

Uputstvo za pokretanje projekta

Prvo pokretanje

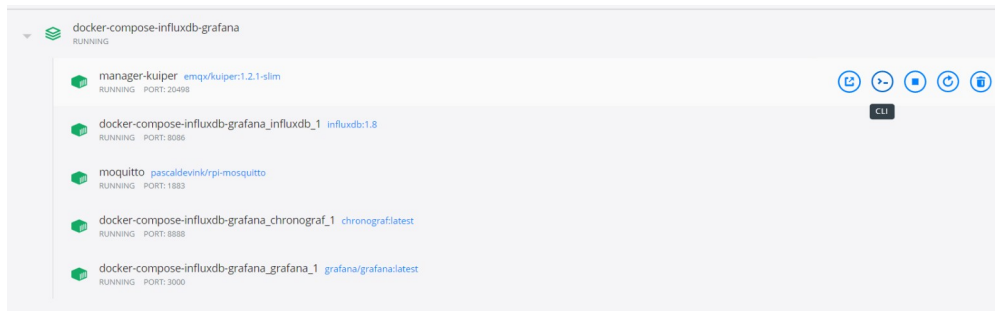
1. Pokretanje docker-compose fajla koji sadrži influx-db, mosquitto broker i Kuiper.

Pozicioniraš se u `C:\Users\danil\Documents\UbiComp\docker-compose-influxdb-grafana` i onda ideš `docker-compose up`.

2. Kreiraj pravilo u Kuiperu:

1. Ideš na Docker desktop i nađeš Kuiper i izabereš CLI opciju.

Ili `docker exec -it manager-kuiper /bin/sh` u shellu



2. Tu uneseš i izvršiš ovu naredbu: `bin/kuiper create stream demo '(gm float, temp float, light float) WITH (FORMAT="JSON", DATASOURCE="devices/checkMeasurement")'`

3. Na endpoint: **POST** <http://localhost:9081/rules> pošalješ body:

```
{
  "id": "rule1",
  "sql": "SELECT * from demo where temp > 60 or (gm > 7 AND light < 150)",
  "actions": [
    {
      "mqtt": {
        "server": "tcp://mosquitto:1883",
        "topic": "devices/action",
        "clientId": "demo_001"
      }
    },
    {
      "log": {}
    }
  ]
}
```

Primer sa cURL-om:

```
curl --location --request POST 'http://localhost:9081/rules' \
--header 'Content-Type: application/json' \
--data-raw '{
  "id": "rule1",
  "sql": "SELECT * from demo where temp > 60 or (gm > 7 AND light < 150)",
  "actions": [
    {
      "mqtt": {
        "server": "tcp://moquitto:1883",
        "topic": "devices/action",
        "clientId": "demo_001"
      }
    },
    {
      "log": {}
    }
  ]
}'
```

3. Pokreni **sensorImitation** aplikaciju. Promeni samo ip adresu hosta

SensorMeasurementCreatedPublisher.java linija 20:

IMqttClient publisher = new MqttClient("tcp://<INSERT_HOSTNAME>:1883",publisherId);

4. Pokreni **usmeni** aplikaciju na Raspberry Pi-ju.

Kako se čiste podaci iz Dockera?

0. powershell

1. docker-compose down

2. docker rm -f \$(docker ps -a -q)

3. docker volume rm \$(docker volume ls -q)

Svako sledece pokretanje

0. NE CISTI PODATKE IZ DOKERA

1. Otvori terminal i izvrši:

pi@raspberrypi:~/Desktop/docker-compose-influxdb-grafana \$ sudo docker-compose up

2. Otvori terminal novi i izvrši:

pi@raspberrypi:~/Desktop/usmeni \$ java -jar target/usmeni-0.0.1-SNAPSHOT.jar

Note: Ako se naprave bilo kakve promene u java aplikaciji, moras da ponovo napravis JAR naredbom:

pi@raspberrypi:~/Desktop/usmeni mvn clean install package

Indikacije uspesno pokrenutog backenda:

```
iApplication      : Started creating dbs.  
2021-07-06 07:01:25.998 INFO 21600 --- [      main] c.u.elfak.usmeni.UsmeniApplication      :  
Finished creating dbs.  
APP READY org.eclipse.paho.client.mqttv3.MqttClient@6bb9bf
```

Ovim se pokrece java aplikacija

3. Pokreni [frontend aplikaciju](#) i promeni IP adresu hosta.

Pokreće se naredbom: `npm start`

4. Promeni IP adresu hosta i pokreni [sensorImitationApplication](#)

Testovi bezbednosti

1. XSS/SQL injection

Note: Zbog toga što je FE pisan u Reactu, koji automatski sanitizira input korisnika, klasični napadi nisu mogući (napadi da se samo prosledi u input element `<script> alert() </script>` se samo renderuju kao klasični stringovi.

SVE TESTOVE RADI U MOZILLI!!!!

Primer XSS napada

1. Na FE postoji link [My Website](#) koji vodi na google.rs stranicu.
2. Nakon učitavanja stranice idi u Inspect sekciju
3. Nađi `<a>` tag i ubaci skriptu `javascript:alert('Hacked!');`
4. Klikni na My Website

Primer SQL injection

1. Otvori inspect tab
2. Klikni jednom na REFRESH opciju na FE
3. Nađi GET request koji ide na BE na sensors endpoint i gađa senzor sa id-jem 1129
4. Idi s desne strane na opciju Resend i promeni URL u:
`http://localhost:8080/api/sensors/1129%20AND%20%20value%20%3E%2080`
tj u postojeći request samo dodaj: `%20AND%20%20value%20%3E%2080`
5. Resenduj zahtev

2. Insecure wireless communications (MiM)

Nemam treći računar.

3. Unprotected communications

1. Otvori Wireshark program na laptopu
2. Dupli klik na WLAN opciju
3. U filter *Apply a display filter ...* unesi **ip.addr == <IP adresa Raspberry PI-ja>**
4. Čitaj saobraćaj