

3DCT

Fichiers à joindre

[ping-exporter.py](#)

[prometheus.yml](#)

Subject

You are working on monitoring ping on your server.

Your development team developed a script in python2 to achieve this.

As a good DevOPS, you want to Dockerize the project to make it portable and available on all servers including the most recent server's running Rocky Linux 9

You will instanciate a Prometheus, the ****an open-source systems monitoring and alerting toolkit originally built by SoundCloud.

Prometheus will scrape the metrics from ping exporter every 15s.

Building the exporter (6 points)

The developer team run away from the company, they gave as instructions the following note:

Our application run on Alpine version 3.9

In this alpine installation, we just install python2, py-pip and fping with the following command:

```
apk add python2 py-pip fping
```

Then, put our python script: `ping-exporter.py` to `/opt/ping-exporter.py`

Finally, just run this script by typing: `python2 /opt/ping-exporter.py`

You must create the Dockerfile and put the required instruction. The source code of the python script was send alongside this PDF subject

Making the port an environment variables (4 points)

After investigating in the script, it looks like if you change the start command to: `python2 /opt/ping-exporter.py -p 80`, you can change the port of the app by passing and pass it with an variable environment called: `PORT`.

In the SysAdmin dream, they can run your image like this:

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

Find a solution to make this possible.

Run the exporter from docker-compose (3 points)

Please provide a docker-compose.yml which gonna run your freshly builded image with the following properties:

- Name of the service: `exporter`
- Environment variables:
 - `PORT` equal to 80.
- Ports: 8000 port of your computer to the port 80 of your container.

💡 To ensure this is working, go to `http://127.0.0.1:8000/?target=1.1.1.1`

You should obtain something like:

ping_avg 28.2 ping_max 30.7 ping_min 25.2 ping_loss 0

Running Prometheus (3 points)

SysAdmin team, juste give you a “example” of prometheus docker-compose file.

```
version: "3"

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

However, there is some missing parts in this file.

For testing purpose, you will mount the 9090 port of the prometheus container to your 9000 port.

And you must also mount the `prometheus.yml` (provided with the subject) in configuration

Prometheus configuration (2 points)

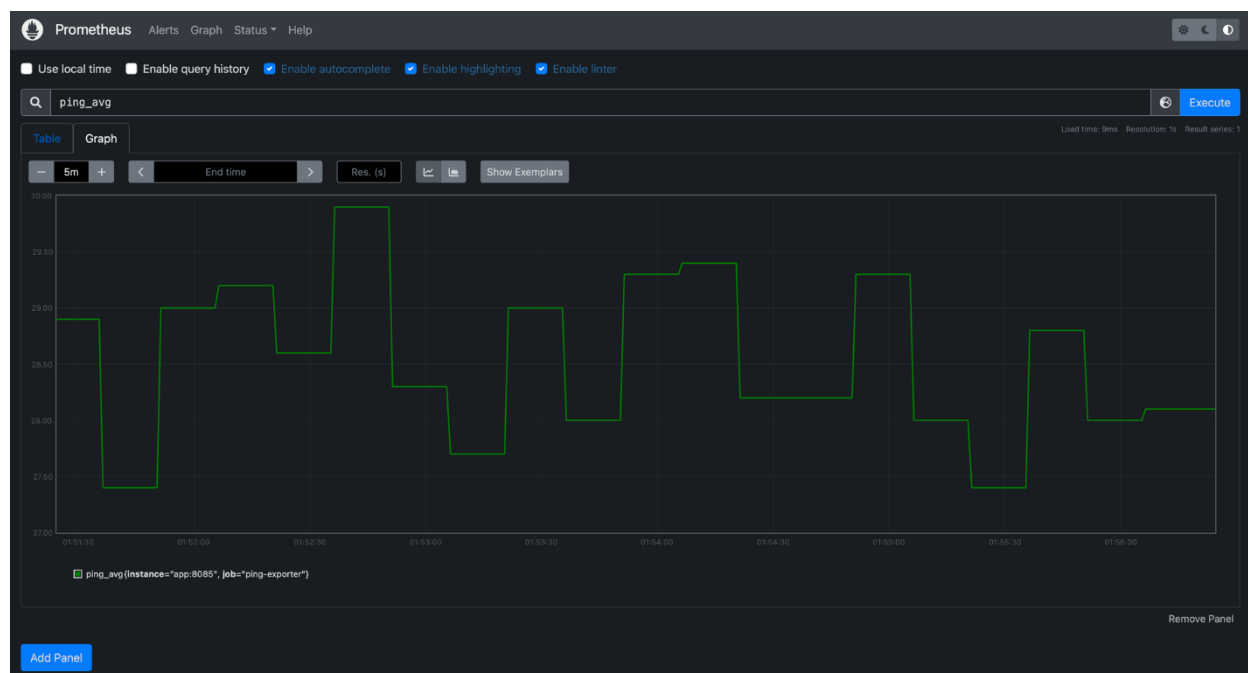
The provided prometheus.yml must be configured, you must update the configuration the good one according to your docker-compose configuration:

On the last line of the file, you must replace :

```
CONTAINER-ADDRESS:CONTAINER-PORT
```

with the good value.

To ensure this is working, start your compose and go to <http://127.0.0.1:9000>. You should obtain something like:



Redaction quality (2 points)

Your documentation must be clear and well presented, if it's not the case, you will not have all points.

Rules

This project must be your creation and done by your own. Plagiarism or copy / past are forbidden and will result of a 0.

This is project can be done in group of 2 students maximum.

Each group containing more than 2 members will be discarded and not evaluated.

Delivery

You must deliver a `ZIP` containing:

- your Dockerfile
- a Technical documentation in PDF explaining your work.

⚠ File extension should be strictly respected. If you deliver file in a different format than expected, the file will be entirely discarded.