



SAE 23 - Partie Node-Red, InfluxDB et Grafana

Table des matières

- 1. Installation de Node-Red.....3
- 2. Configuration de MQTT dans Node-Red..... 4
- 3. Configuration de Node-Red-Dashboard.....5
- 4. Mise en place d’InfluxDB.....7
- 5. Connexion à InfluxDB depuis Node-Red..... 8
- 6. Configuration de Grafana..... 10

Il faut d'abord installer le paquet docker.io afin de créer les dockers nécessaires pour cette partie de la SAE :

```
apt install docker.io
```

1.Installation de Node-Red

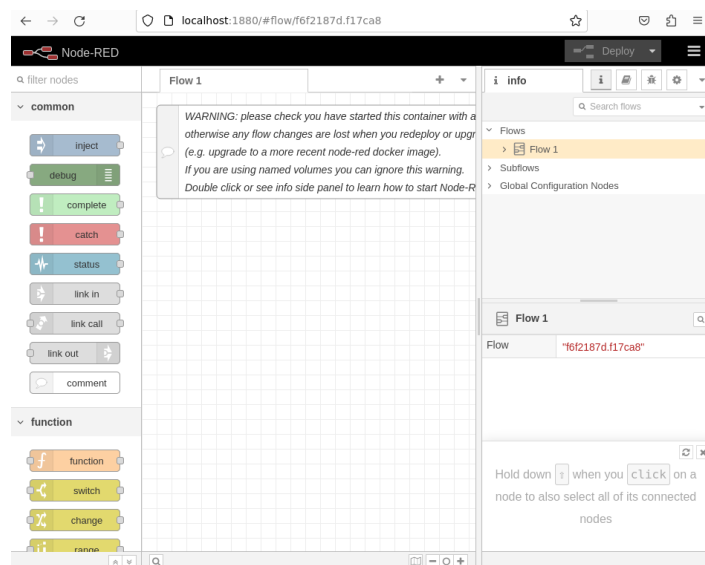
Commande pour l'installation du docker Node-Red :

```
docker run -d -p 1880:1880 --restart=always --name noderedSAE23 -v volNodeRed:/data nodered/node-red
```

- 1880:1880 : port choisi pour l'hôte (premier port) lié à celui du docker (deuxième port)
- --restart=always : le docker démarre quand on démarre la VM
- --name noderedSAE23 : nomme le docker créé "noderedSAE23"
- nodered/node-red : image du docker voulu

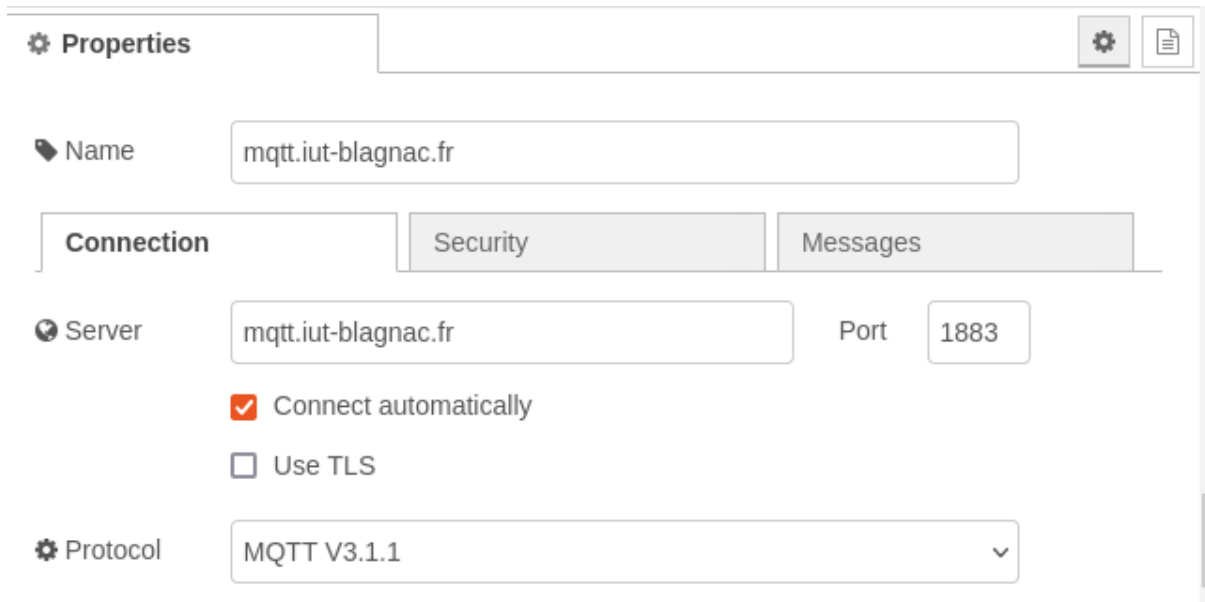
```
root@sae23julou:/home/ljulou# docker run -d -p 1880:1880 --restart=always --name noderedSAE23 -v volNodeRed:/data nodered/node-red
Unable to find image 'nodered/node-red:latest' locally
latest: Pulling from nodered/node-red
f56be85fc22e: Pull complete
4e21e25411da: Pull complete
9890dc0be345: Pull complete
61d1890b04b0: Pull complete
d70578450bb7: Pull complete
a8a78a3a522d: Pull complete
39e5079d72de: Pull complete
4f4fb700ef54: Pull complete
2d131dc3987: Pull complete
1ca3dfe8149b: Pull complete
e5aaa631bfed: Pull complete
9e246a4bf10c: Pull complete
851db8d1796d: Pull complete
f4c5c909de12: Pull complete
e0edbc440ebc: Pull complete
fb75fdab619c: Pull complete
32fbd9cc73be: Pull complete
Digest: sha256:c2f2c732ab2b6f7237c3c88f941864e847fa64a53cbc6a608d156f306c7ef48b
Status: Downloaded newer image for nodered/node-red:latest
88dd568d58949d33fff7833e4ed209f483edb380b4b669985a5b9d8bd16bf878
root@sae23julou:/home/ljulou#
```

Pour accéder à l'interface de Node-Red sur la VM, on se rend sur un navigateur à l'adresse <http://localhost:1880> (adress IP de la VM puis le port lié) :

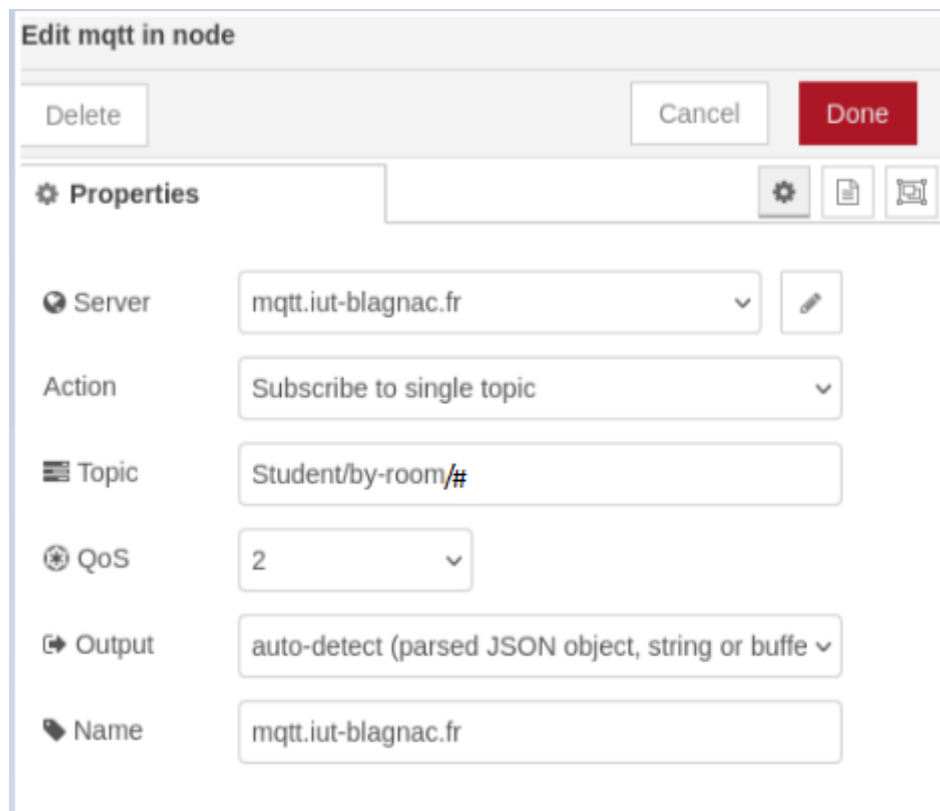


2. Configuration de MQTT dans Node-Red

- L'URL du serveur MQTT est mqtt.iut-blagnac.fr.
- Le port utilisé est 1883
- Le topic est 'Student/by-room/#'. Le dièse assure que l'on récupère les données de toutes les salles.



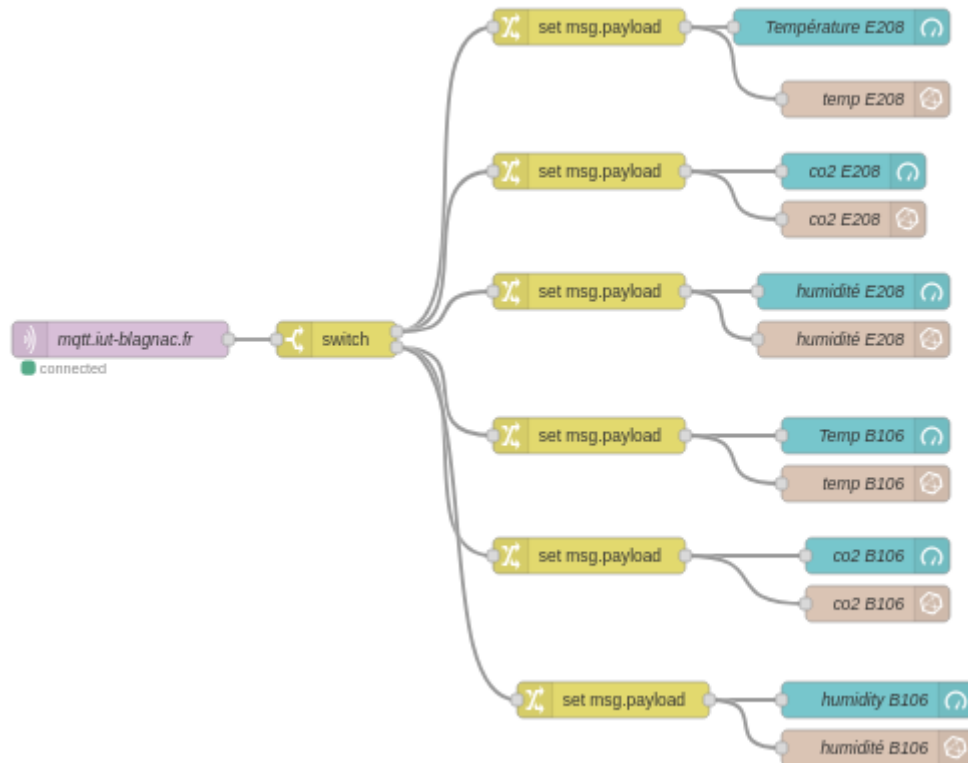
The screenshot shows the 'Properties' panel for an MQTT node in Node-Red. The panel has a title bar with a gear icon and a document icon. Below the title bar, there is a 'Name' field containing 'mqtt.iut-blagnac.fr'. Below this, there are three tabs: 'Connection', 'Security', and 'Messages'. The 'Connection' tab is active. Under the 'Connection' tab, there is a 'Server' field containing 'mqtt.iut-blagnac.fr' and a 'Port' field containing '1883'. Below these fields, there are two checkboxes: 'Connect automatically' (checked) and 'Use TLS' (unchecked). At the bottom, there is a 'Protocol' dropdown menu set to 'MQTT V3.1.1'.



The screenshot shows the 'Edit mqtt in node' dialog box in Node-Red. The dialog has a title bar with the text 'Edit mqtt in node'. Below the title bar, there are three buttons: 'Delete', 'Cancel', and 'Done'. Below the buttons, there is a 'Properties' panel. The 'Properties' panel has a title bar with a gear icon and a document icon. Below the title bar, there are several fields: 'Server' (mqtt.iut-blagnac.fr), 'Action' (Subscribe to single topic), 'Topic' (Student/by-room/#), 'QoS' (2), 'Output' (auto-detect (parsed JSON object, string or buffer)), and 'Name' (mqtt.iut-blagnac.fr).

3. Configuration de Node-Red-Dashboard

Voici le dashboard :



Configuration du switch de Node-Red afin de récupérer les données des salles concernées :

Edit switch node

Delete

Cancel

Done

⚙ Properties

⚙

📄

🖨

🔍 Name

Name

⋮ Property

▼ msg. topic

≡

== ▼

▼ ^a_z Student/by-room/E208/data

→ 1

✕

≡

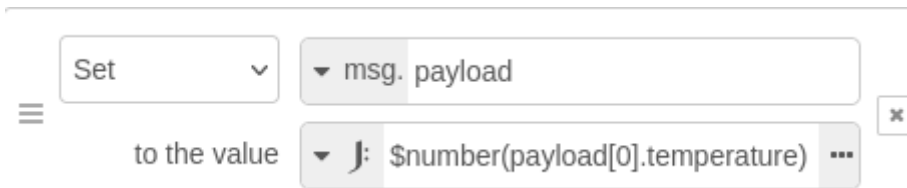
== ▼

▼ ^a_z Student/by-room/B106/data

→ 2

✕

On récupère les valeurs voulues dans les modules “set” :

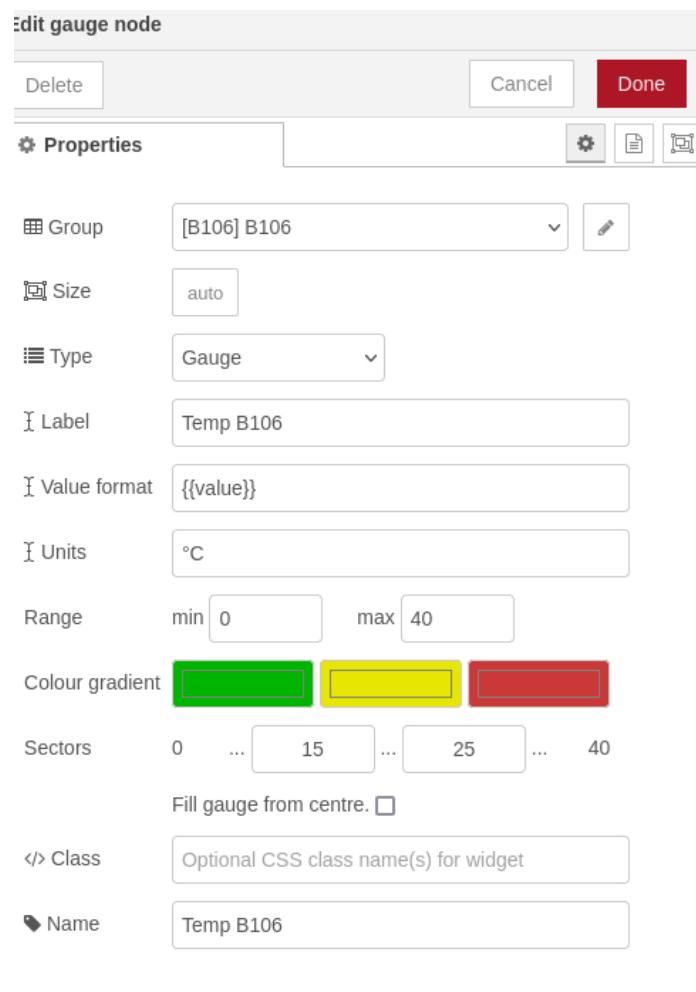


L'expression “\$number(payload[0].temperature)” récupère la température dans une salle pour chacune des salles E208 et B106. Nous voulons aussi le CO2 en ppm et le taux d'humidité. Pour cela il faut changer dans l'expression précédente le “.température” en “.co2” pour le co2 et en “.humidity” pour l'humidité.

Pour configurer les jauges afin d'afficher graphiquement les données ainsi récupérées, il faut se rendre dans le menu > Manage Palette > Install > node-red-dashboard > install.

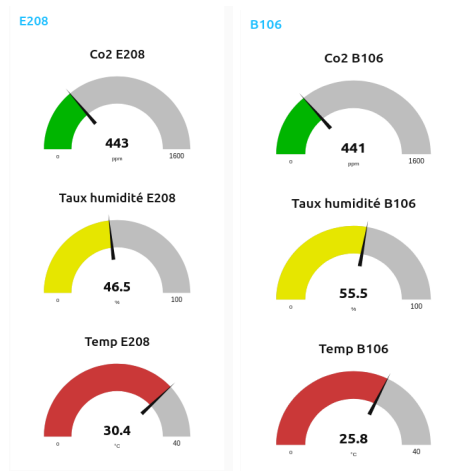
Ceci installe le module pour avoir le dashboard accessible sur <http://localhost:1880/ui>

Il faut par la suite définir les bornes des jauges et leurs couleurs :



On répète ce processus pour toutes les valeurs voulues.

Le rendu est visible à l'adresse <http://localhost:1880/ui> :



4. Mise en place d'InfluxDB

```
docker run -d -p 8086:8086 --restart=always --name influxdbSAE23 -v volinfluxdb:/var/lib/influxdb influxdb:1.8
```

La commande utilise les mêmes options que celle du docker Node-Red.

```
root@sae23julou:/home/ljulou# docker run -d -p 8086:8086 --restart=always --name influxdbSAE23 -v volinfluxdb:/var/lib/influxdb influxdb:1.8
1bfc93a71527c373e6c516dd4e2ffcc913e50d6721be6c6c554455d0064430b
```

Il faut par la suite se connecter à l'interface d'InfluxDB afin de configurer la base de données avec la commande `docker exec -it influxdbSAE23 influx` :

```
root@sae23julou:/home/ljulou# docker exec -it influxdbSAE23 influx
Connected to http://localhost:8086 version 1.8.10
InfluxDB shell version: 1.8.10
```

Nous créons ensuite la base de données SAE23 avec la commande `create database SAE23` :

```
> show databases
name: databases
name
----
_internal
SAE23
> use SAE23
Using database SAE23
>
```

L'utilisateur 'root' au mot de passe 'passroot' est créé avec la commande `create user` :

```
> create user root with password 'passroot' with all privileges
> show users
user admin
-----
root true
```


5. Connexion à InfluxDB depuis Node-Red


Il est nécessaire d'installer le module qui permet de dialoguer avec InfluxDB au préalable dans Menu > Manage Palette > Install > node-red-contrib-influxdb > install.

On configure ensuite le noeud 'influxdb out' avec les paramètres de la base de données InfluxDB :


Name	SAE23		
Version	1.x		
Host	192.168.108.207	Port	8086
Database	SAE23		
Username	root		
Password	••••••••		


Sélection de la base de données et des mesures à insérer :


 **Properties**


 Name

temp E208

 Server

[v1.x] SAE23 



 Measurement

temperature,building=E,room=E208

☐ Advanced Query Options

Après avoir fait tourner Node-Red pour récupérer un jeu de données, on vérifie la présence de ces dernières dans InfluxDB avec la commande `select` :

```
name: temperature
time                building room value
----
1685719474757724748 E          E208  24.4
1685720074459917336 E          E208  24.5
1685720224317578535 B          B106  24.5
1685720674222006191 E          E208  24.5
1685720825036921884 B          B106  24.4
1685721274910299969 E          E208  24.5
> select * from co2
name: co2
time                building room value
----
1685719474758900901 E          E208  675
1685720074459802238 E          E208  726
1685720224318593899 B          B106  432
1685720674222846541 E          E208  724
1685720825037938444 B          B106  431
1685721274909858662 E          E208  717
```

```
> select * from humidity
name: humidity
time                building room value
-----
1685719474757938135 E          E208 49
1685720074459877875 E          E208 48.5
1685720224317995069 B          B106 49
1685720674222897008 E          E208 48
1685720825037700329 B          B106 49
1685721274909932678 E          E208 48
```

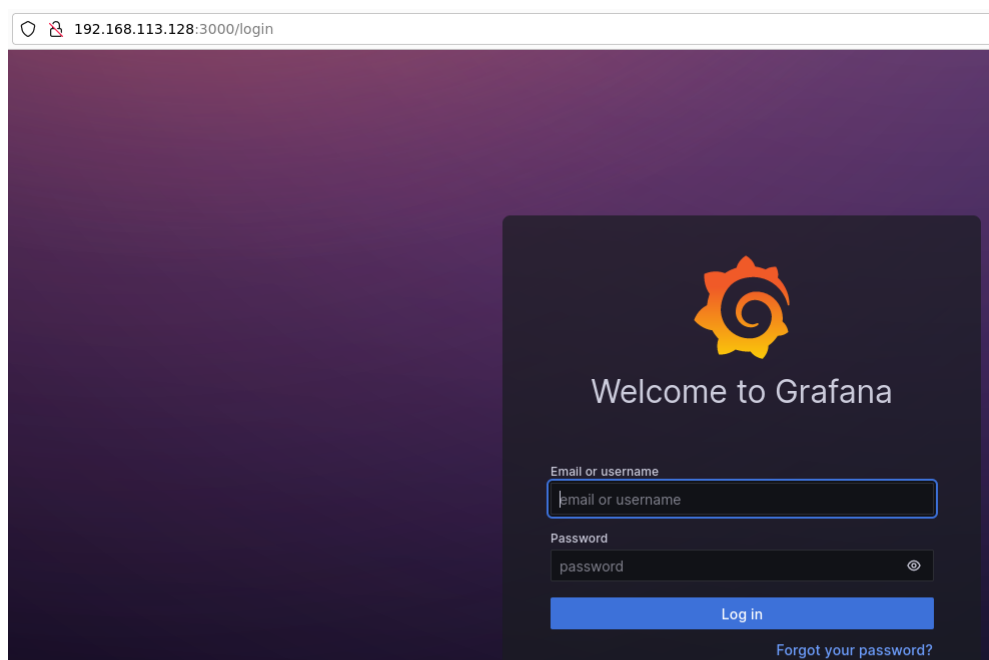
6. Configuration de Grafana

Création du docker grafana :

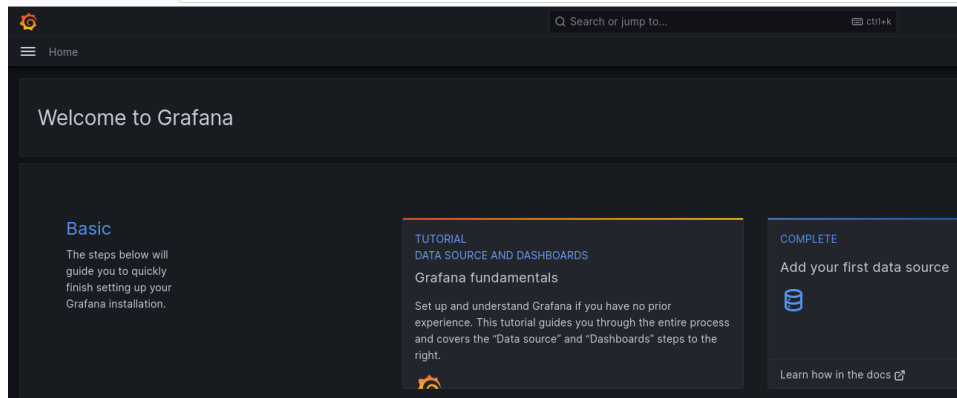
```
docker run -d -p 3000:3000 --restart=always --name grafanaSAE23 -v
volGrafana:/var/lib/grafana grafana/grafana
```

```
root@SAEljulou:/home/loicj# docker run -d -p 3000:3000 --restart=always --name gr
afanaSAE23 -v volGrafana:/var/lib/grafana grafana/grafana
7de5bad580f3beedc7cbcd26a5490c3e0500529c68ae768d4e1f6f21414cbf9
root@SAEljulou:/home/loicj#
```

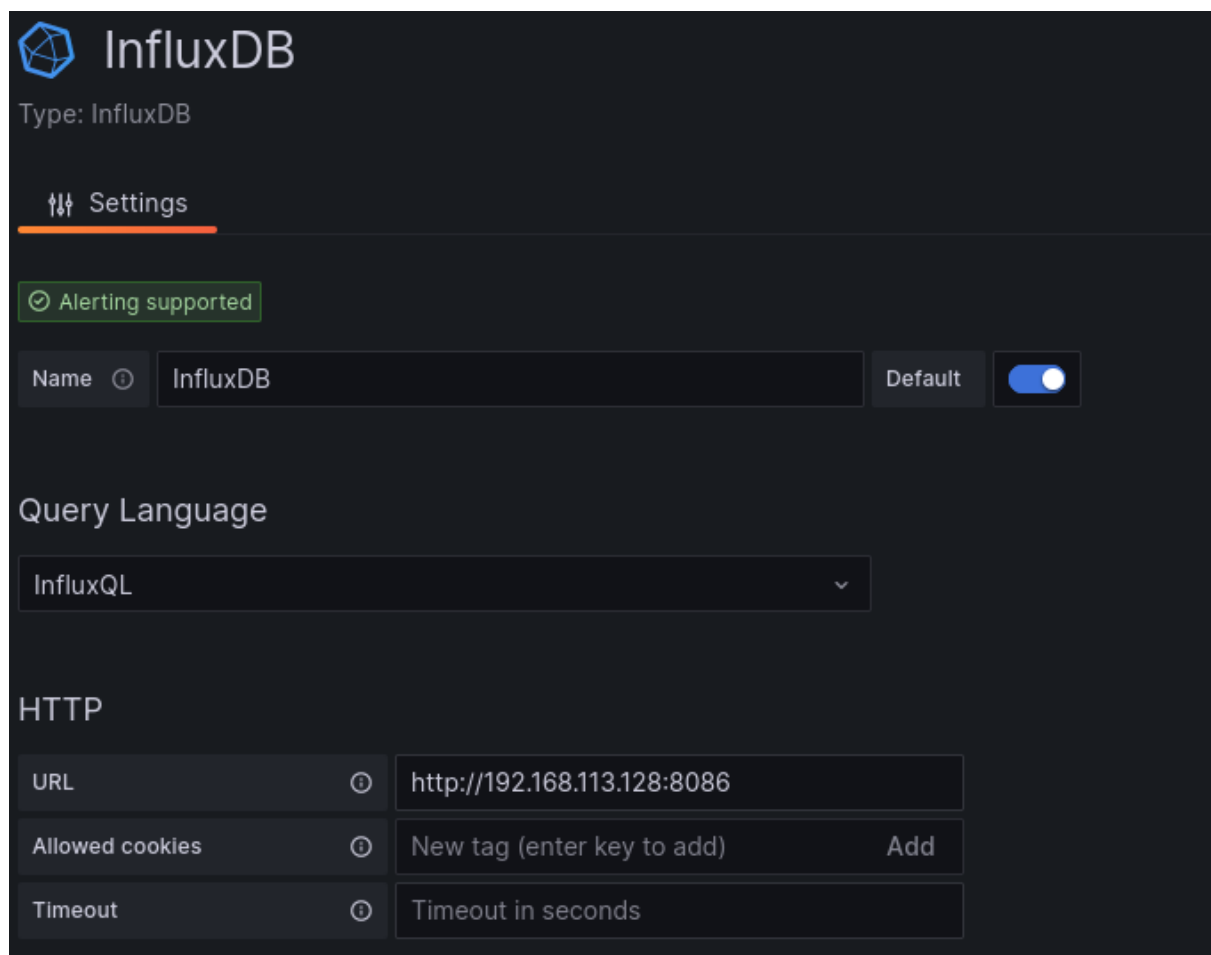
Grafana est accessible en HTTP au port 3000 :



On se connecte avec les identifiants de l'utilisateur admin (mot de passe passroot)



Après avoir cliqué sur “Add your first data source”, il faut rentrer les paramètres requis pour que Grafana puisse récupérer les données de la base de données InfluxDB :

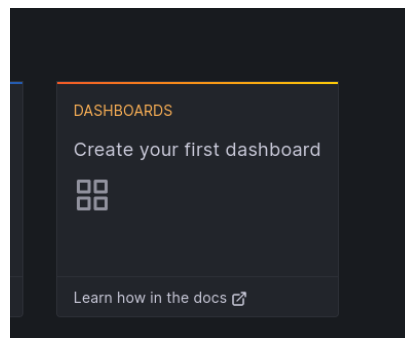


On rentre par la suite le nom de la base de données ciblée ainsi qu'un utilisateur d'InfluxDB que l'on a créé précédemment :

Database	SAE23	
User	root	
Password	configured	Reset
HTTP Method	Choose ▾	
Min time interval	10s	
Max series	1000	

[Back](#) [Explore](#) [Delete](#) [Save & test](#)

Pour créer notre dashboard, on clique sur “Create your first dashboard” :



Dans un premier temps, on formalise la requête qui sera réalisée périodiquement sur la base de données contenant les relevés des capteurs.

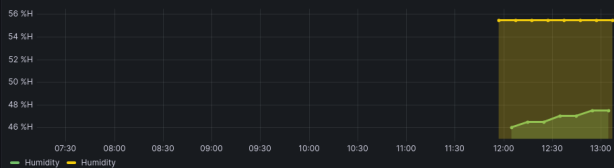
Sur la première ligne de la requête, on paramètre tous les menus pour réaliser une requête de la forme `FROM default temperature WHERE building = RT AND room = E208`. Bien sûr, il faut répéter ce processus pour les autres valeurs dans la base de données SAE23.

Enfin, on peut visualiser graphiquement les résultats :

Co2 en E208(vert) et B106(jaune)



Taux d'humidité en E208(vert) et B106(jaune)



Temperature en E208(vert) et B106(jaune)

