

TP DEVOPS :

Objectifs :

Créer un repository Github Créer un wrapper qui retourne la météo d'un lieu donné avec sa latitude et sa longitude (passées en variable d'environnement) en utilisant openweather API dans le langage de programmation de notre choix (bash, python, go, nodejs, etc).

Ensuite, packager notre code dans une image Docker Mettre à disposition son image sur DockerHub et mettre à disposition notre code dans un repository Github.

Contenu du code :

Fichiers :

- [docker.dockerfile](#) : Les Dockerfiles sont des fichiers qui permettent de construire une image Docker adaptée à nos besoins, c'est pour cela qu'on l'utilise pour créer notre environnement.

[Code](#) :

```
🐚 docker.dockerfile > ...
1  FROM python:3.8-alpine
2  RUN mkdir /app
3  ADD . /app
4  WORKDIR /app
5  RUN pip install -r requirements.txt
6  CMD ["python", "tp_devops_api.py"]
```

- [tp_devops_api](#) : Le wrapper est nécessaire pour faire l'appel à notre API, pour cela on utilise le os.environ pour récupérer les variables depuis la commande docker -env. A l'aide du request on peut récupérer les données des villes qu'on veut et les afficher au format souhaité (JSON).

[Code](#) :

```

tp_devops_api.py > ...
1  import os
2  from datetime import datetime as dt
3  import requests
4  import json
5  api_key = "149a0e874b68e72881cb912046bd4bc6"
6  lat = os.environ['LAT']
7  lon = os.environ['LONG']
8  url = "https://api.openweathermap.org/data/2.5/weather?lat=%s&lon=%s&appid=%s&units=metric" % (lat, lon, api_key)
9  response = requests.get(url)
10 data = json.loads(response.text)
11 print(data)
12

```

- [requirements.txt](#) : Il s'agit de la liste des paquets Python dont l'installation est requise dans un environnement virtuel pour que l'application s'exécute correctement.

Open weather API :

OpenWeatherMap est un service en ligne, appartenant à OpenWeather Ltd, qui fournit des données météorologiques mondiales via API, y compris des données météorologiques actuelles, des prévisions, des prévisions actuelles et des données météorologiques historiques pour n'importe quel emplacement géographique.

Pour faire l'appel à l'API on utilise le lien suivant :

```

"https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid={API key}"

```

Fonctionnement du projet :

Commande :

- Build :

```

PS C:\Users\azagh\Desktop\devops> docker build . -f .\docker.dockerfile -t tp1

```

- Execute :

```
PS C:\Users\azagh\Desktop\devops> docker run --env LAT="5.902785" --env LONG="102.754175" --env API_KEY=149a0e874b68e72881cb912046bd4bc6 tp1
```

- Résultat obtenu :

```
PS C:\Users\azagh\Desktop\devops> docker run --env LAT="5.902785" --env LONG="102.754175" --env API_KEY=149a0e874b68e72881cb912046bd4bc6 tp1
{'coord': {'lon': 102.7542, 'lat': 5.9028}, 'weather': [{'id': 802, 'main': 'Clouds', 'description': 'scattered clouds', 'icon': '03n'}], 'base': 'stations', 'main': {'temp': 27.45, 'feels_like': 29.96, 'temp_min': 27.45, 'temp_max': 27.45, 'pressure': 1009, 'humidity': 73, 'sea_level': 1009, 'grnd_level': 982}, 'visibility': 10000, 'wind': {'speed': 4.41, 'deg': 128, 'gust': 4.82}, 'clouds': {'all': 29}, 'dt': 1654437803, 'sys': {'country': 'MY', 'sunrise': 1654383247, 'sunset': 1654428060}, 'timezone': 28800, 'id': 1736405, 'name': 'Jertih', 'cod': 200}
PS C:\Users\azagh\Desktop\devops>
```

- Tag :

```
PS C:\Users\azagh\Desktop\devops> docker tag tp1 anaszaghloul/tp1
```

- Connexion to Docker hub :

```
Password:
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
```


- Push :


```


PS C:\Users\azagh\Desktop\devops> docker push anaszaghloul/tp1
Using default tag: latest
The push refers to repository [docker.io/anaszaghloul/tp1]
ca4fb64399c8: Pushed
5f70bf18a086: Pushed
295fa4da795a: Pushed
329573533df5: Pushed
4531d365dde0: Mounted from library/python
9e567d1bb30a: Mounted from library/python
56bb29f4ee33: Mounted from library/python
09c126bb3acd: Mounted from library/python
24302eb7d908: Mounted from library/python
latest: digest: sha256:d9634b38cfa3e305101382cabdd44563cb6122aa4b8cc1d4592f9461b9a24d38 size: 2199

```

- in Docker HUB : (check)




anaszaghloul / tp1

This repository does not have a description 

 Last pushed: 5 minutes ago

Tags and Scans
VULNERABILITY SCANNING - DISABLED [Enable](#)

This repository contains 1 tag(s).

TAG	OS	PULLED	PUSHED
 latest		---	5 minutes ago

[See all](#)

Bonus :

- recherche de vulnérabilité avec trivy :

```

anas@DESKTOP-1DM2KEV:/mnt/c/Users/azagh$ trivy image anasazghloul/tp1
2022-06-05T21:55:39.126+0200 INFO Need to update DB
2022-06-05T21:55:39.127+0200 INFO DB Repository: ghcr.io/aquasecurity/trivy-db
2022-06-05T21:55:39.129+0200 INFO Downloading DB...
32.34 MiB / 32.34 MiB [-----] 100.0%
2022-06-05T21:56:14.137+0200 INFO Detected OS: alpine
2022-06-05T21:56:14.140+0200 INFO This OS version is not on the EOL list: alpine 3.16
2022-06-05T21:56:14.141+0200 INFO Detecting Alpine vulnerabilities...
2022-06-05T21:56:14.156+0200 INFO Number of language-specific files: 1
2022-06-05T21:56:14.156+0200 INFO Detecting python-pkg vulnerabilities...

anasazghloul/tp1 (alpine 3.16.0)

Total: 0 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 0, CRITICAL: 0)

Python (python-pkg)

Total: 0 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 0, CRITICAL: 0)

anas@DESKTOP-1DM2KEV:/mnt/c/Users/azagh$

```

- handolint :

```

anas@DESKTOP-1DM2KEV:/mnt/c/Users/azagh/Desktop/devops$ docker run --rm -i hadolint/hadolint < docker.dockerfile
-:5 DL3042 warning: Avoid use of cache directory with pip. Use `pip install --no-cache-dir <package>`

```

- Aucune données sensibles stockées dans l'image :
Il suffit de changer notre API KEY par celui de l'environnement comme ça personne ne le verra.

```

tp_devops_api.py > ...
1 import os
2 from datetime import datetime as dt
3 import requests
4 import json
5 api_key = os.environ['API_KEY']
6 lat = os.environ['LAT']
7 lon = os.environ['LONG']
8 url = "https://api.openweathermap.org/data/2.5/weather?lat=%s&lon=%s&appid=%s&units=metric" % (lat, lon, api_key)
9 response = requests.get(url)
10 data = json.loads(response.text)
11 print(data)
12
13

```