

Inhaltsverzeichnis

- InfluxDB 2.0 mit Docker und Python** 3
 - Installieren von Grafana*** 4
 - Python Anbindung*** 4
- Kommandos** 5

InfluxDB 2.0 mit Docker und Python

Zuerst mal erzeugt man sich in einem leeren Verzeichnis ein Docker Setup- Script. In diesem Verzeichnis wird InfluxDB dann auch alle Daten speichern, darum sollte die Partition **ausreichend** Platz bieten

[run_docker.sh](#)

```
docker run -d -p 8086:8086 \  
  --name influxdbv2 \  
  -v $PWD/data:/var/lib/influxdb2 \  
  -v $PWD/config:/etc/influxdb2 \  
  -e DOCKER_INFLUXDB_INIT_MODE=setup \  
  -e DOCKER_INFLUXDB_INIT_USERNAME=yourInfluxAccount \  
  -e DOCKER_INFLUXDB_INIT_PASSWORD=yourInfluxPassword \  
  -e DOCKER_INFLUXDB_INIT_ORG=chooseOrgName \  
  -e DOCKER_INFLUXDB_INIT_BUCKET=chooseBucketName \  
  influxdb:2.0
```

Achtung: Die Volume- Mount Directories müssen vorhanden und beschreibbar sein.

Hier als Compose- File (dabei ggf. die RETENTION- Zeit anpassen, je nachdem, wie lange die Daten gehalten werden sollen)

[docker-compose.yaml](#)

```
version: "3.5"  
  
services:  
  influxdb2:  
    image: influxdb:latest  
    network_mode: "bridge"  
    container_name: influxdb2  
    ports:  
      - "8086:8086"  
    volumes:  
      - type: bind  
        source: /volumel/docker/influxdb2/data  
        target: /var/lib/influxdb2  
      - type: bind  
        source: /volumel/docker/influxdb2/config  
        target: /etc/influxdb2  
    environment:  
      - DOCKER_INFLUXDB_INIT_MODE=setup  
      - DOCKER_INFLUXDB_INIT_USERNAME=yourInfluxAccount  
      - DOCKER_INFLUXDB_INIT_PASSWORD=yourInfluxPassword  
      - DOCKER_INFLUXDB_INIT_ORG=chooseOrgName  
      - DOCKER_INFLUXDB_INIT_BUCKET=chooseBucketName
```

```
- DOCKER_INFLUXDB_INIT_RETENTION=1w
- DOCKER_INFLUXDB_INIT_ADMIN_TOKEN=randomTokenValue
restart: always
```

Wenn man selber kein Token angegeben hatte (DOCKER_INFLUXDB_INIT_ADMIN_TOKEN), erzeugt influxdb selber eins, was man sich dann mit

```
$ docker exec influxdb2 influx auth list \
  --user $USERNAME \
  --hide-headers | cut -f 3
```

abholen kann.

Achtung: Falls man auch Telegraph als Daten- Collector installieren möchte, muß man sich durch die Konfiguration quälen:

<https://www.blackvoid.club/grafana-8-influxdb-2-telegraf-2021-monitoring-stack/>

Installieren von Grafana

[docker-compose.yaml](#)

```
version: "3.5"

services:
  grafana8:
    image: grafana/grafana:latest
    network_mode: "bridge"
    container_name: grafana8
    volumes:
      - /yourGrafanafolder:/var/lib/grafana
    ports:
      - "3000:3000"
    restart: always
```

Python Anbindung

Beim Installieren aufpassen, das man den Treiber für InfluxDB 2.0 erwischt (influxdb_client), und nicht den alten (influxdb) für V 1.x

```
sudo pip3 install influxdb_client
```

dann kann's losgehen:

[influx_test.py](#)

```
#
https://www.influxdata.com/blog/getting-started-with-python-and-influxdb-v2-0/

from influxdb_client import InfluxDBClient, Point
from influxdb_client.client.write_api import SYNCHRONOUS

token='randomTokenValue'
org='chooseOrgName'
bucket='chooseBucketName'
with InfluxDBClient(url='http://localhost:8086', host='localhost',
token=token, org=org, debug=False) as client:

    """
    Create client that writes data in batches with 50_000 items.
    """

    with client.write_api( write_options=WriteOptions(SYNCHRONOUS,
batch_size=50_000, flush_interval=10_000)) as write_api:
        for item in op.get_device(device_id, startUnixtime,
endUnixtime):
            if not item[4]: # no error
                point = Point('vehicle').tag('device',
device_id).field(item[1], item[3]).time(time=item[0])
                write_api.write(bucket=bucket, record=point)
```

Kommandos

Alle Bucket Daten löschen

```
docker exec influxdb2 influx delete \
--org chooseOrgName \
--bucket chooseBucketName \
--start 2020-03-01T00:00:00Z \
--stop $(date +"%Y-%m-%dT%H:%M:%SZ")
```

From:
<http://koehlers.de/wiki/> - **Steffen Köhlers Online- Bastelbuch**

Permanent link:
<http://koehlers.de/wiki/doku.php?id=pc:influxdbdocker>

Last update: **2021/10/18 15:00**

