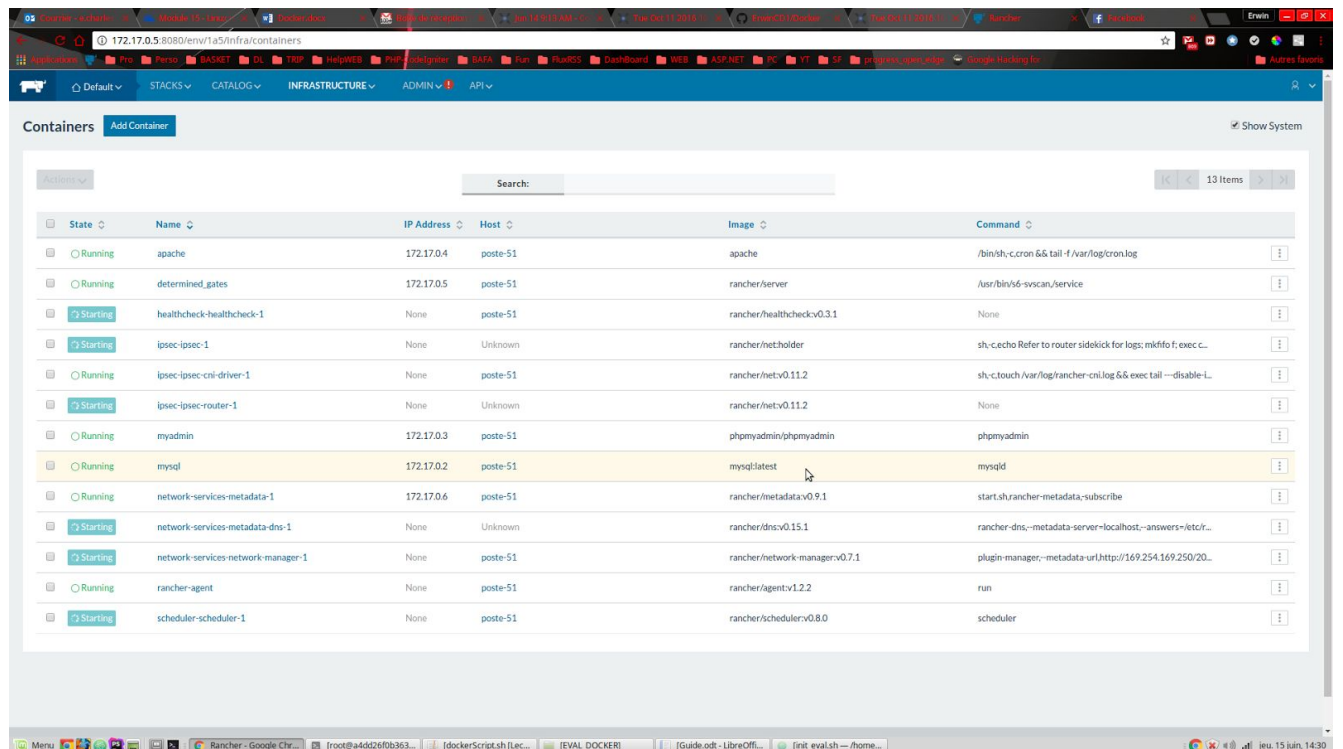
## ETAPE 9

Cliquez sur le bouton “Infrastructure” dans la barre de menu puis sur “Containers”



## ETAPE 10

Vous avez devant vous tous les conteneurs disponibles sur votre Rancher.



A partir de là, vous pouvez arrêter, supprimer et redémarrer vos conteneurs.

## Sécurité

L'utilisation d'un firewall type ufw n'est pas nécessaire puisque de base Docker ferme automatiquement tous les ports et ouvre que ceux nécessaires, c'est-à-dire que ceux que l'on a dédié aux conteneurs. De plus, si nous mettons un firewall en plus du travail que Docker fait, la performance de Docker serait ralentie.

Nous avons utilisé les paquets suivants sur Rancher afin de le sécuriser :

- logwatch : envoi d'un rapport journalier de l'analyse des fichiers journaux
- fail2ban : banni certaines adresses IP si leur authentification a échoué et est répétitive
- rkhunter : détecteur de rootkit
- chkrootkit : détecte si le système n'a pas été compromis par un rootkit

## Erreurs possibles

Si lors du redémarrage du conteneur apache depuis rancher, vous n'arrivez plus à accéder au serveur apache depuis l'url "localhost:5000" vous devez réaliser ses changements :

- Modifier les scripts "index\_eval.sh"
  - Commenter toutes les lignes sauf :
    - touch dockerFile\_apache
    - echo 'FROM httpd:latest  
WORKDIR /usr/local/apache2/htdocs  
EXPOSE 80  
RUN apt-get -y update && apt-get install -y apache2 && a2enmod  
rewrite headers && mkdir /var/www/webapp && sed -i -e  
"s/\var/www/html/\var/www/webapp/g"  
/etc/apache2/sites-available/000-default.conf && apt-get install -y  
nano cron logwatch fail2ban rkhunter chkrootkit ufw mysql-client  
ADD cronTab /etc/cron.d/hello-cron  
RUN chmod 0644 /etc/cron.d/hello-cron  
RUN touch /var/log/cron.log  
ADD index\_apache /usr/local/apache2/htdocs/index.html  
CMD cron && tail -f /var/log/cron.log' > dockerFile\_apache
    - docker build -f dockerFile\_apache -t apache .
    - docker run --name apache -p 5000:80 -d apache
- Lancer le script
- Une fois le script terminé, tester dans le navigateur l'affichage du index.html du serveur apache (localhost:5000)
  - Si un "It's work" s'affiche :
    - Relancer le script tel qu'il était au début avant toutes les modifications
    - Vous devriez arriver sur la page du serveur apache depuis l'url suivante localhost:5000
- Si vous avez toujours aucun page affiché, alors rebooter votre pc !