



Documentation Technique - Projet "Monitor the ping"

Introduction

Ce document fournit des instructions détaillées sur la façon de dockeriser une application Python de surveillance de ping, la rendre disponible sur Docker Hub, la déployer sur différents serveurs et configurer Prometheus pour surveiller notre application. Notre document aura deux parties, la première parlera des étapes pour dockeriser l'application python et dans la deuxième partie on montrera comment le récupérer et le déployer cela.

💡 Par abus on utilisera le terme «
dockeriser »

Partie 1 : Dockeriser l'application

Dans le sujet on nous a fourni une application [ping-exporter.py](#) . Il nous faut créer le fichier Dockerfile. Ceci se fera en quelques lignes:

```
FROM alpine:3.9
RUN apk add python2 py-pip fping
COPY ping-exporter.py /opt/ping-exporter.py
WORKDIR /opt
CMD ["python2", "/opt/ping-exporter.py"]
```

Une fois le fichier créé, on va le Builder et le Run, mais on se rassure que les fichiers sont dans le même répertoire.

```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker build -t ping-exporter .
[+] Building 1.5s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 228B
=> [internal] load metadata for docker.io/library/alpine:3.9
```

Une fois le Build OK, on le run.

```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker run -d ping-exporter
bf76ab701dab6125791b05f597e7288521b439deb75a4f2501c9e683553741ba
```

Ensuite il faut créer une variable d'environnement **PORT** et rediriger pour qu'il écoute sur le port 80. Pour cela on doit modifier l'application [ping-exporter.py](#).

```
FROM alpine:3.9
RUN apk add python2 py-pip fping
COPY ping-exporter.py /opt/ping-exporter.py
WORKDIR /opt
ENV PORT =80
CMD ["python2", "/opt/ping-exporter.py", "-p", "80"]
```

Après modification, nous devons refaire le Build avec la commande:

```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker run -d ping-exporter
bf76ab701dab6125791b05f597e7288521b439deb75a4f2501c9e683553741ba
```

Ensuite on run en tenant compte de la variable d'environnement

docker run -e "PORT=80" ping-exporter

```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker run -d -e "PORT=80" ping-exporter
8b6b2d6ca94fb7edc41e973bca964d7bce4fe259c9fd62d472d7b359eb835824
```

Par la suite on va créer le fichier **docker-compose.yml** comme suit et toujours dans le même répertoire que les fichiers précédents:

```
version: '3'
services:
  exporter:
    image: ping-exporter
```

```
environment:
```

```
PORT: '80'
```

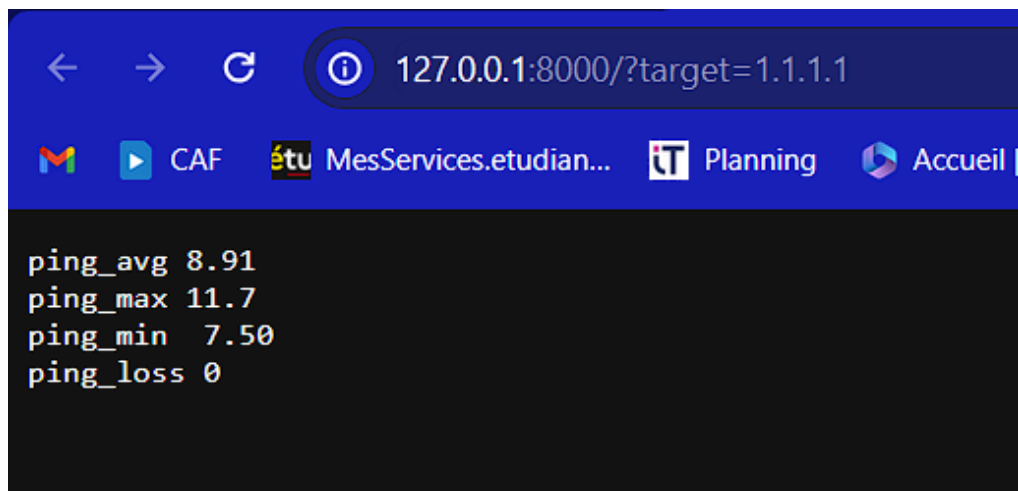
```
ports:
```

```
- "8000:80"
```

On va utiliser la commande `docker compose up -d` pour run cela.

```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker-compose up -d
[+] Running 1/1
✓ Container 3dct-subject-exporter-1 Started
```

Maintenant on se rend sur dans le navigateur et on saisi l'url suivante:
`http://127.0.0.1:8000/?target=1.1.1.1` et on aura ce résultat 📌 :



Ceci nous confirme que cela fonctionne correctement.

Dans la suite on va utiliser Prometheus pour surveiller le ping.

On nous a fourni dans le projet le fichier `prometheus.yml`, on va aussi configurer son fichier docker compose et monter le port `9090` de Prometheus en le redirigeant vers le port `9000` et le mettre dans un volume.

Dans le même fichier `docker-compose.yml`, on va rajouter les lignes suivantes:

```
prometheus:
```

```
image: prom/prometheus:v2.36.2
```

```
ports:
```

```
- "9000:9090"
```

```
volumes:
```

```
- ./prometheus.yml:/etc/prometheus/prometheus.yml
```

```
command:
```

```
- "--config.file=/etc/prometheus/prometheus.yml"
- "--storage.tsdb.path=/prometheus" - "--
web.console.libraries=/usr/share/prometheus/console_libraries"
- "--web.console.templates=/usr/share/prometheus/consoles"
```

En suite on modifier la dernière ligne du fichier `prometheus.yml` .



`["CONTAINER-ADDRESS:CONTAINER-PORT"]`

On modifie par les valeurs en question: `["172.18.0.2:80"]`

Et dès lors que c'est terminé on run la commande docker

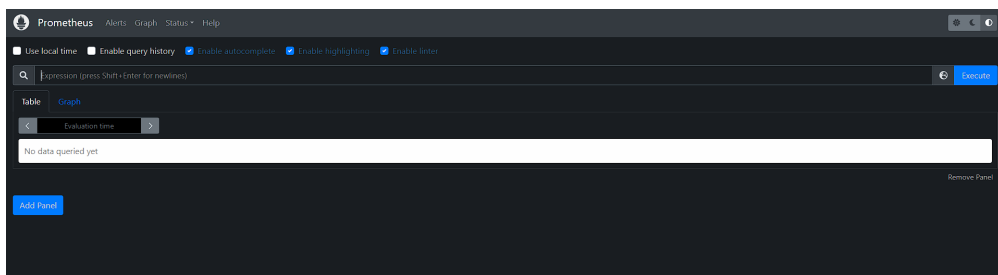


`compose up -d`

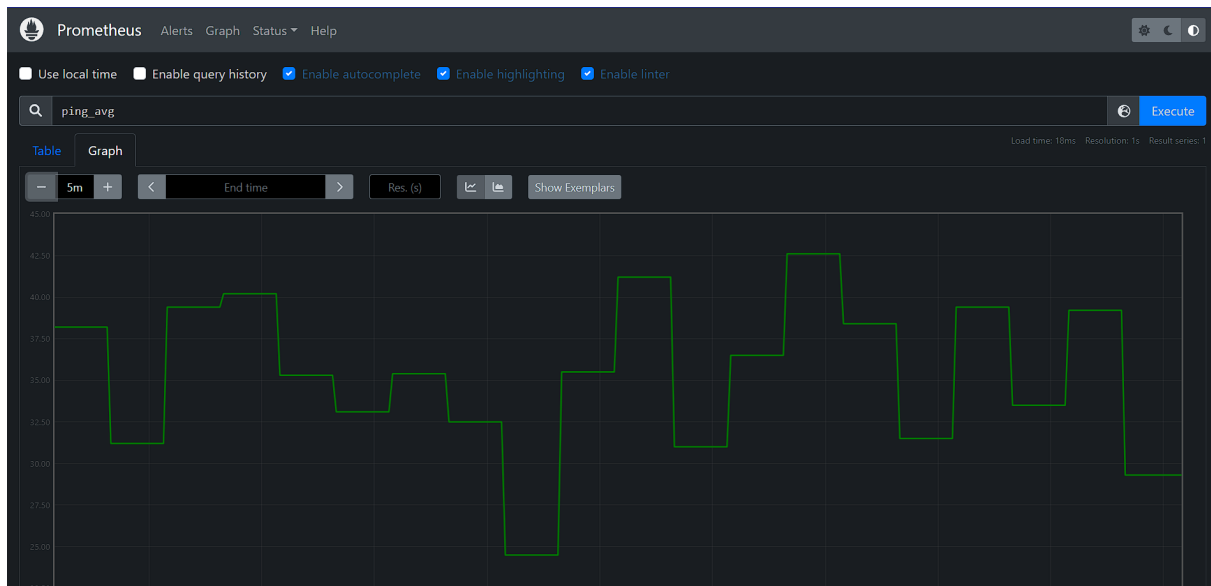
```
PS C:\Users\scarf\Desktop\3DCT-Subject> docker compose up -d
[+] Running 13/1
✓ prometheus 12 layers [#####] 0B/0B Pulled
[+] Running 2/2
✓ Container 3dct-subject-exporter-1 Running
✓ Container 3dct-subject-prometheus-1 Started
```

Et dès que la configuration est terminé, on va dans le navigateur puis on rentre l'url suivante: `http://127.0.0.1:9000`

Si tout est okay, on aura la page d'accueil de Prometheus:



On va choisir `Graph` et exécuter la recherche de `ping_avg` et régler sur `5 min`. Une fois que c'est fait on aura cette jolie page.



C'est très beau n'est-ce pas 😊?

Maintenant à vous de configurer ce dont vous aimerez.

Partie 2 : Comment avoir les fichiers et les déployer?

Pour la suite, ce projet est disponible sur mon GitHub et l'image du conteneur se trouve sur mon Docker Hub.

1. Téléchargement des fichiers sur GitHub

Clonez le dépôt GitHub contenant les fichiers du projet en utilisant la commande suivante dans votre terminal :



```
git clone https://github.com/Scarfacemoignon/ping-exporter.git
```

2. Une fois le clonage terminé, vous aurez accès aux fichiers suivants :

- **Dockerfile** : Contient les instructions pour construire l'image Docker de l'application.
- **docker-compose.yml** : Définit les services Docker à exécuter.

- `prometheus.yml` : Fichier de configuration pour Prometheus.
 - `ping-exporter.py` : **Notre application python.**
3. On télécharge l'image Docker depuis mon Docker Hub grâce à la commande:

💡 `docker pull scarface05/ping-exporter:1.1`

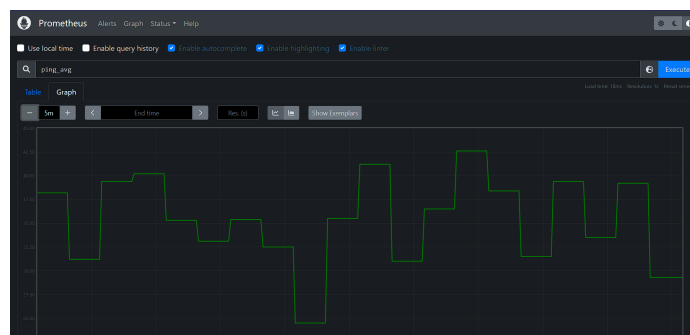
4. Ensuite on va run l'image grâce à la commande

💡 `docker run --name ping-exporter -e "PORT=80" scarface05/ping-exporter:1.1`

5. Pour finir on run la commande docker compose suivante:

💡 `docker compose up -d`

Si tout est OK ceci fonctionnera très bien.



Conclusion

Une fois que ces étapes sont suivies, l'application de surveillance de ping sera dockerisée et prête à être déployée sur différents serveurs.

Liens utiles :

<https://github.com/Scarfacemoignon/ping-exporter>



[Documentation Technique - Projet "Monitor the ping"](#)

Me contacter par mail : Dierry-Nevyl.TCHUENDOM-OUAMBO@ecole-it.com

Par phone au 06 16 41 90 52