How to connect your HummBox to Node-Red (using MQTT). Example: generating an email alert if the temperature exceed a certain value in a 50 km wide zone

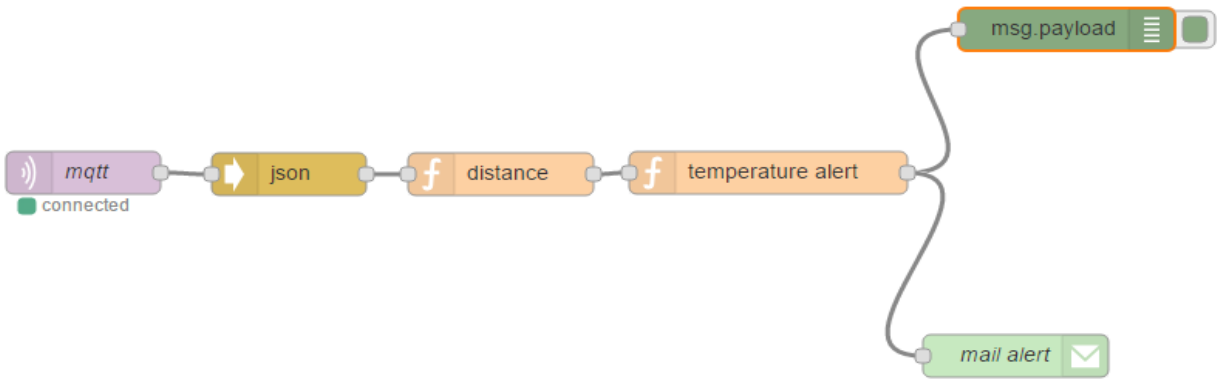
1-Node Red installation:

Go to http://nodered.org/docs/getting-started/installation and choose the node Red installation to your operating system.

1-Connect to MQTT:

Once installed lunch Node Red.

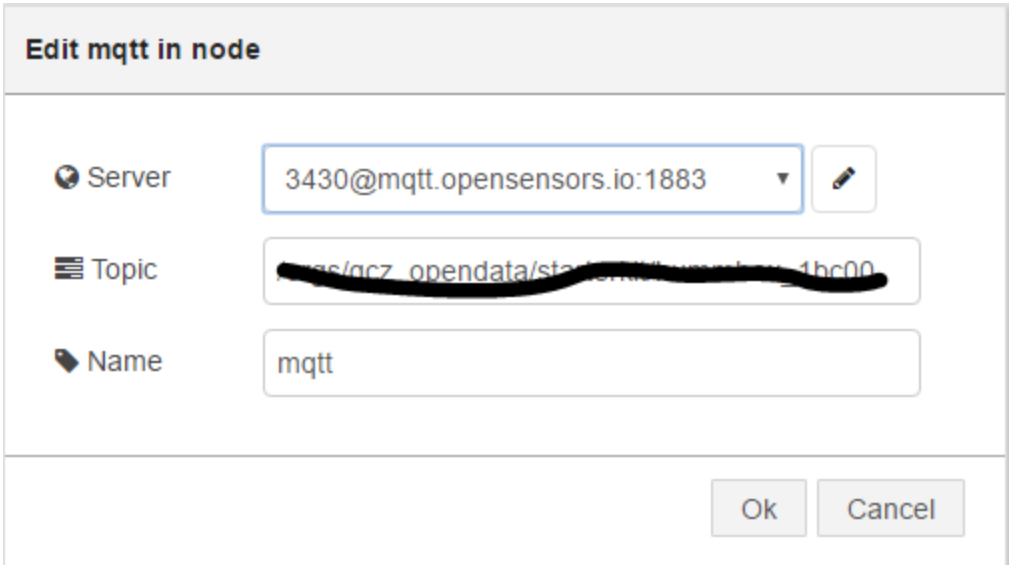
We propose you a tutorial where you can visualize your data and send an email alert if the temperature exceed 50°C in a radius of 50 km from your position.



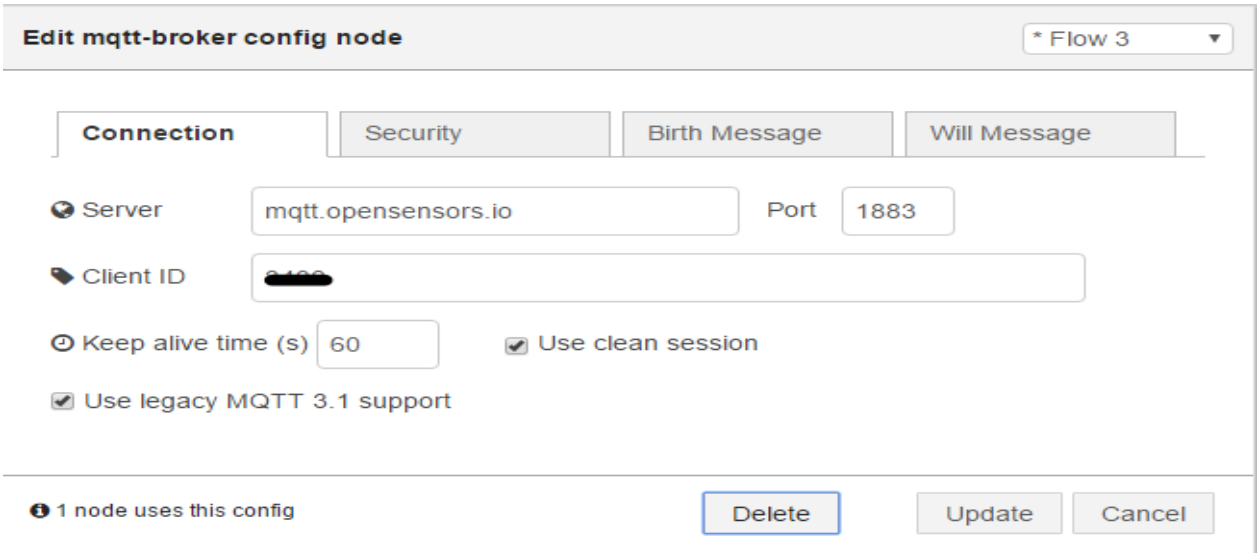
We will start by configuring node by node.

Mqtt:

First step:
- add an *mqtt input* node onto the canvas.
- Double click on the node to edit the configuration.
- Enter your topic in the Topic field.

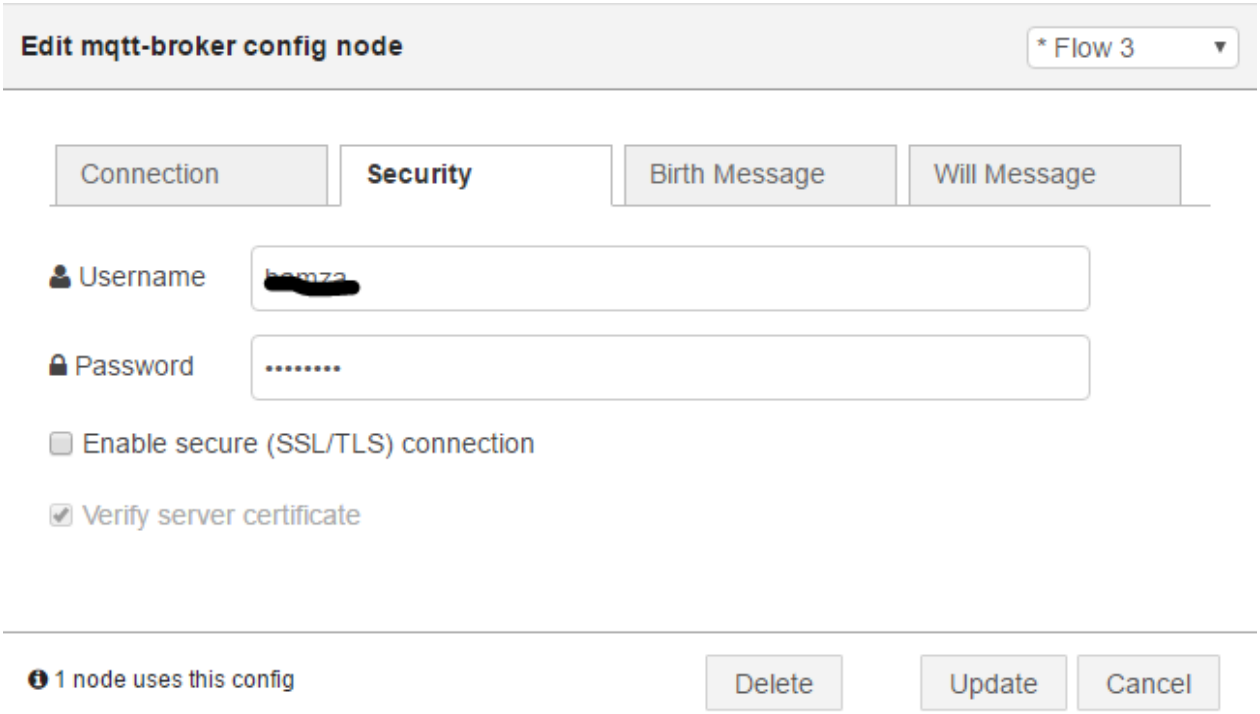


Then click on the pen at the left side of server:



Here you find several windows, we start with the first one *connection,* in this window you have to insert the server name: **mqtt.opensenosrs.io** and your client ID.

Then go to Security window:
You have to enter your opensensors.io Username and the password of your device.



Then click update that is all for the mqtt node.

JSON node:

Just add a *Json* node onto the canvas and wire it to the mqtt node.

Function node (distance):

Add a *Function* node onto the canvas, wire it to the JSON node and double click on it to insert the following code:

```
var msg= {payload: msg.payload};
var Mlat={payload:  x};  // change x by your latitude
var Mlon= {payload: y}; // change y by your longitude
var lat={payload: msg.payload.__location.lat};
var lon={payload: msg.payload.__location.lon};
```

```
var radlat1 ={payload: Math.PI*(Mlat.payload)/180};

var radlat2 ={payload: Math.PI*(lat.payload)/180};

var theta ={payload: Mlon.payload-lon.payload};

var radtheta ={payload: Math.PI*(theta.payload)/180};

var dist ={payload:
Math.sin(radlat1.payload)*Math.sin(radlat2.payload)+Math.cos(radlat1.payload)*Math.cos(radlat2.payload)*Math.cos(radtheta.payload)};

var d1={payload: Math.acos(dist.payload)};

var d2 ={payload: (d1.payload)* 180/Math.PI};

var d3={payload: (d2.payload)*60*1.1515};

var distance={payload: (d3.payload)*1.609344};

var bool= {payload: distance.payload <= X }; //change  X by the Km value you want

if(bool.payload === true){

   return msg;

}
```

Function node (temperature alert):

Add a *Function* node onto the canvas, and double click on it to insert the following code:

```
var msg= {payload: msg.payload};

var temp={payload: msg.payload.payload.temperature_catnip};

var bool= {payload: temp.payload >= x }; // change x by the temperature value

if(bool.payload === true){

   return msg;

}
```

Remark:

The temperature_catnip can have another name you can get this name from the message in your topics dashboard.
For example if in your dashboard you got as message:

{ "payload": { "created_at": "2016-05-27T07:32:04.256Z", "distance_maxbotix": 205, "alert_distance_maxbotix": false, "temperature_air_carte": 24, "alert_temperature_air_carte": false }, "__location": { "lat": "1", "lon": "113" } }

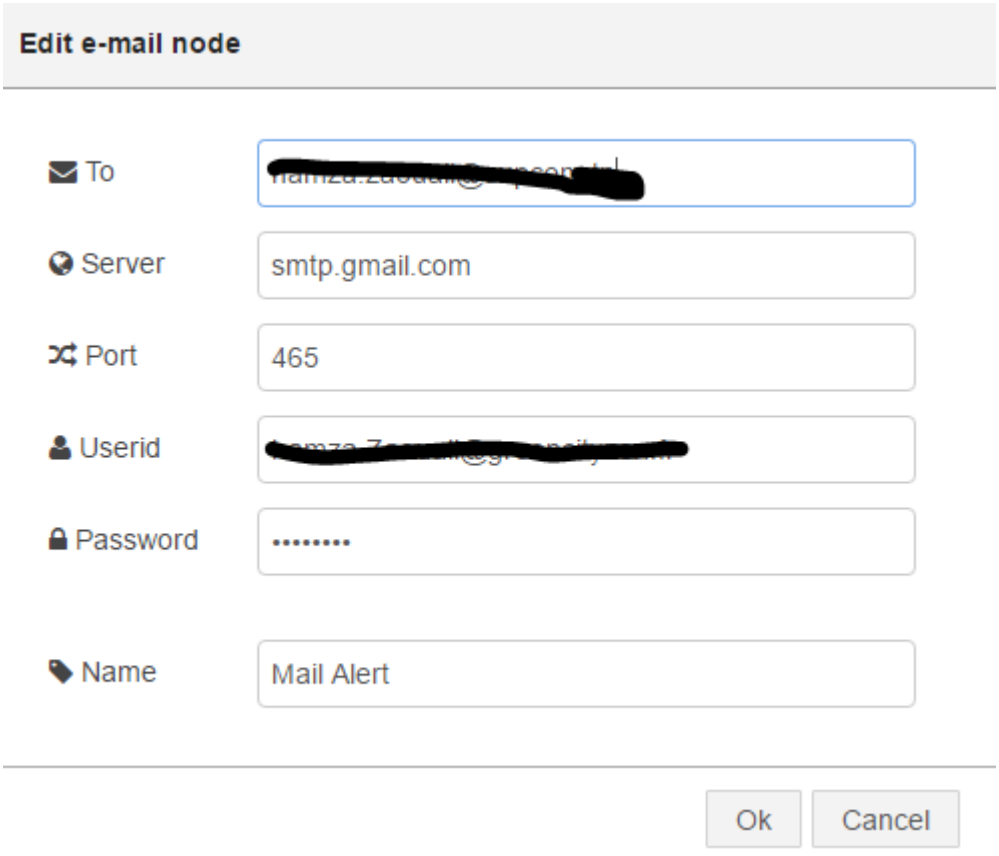
Then the field temperature_catnip should be temperature_air_carte.

Email node:

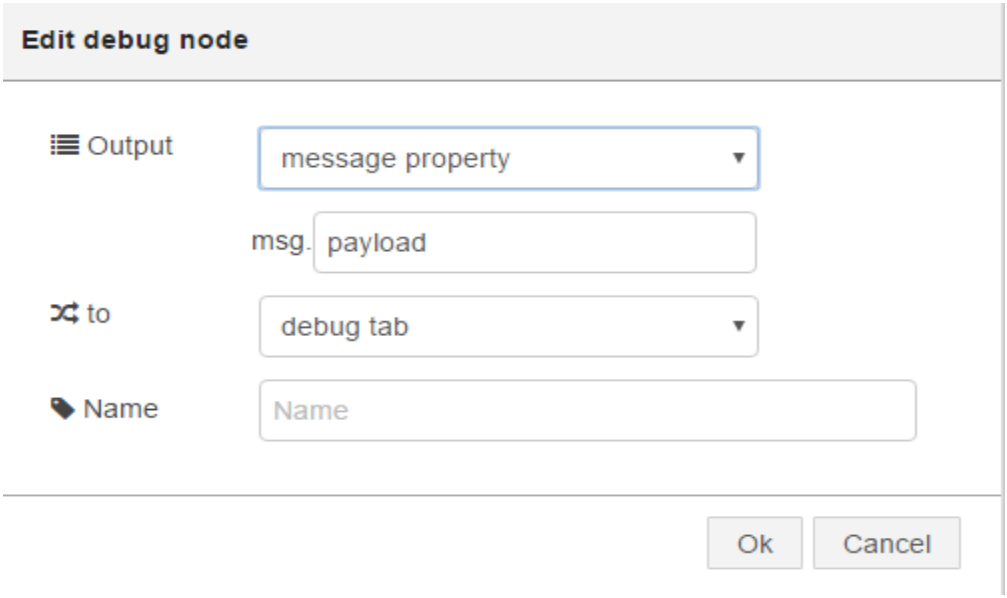
In order to get your data by email, Click on **e-mail node** and configure it:
- add the email address to which you will send the alert in the field **to**.
- You google address in the field **Userid** and your password.

***Note*: - make sure your gmail account setting "Access to less secure apps" is ON**

-your password is 8 characters, not longer



Debug node:



Add a debug node and configure it like shown above.

Results:

If there is a device which the temperature indicated by the sensors exceed the value you set up in the area you set up too you will get a debug message in node red and you will get the same message in your email address.

Remark:

If you want to just generate the temperature alert for a given device you have to:

- Subscribe to the device topic.
- Delete the distance node and keep the temperature node.