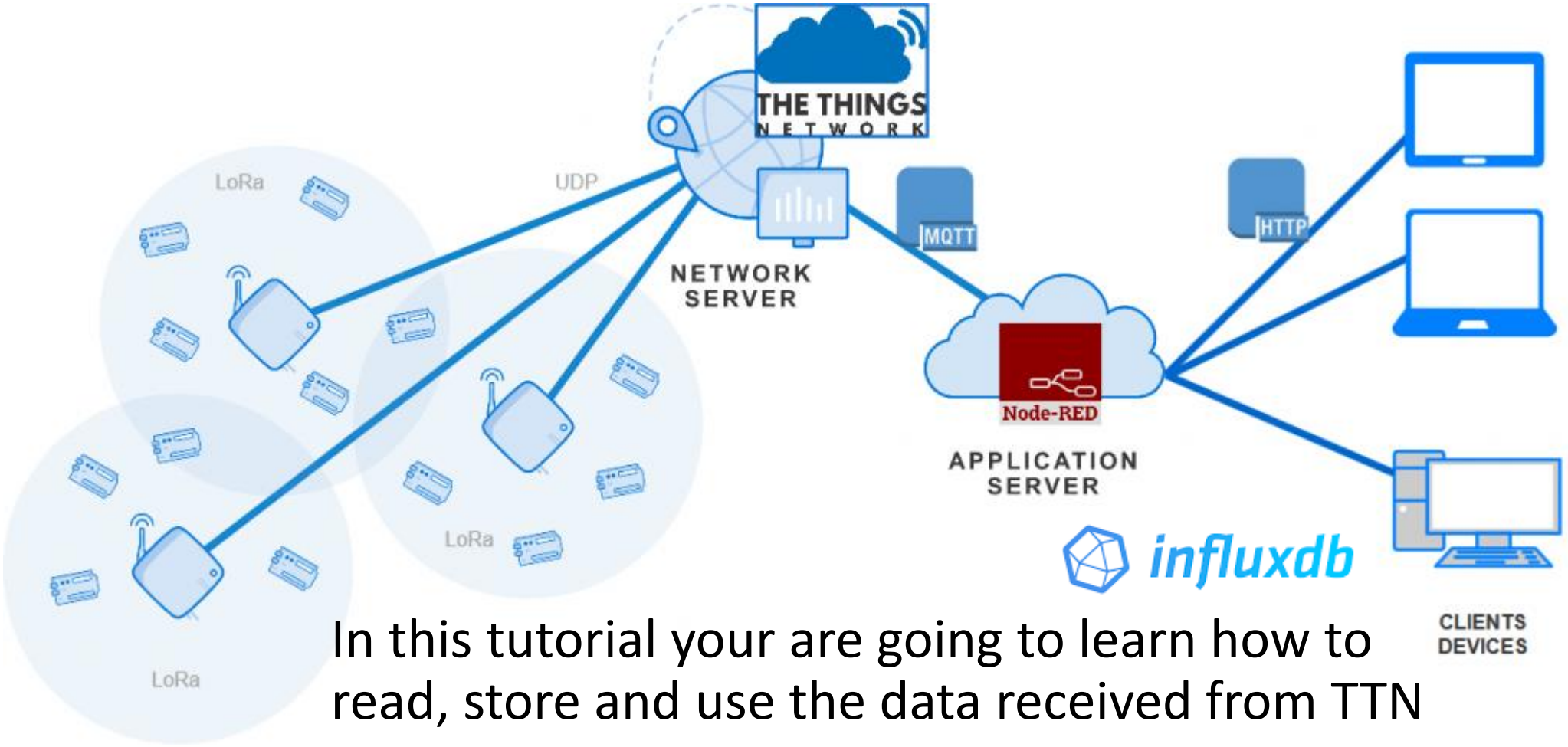
IoT LoRa application service Tutorial

F. Ferrero

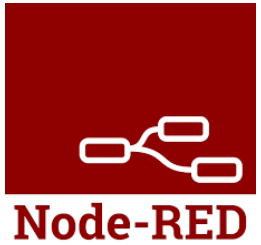
UNIVERSITÉ CÔTE D'AZUR

Node Red – InFluxDB - GRAFANA



In this tutorial your are going to learn how to read, store and use the data received from TTN

Node Red



- Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.
- It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.
- Built on Node.js
  - The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.

InfluxDB

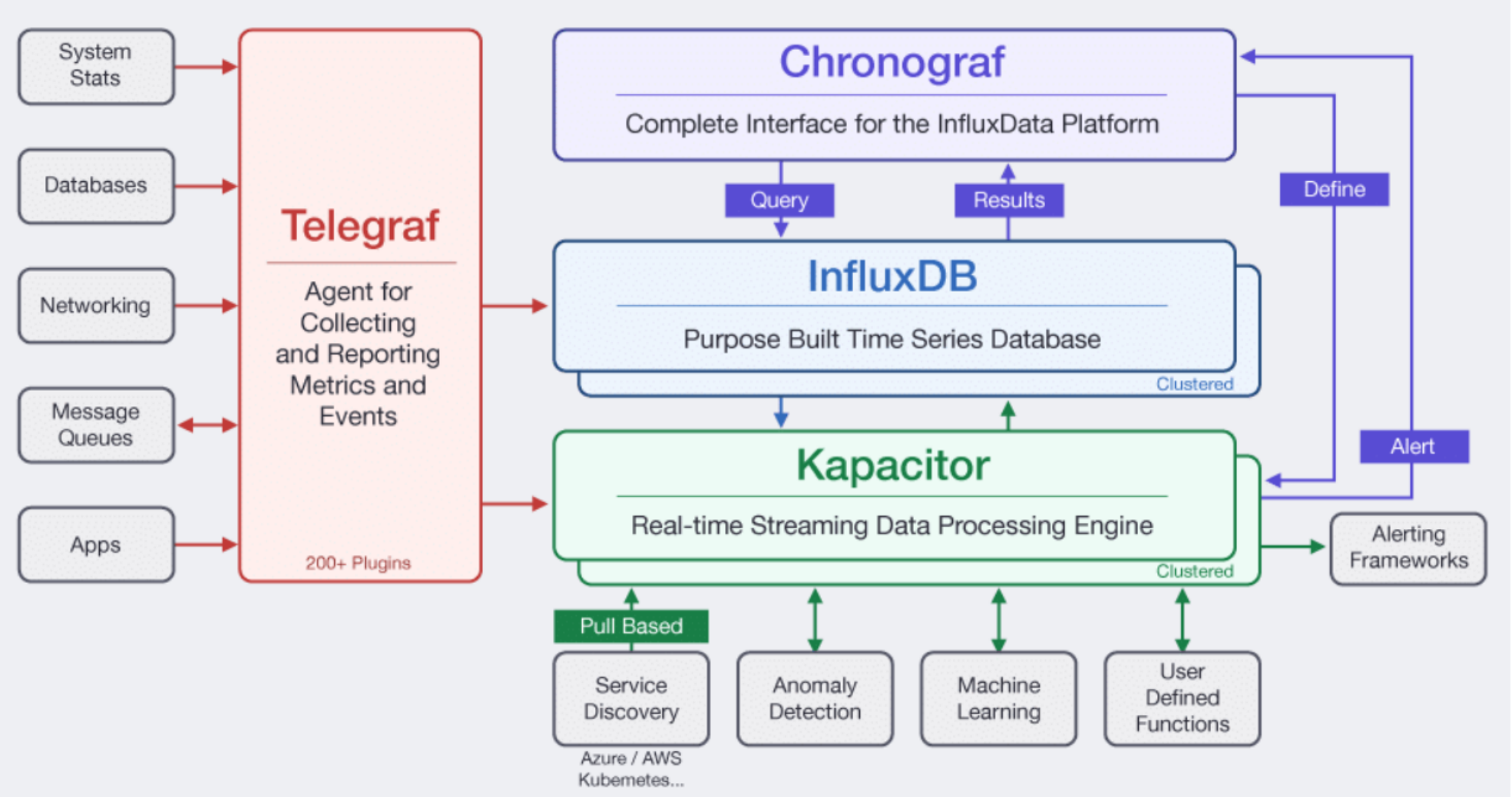
InfluxDB

- InfluxDB is an open source distributed time series database developed by InfluxData. The main advantage of InfluxDB is its capacity to aggregate values in time buckets on-the-fly without any manual intervention.
- InfluxDB can be accessed by software like Grafana, which is a powerful front-end tool providing visualisation features for time series data. Each point consists of varied key-value pairs called fieldset and timestamp. Points are indexed by their time and tagset. InfluxDB stores data via HTTP, TCP and UDP.

Features

- Purely written in the Go programming language and facilitates compilation into a single binary with no external dependencies.
- High performance customised data store written especially for time series data. The TSM engine of InfluxDB allows efficient and high speed data storage and compression.
- In-built Web front-end tool for database and user administration.
- Competent in merging multiple series together.
- **Official website:** *https://www.influxdata.com/*

InfluxDB



Additionaly to the database, influx data provide interesting applications

Grafana

- Grafana allows you to query, visualize, alert on and understand your metrics no matter where they are stored. Create, explore, and share dashboards with your team and foster a data driven culture.

- Grafana includes a built in Graphite query parser that takes writing graphite metric expressions to a whole new level. Expressions are easier to read and faster to edit than ever.
- Click on any metric segment to change it
- Quickly add functions (search, typeahead)
- Click on a function parameter to change it
- Move function order to the left or right
- Direct link to Graphite function documentation
- Rich templating support

Node-Red

- Please install Node Red and TTn lib

https://www.thethingsnetwork.org/docs/applications/nodered/quick-start.html

If git is not installed on your PC : https://git-scm.com/downloads

You will also need to install :

node-red-contrib-influxdb

Influx DB Data base

- Please install InfluxDB v1.6.4

https://portal.influxdata.com/downloads

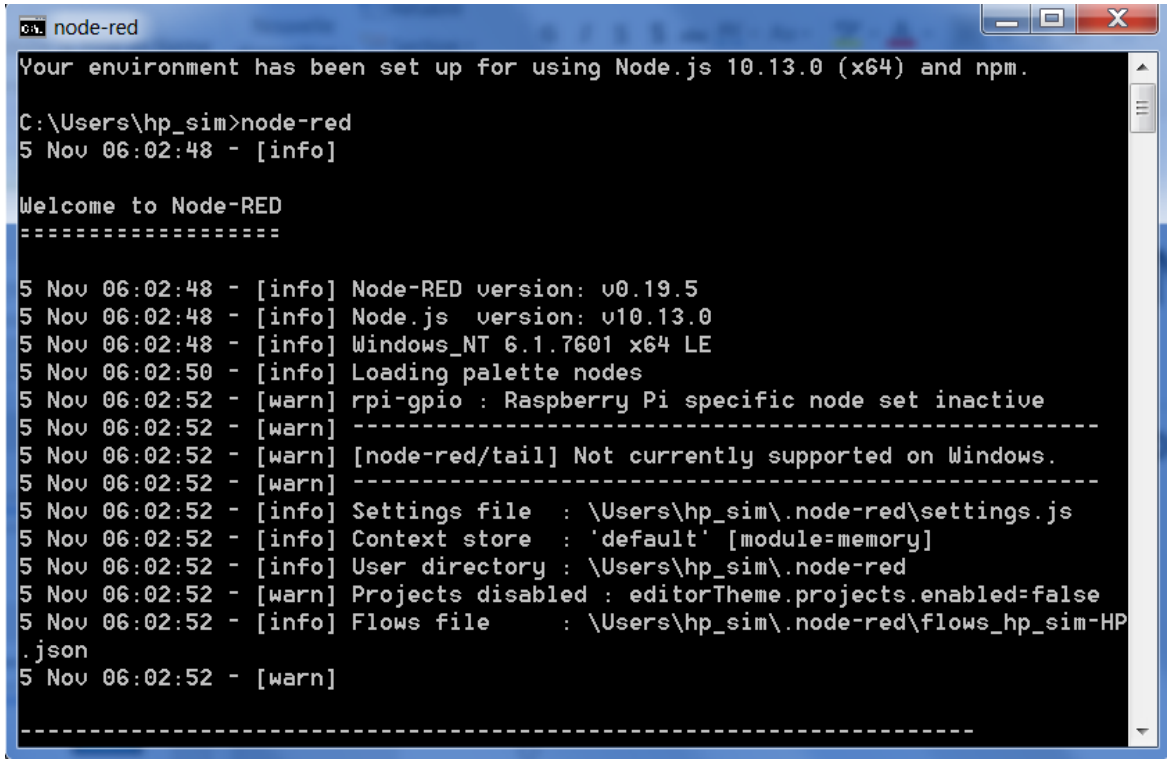
Grafana

- Please install Grafana

https://grafana.com/get

Connecting to TTN

- Start NODE.js command prompt
- Run : node-red
- Open your web browser and go to http://127.0.0.1:1880

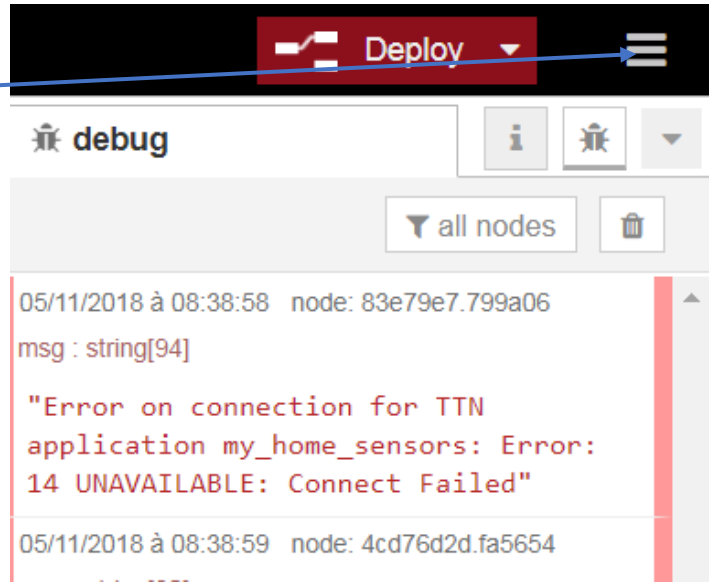


- On the editor, click here

And go to palette editor

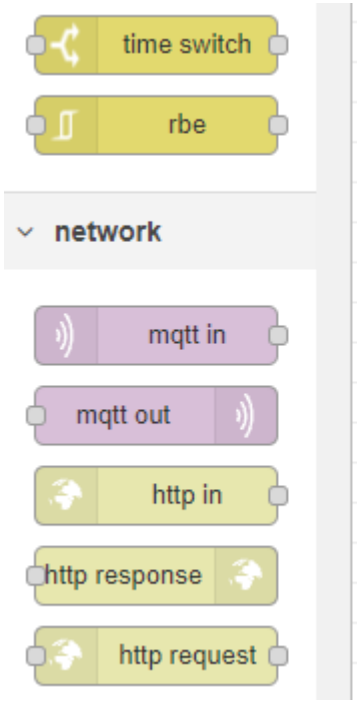
Install :

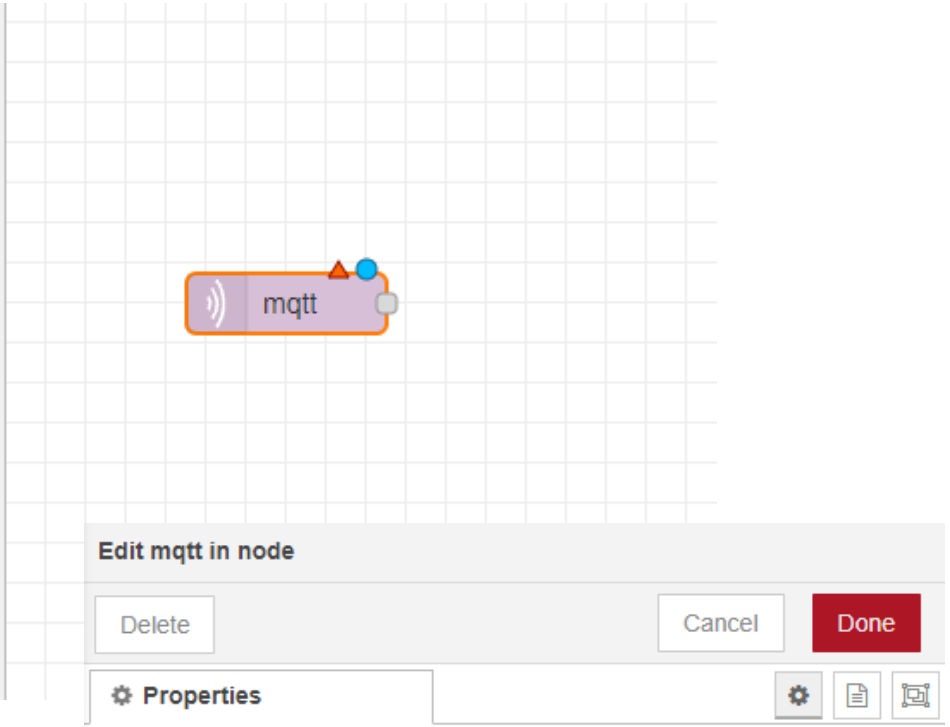
- node-red-contrib-influxdb

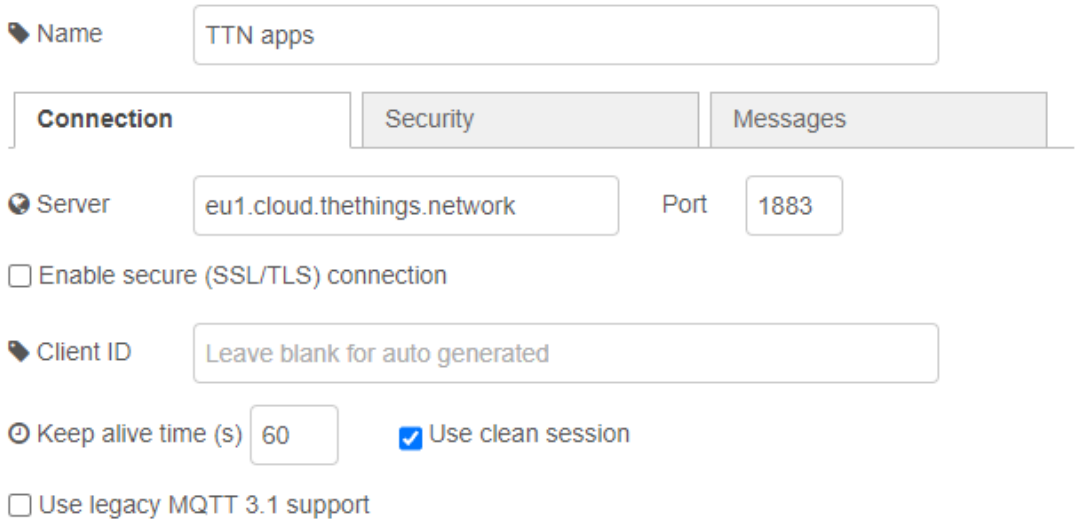


Connecting to TTN

- You have the graphical Node-red editor
- Add mqtt in node
- Edit mqtt
- Choose « Add new mqtt-broker … » in App and click on edit







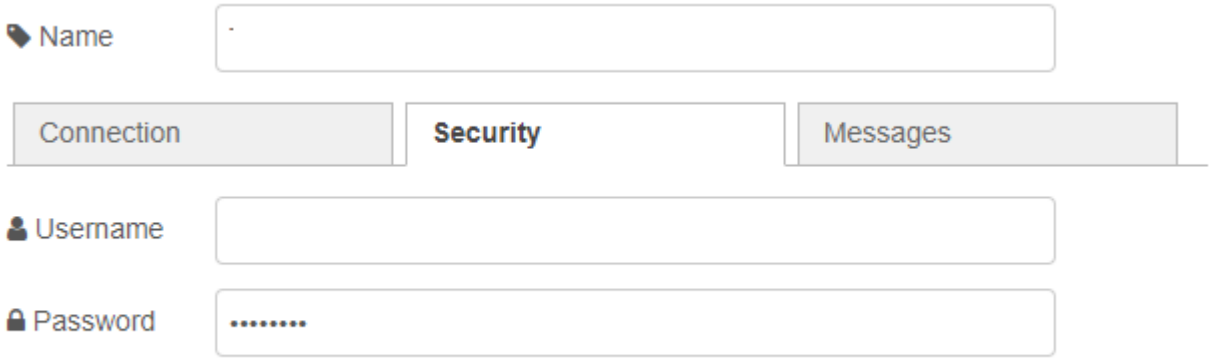
Connecting to TTN

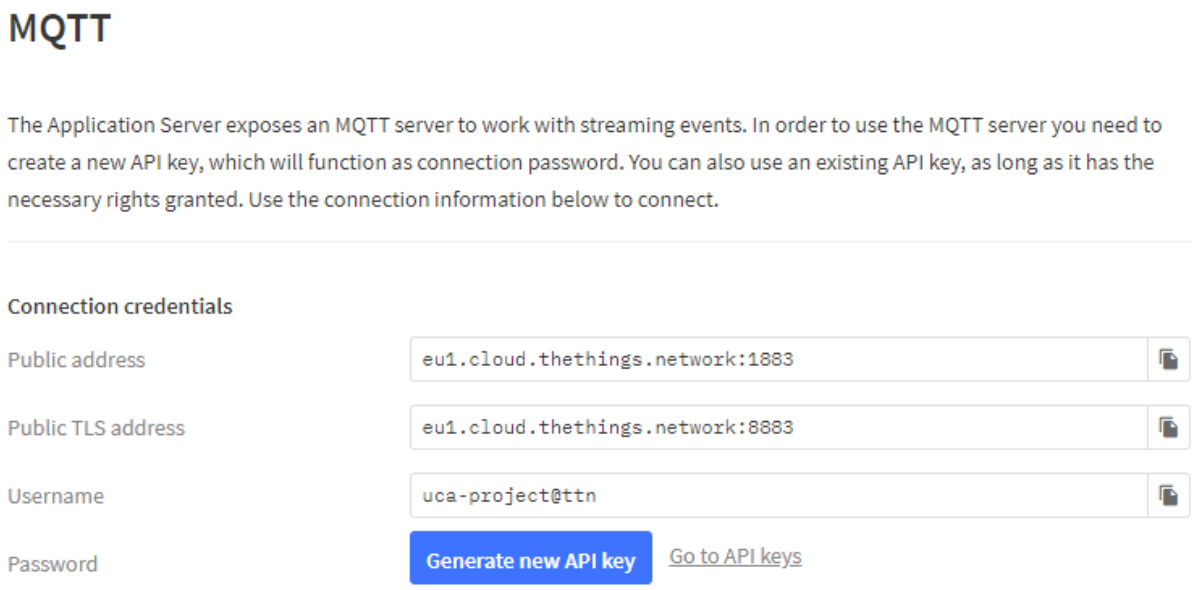
- Update security and topic:

discovery.thethingsnetwork.org:1900

- Go to you application in TTN
- Copy past the User name and keys

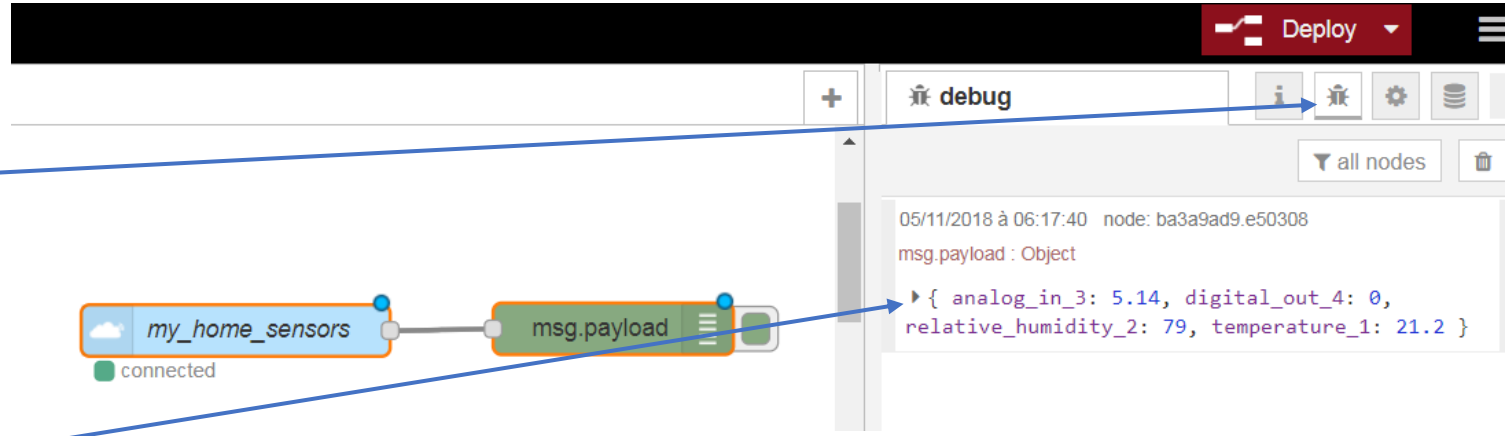
v3/uca-project@ttn/devices/**device_name**/up

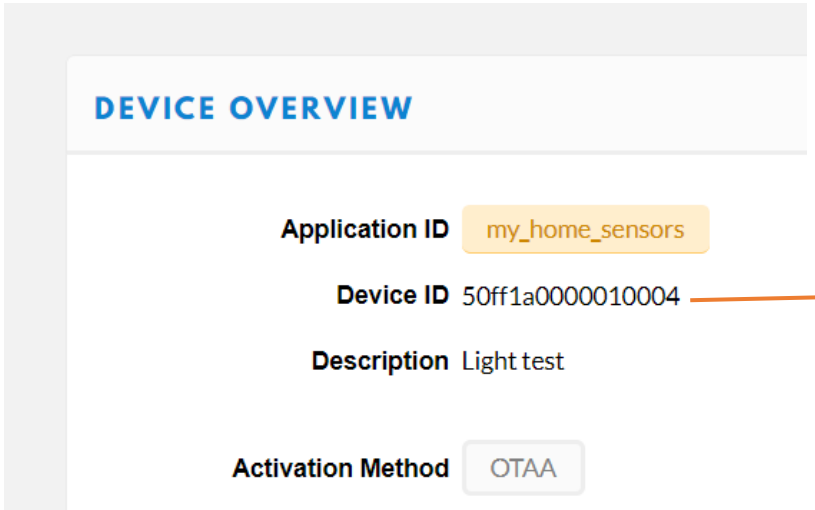


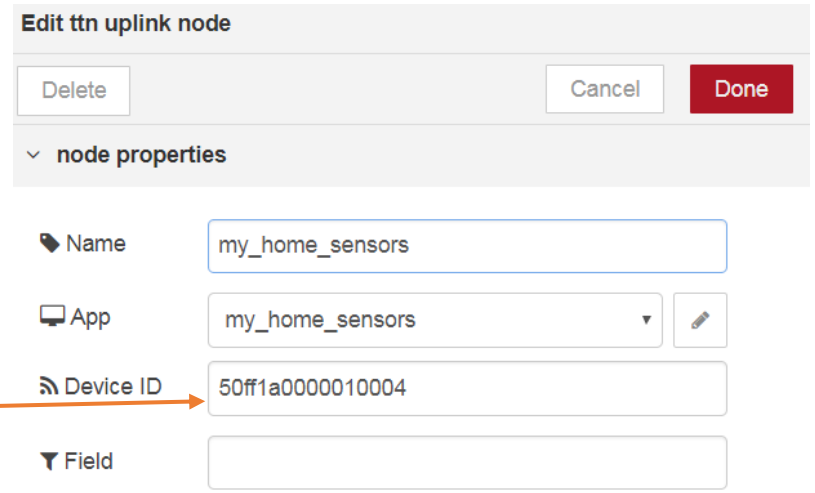


Connecting to TTN

- Click on Deploy
- You uplink TTN should be connected
- Click on debug window
- You will receive the packet of the application
- If you want to filter only your device, add your device ID
- Click here :

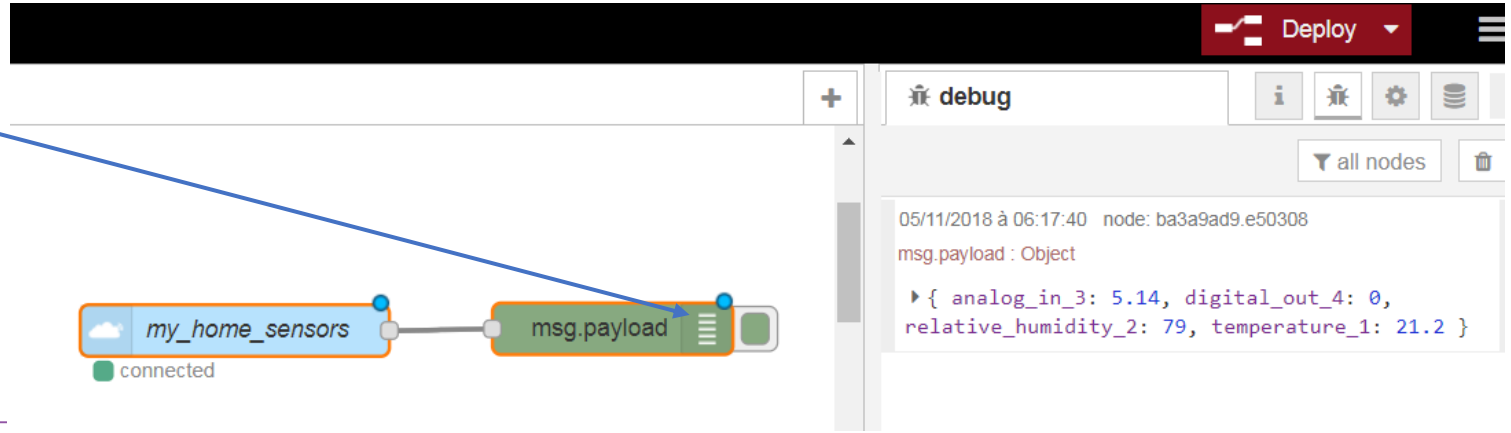


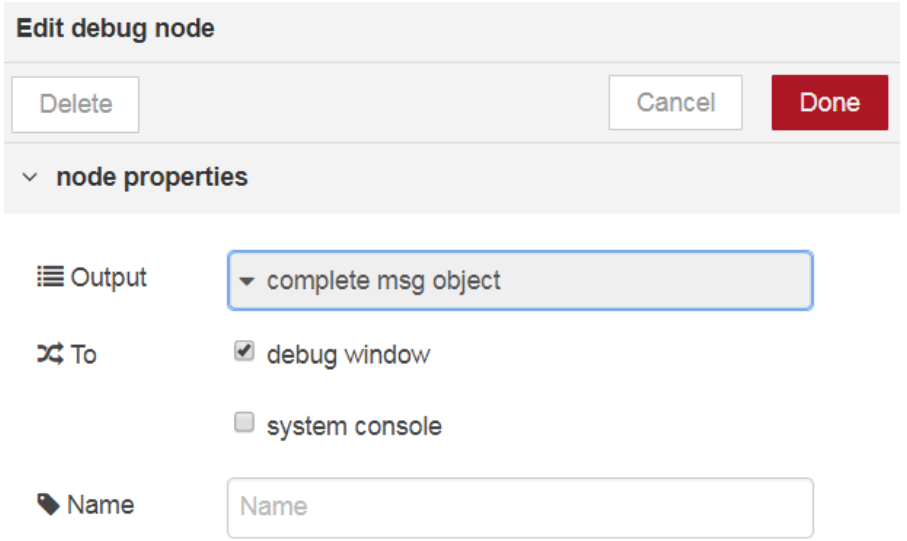




Connecting to TTN

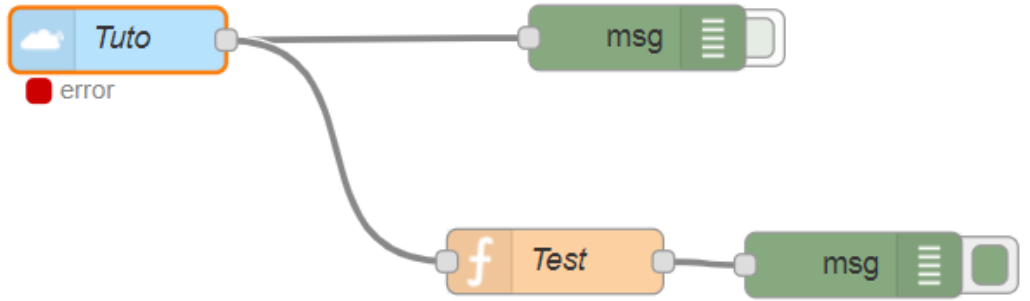
- Click here :
- Choose « complete msg object »
- And Deploy
- You have now more information of your uplink





Connecting to TTN

- If you want to extract only 1 data,
- as an exemple the RSSI (received signal Strength indicator
- Use a function to extract the wanted data




```
return {
 // Some fields from the metadata  freq:
msg.metadata.frequency,
cr: msg.metadata.cr,
dr: msg.metadata.dr,

// Combine RSSI and SNR of all gateways into two arrays:
rssi: gateways.map(gw => gw.rssi),
snr: gateways.map(gw => gw.snr),

};
```

InfluxDB

- Run « influxd.exe » , it will start the database
- Run « influx.exe », it will open a shell
- Write : « CREATE DATABASE mySensor »
- Then write : « SHOW DATABASE »

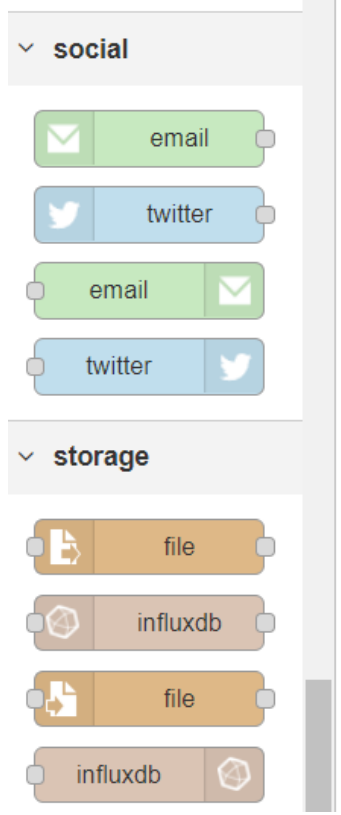
Your database is created

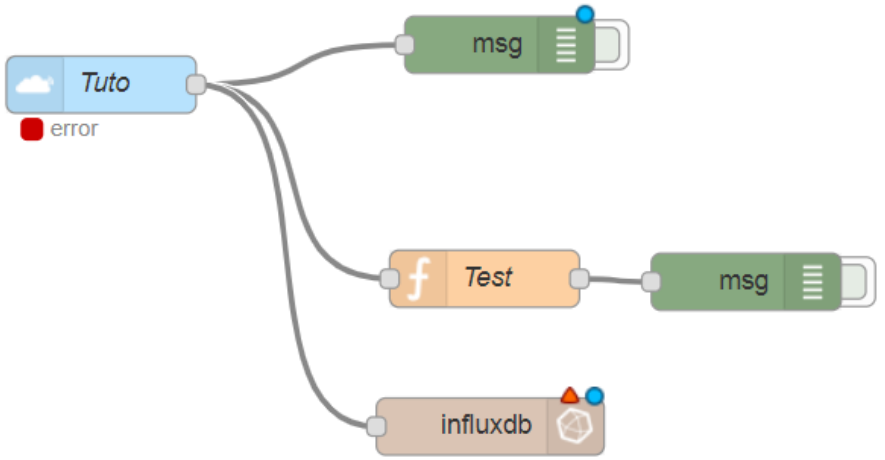
https://docs.influxdata.com/influxdb/v1.6/introduction/getting-started/

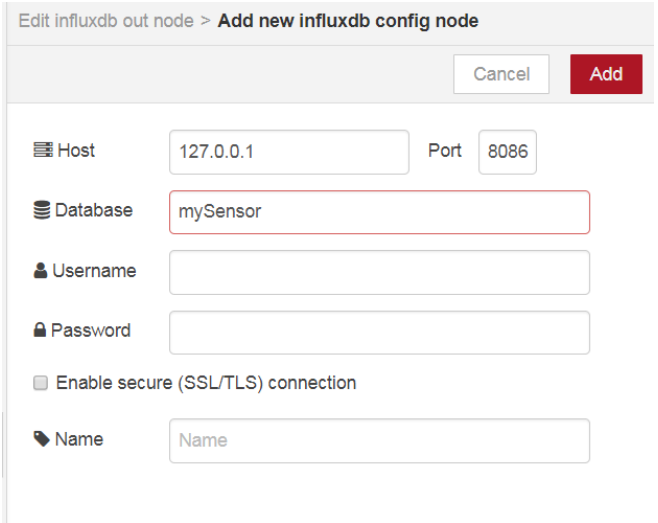
InfluxDB – Node Red

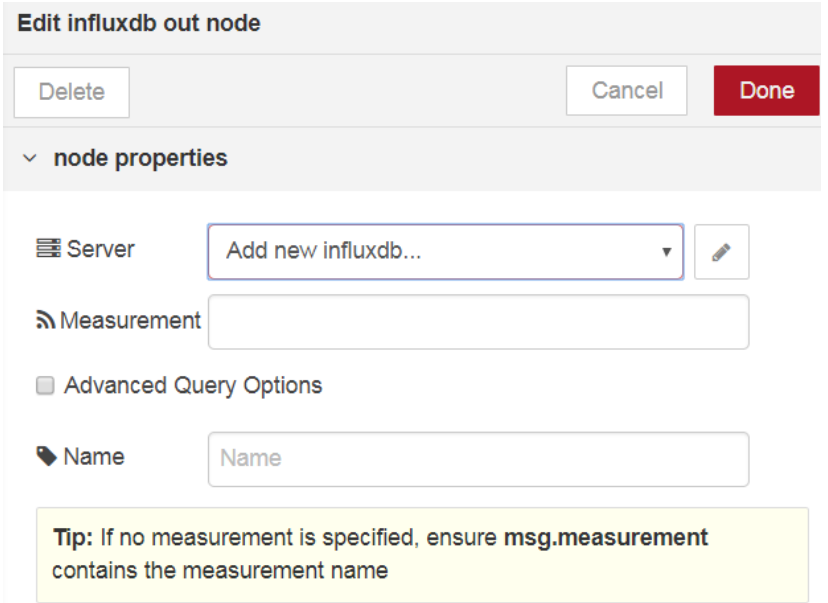
- How to store data in your database ?
- Add an influxdb storage and connect it to your uplink
- Define a server, just add the Database name : mySensor
- Add
- In measurement field, add a name for your device : device1

- Go to InfluxDB shell
- Run : SHOW SERIES ON mySensor





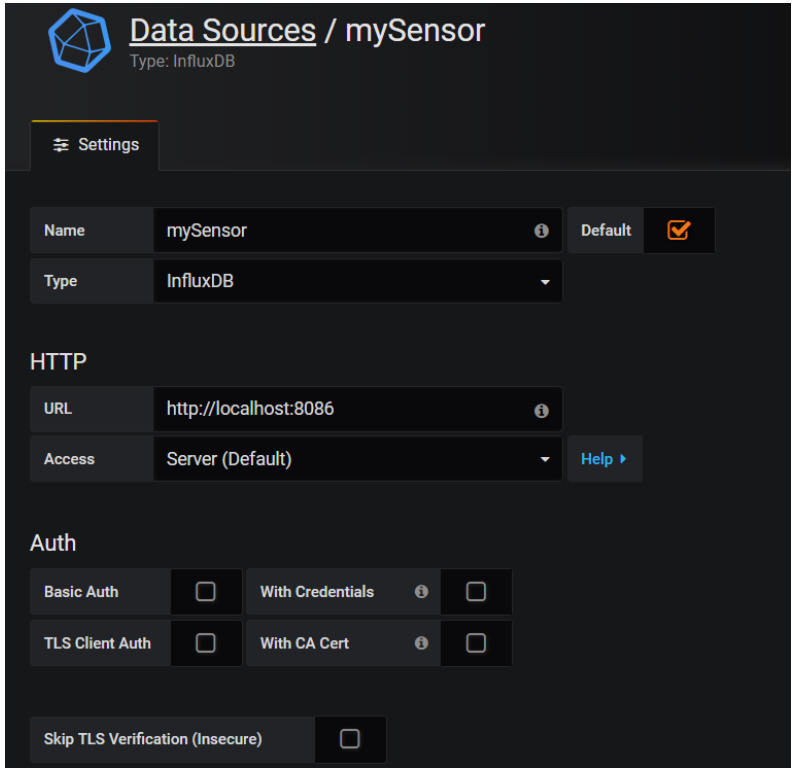


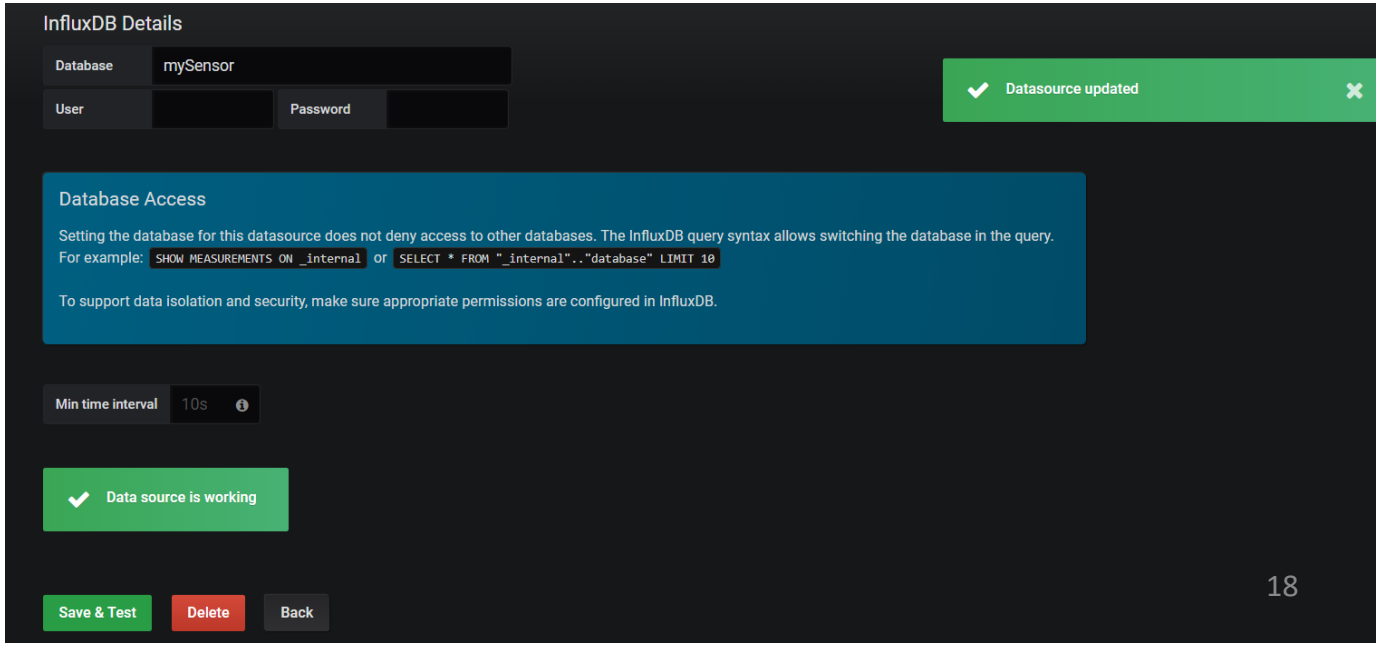


https://docs.influxdata.com/influxdb/v1.6/introduction/getting-started/

Grafana

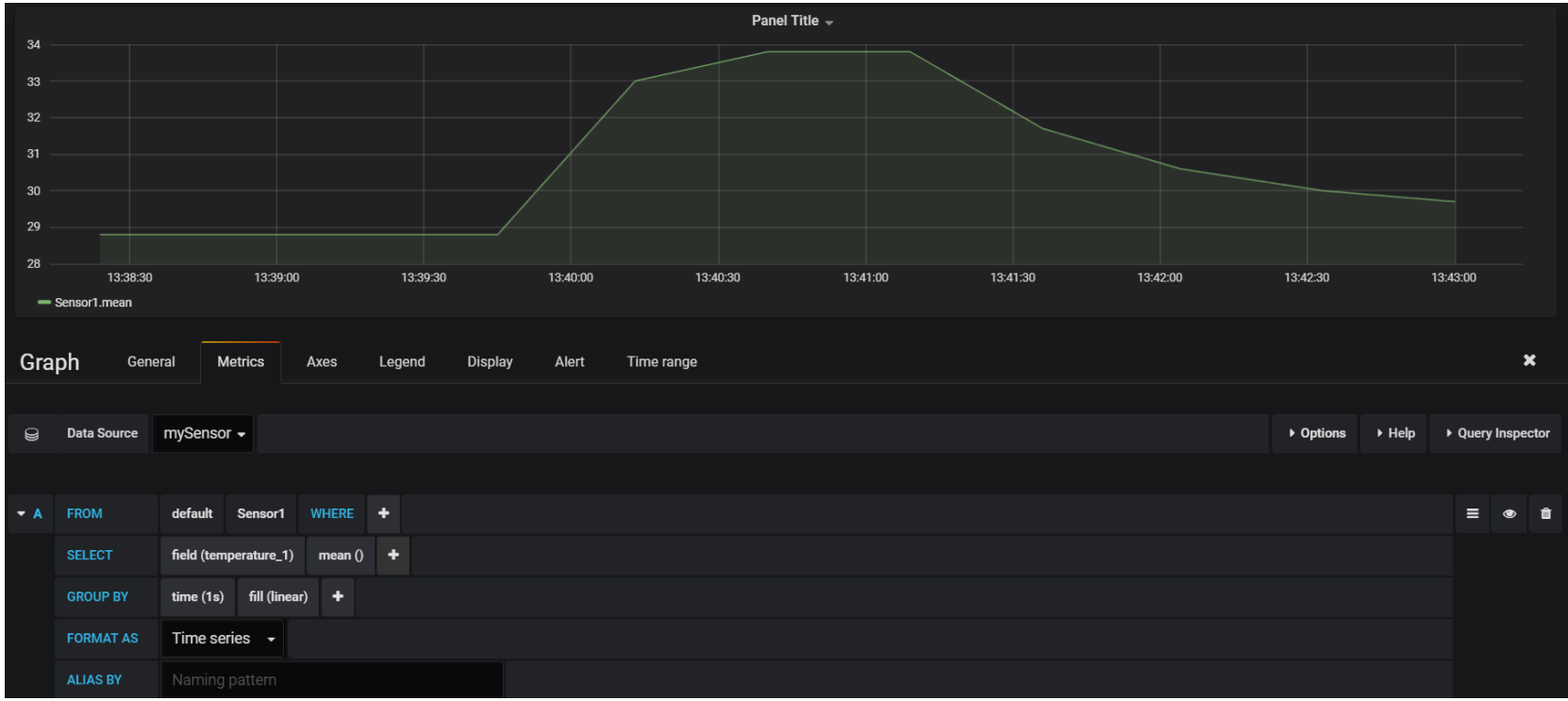
- Go to your unzip Grafana directory/bin
- Start grafana-server.exe
- Go to : http://127.0.0.1:3000
- User name and password is : admin
- Provide a new password
- Click « Add data source »
- Add a name
- Choose InfluxDB type
- Define Database name « mySensor »
- Click on Save and Test





Grafana

- Create a new dashboard
- Click on Graph
- Panel Title / Edit
- Select your data source and measurement, field temperature, time 1s, fill linear
- Change to the last 5mn
- Put your finger on the sensor
- Look at your curve



- To speed your measurement :

In Arduino code, change Tx interval to be 20s for SF7

```
case DR_SF7: debugPrintLn(F("Datarate: SF7"));
    TX_INTERVAL = 20;
```

Good luck for you projects !