

# 1 Getting start with Node-RED

## 1.1 Things to know

To install node-RED into a XAP device, you need

- an empty USB Stick
- the Node-RED\_package.zip available on productselection.net

## 1.2 Procedures

### 1.2.1 Install Node-RED on XAP device

Follow this procedure to install Node-RED on XAP device.

*Note: Autorun scripts from external storage option in XAP machine\_config -> System settings -> Services must be enabled.*

Step	Action
1	Extract the content of the folder “Node-RED_installer” into the root folder of a USB stick.
2	Plug the USB stick into the XAP device: it buzzes <b>one time</b> to indicate the process is started.  <i>Note: If XAP device does not emit any sound this means the USB is not working or the content is not present.</i>
3	<b>The installation procedure usually takes up to three minutes.</b> Once the procedure successfully finishes, the XAP device <b>buzzes three times.</b>
4	Remove the USB sick to automatically reboot the XAP device.  <i>Note: Please wait two minutes to get the device up and running</i>
5	After installation, in order to configure node-RED you need to open web browser and digit the following url:  <u><a href="http://&lt;XAP IP address&gt;:1880">http://&lt;XAP IP address&gt;:1880</a></u>

## User interface

At the first time, the **Node-RED** interface appears as shown in the picture below:

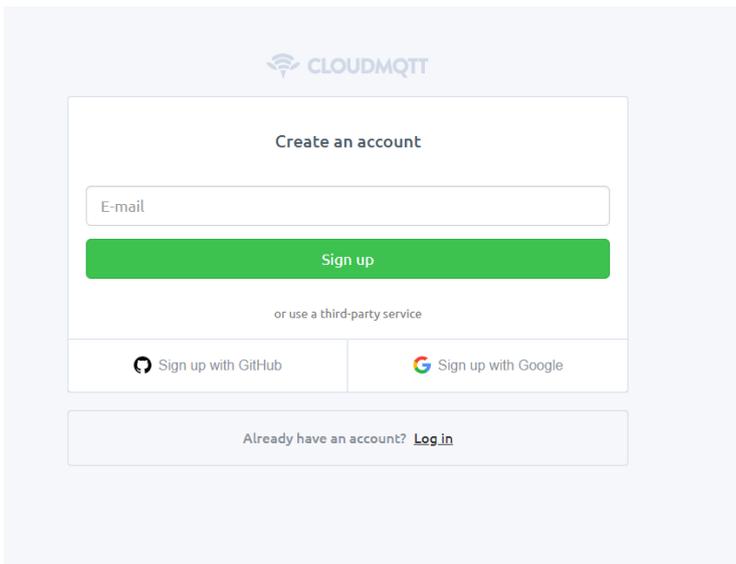


## 1.3 Procedures

To do a communication test with Node-RED and tags that are running on XAP device, you can use a free MQTT solution. Follow the procedures below.

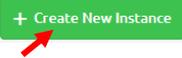
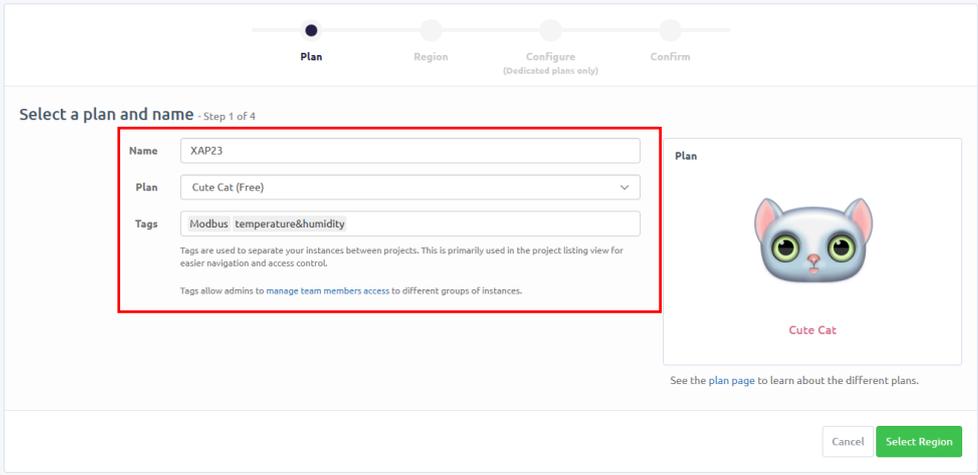
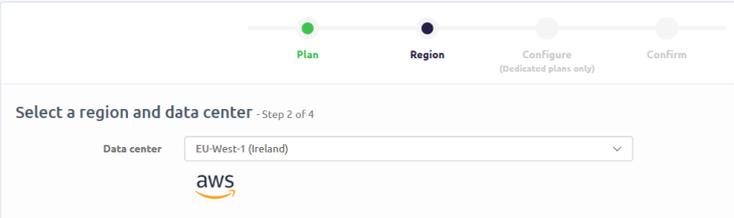
### 1.3.1 Create a CloudMQTT account

Access to <https://customer.cloudmqtt.com/signup/> and provide your information to create a new Free account.



Follow the indication on screen to finish the registration procedure.

### 1.3.2 Create a new Instance

Step	Action
1	Access to CloudMQTT account.
2	Click to  button in the top right corner.
3	<p>a) Enter an identification name in <b>Name</b> field.</p> <p>b) Select <b>Cute Cat (Free)</b> value form brop-down menu.</p> <p><i>Note: see <a href="https://www.cloudmqtt.com/plans.html">https://www.cloudmqtt.com/plans.html</a> for details about available plans</i></p> <p>c) Enter option tags value in <b>Tags</b> field.</p>  <p>Click <b>Select Region</b> button to continue.</p>
4	<p>Select the preferred <b>Data center</b> form the drop-down menu.</p>  <p>Click on <b>Review</b> button to continue.</p>
5	In the next page, check that all the information provided are valid then click on <b>Create instance</b> to finish the procedure.

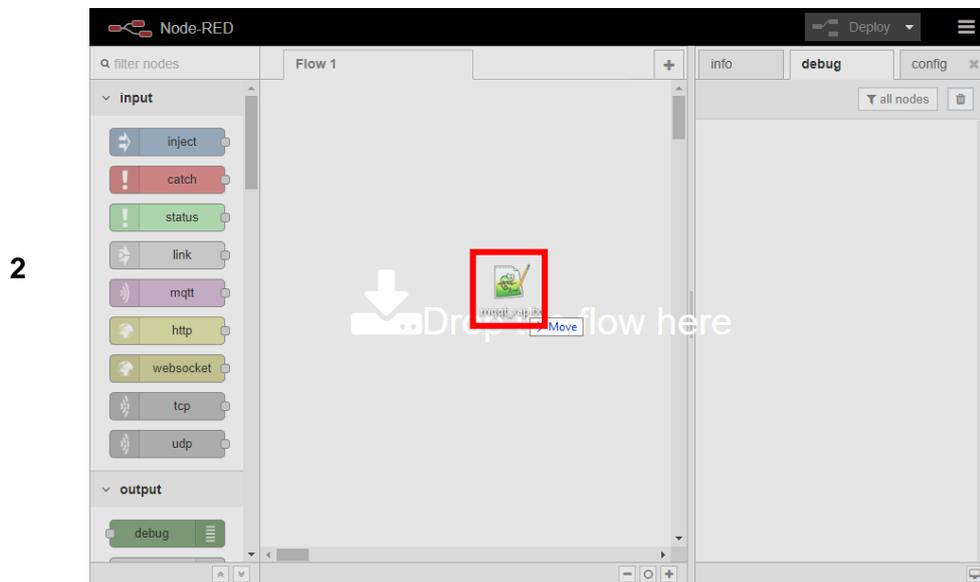
### 1.3.3 Import the Node-RED example project

The file *Generic\_MQTT\_example.txt* is a Node-RED flow available in the “*Node-RED package.zip*” folder. Follow this procedure to import and set the example within your references:

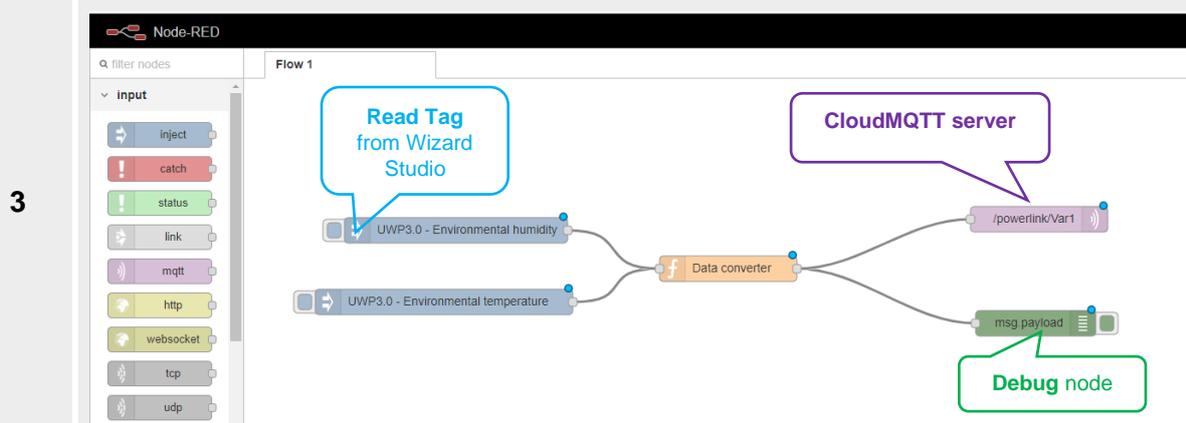
- ❗ In the Wizard Studio project, the **Security** option (**ProjectView > Security**) must be disabled.
- 📄 You must change all the tags reference and the MQTT server settings in the example file.

Step	Action
1	Open the Node-RED interface in a web browser.

Drag&Drop the *Generic\_MQTT\_example.txt* file onto Flow area.

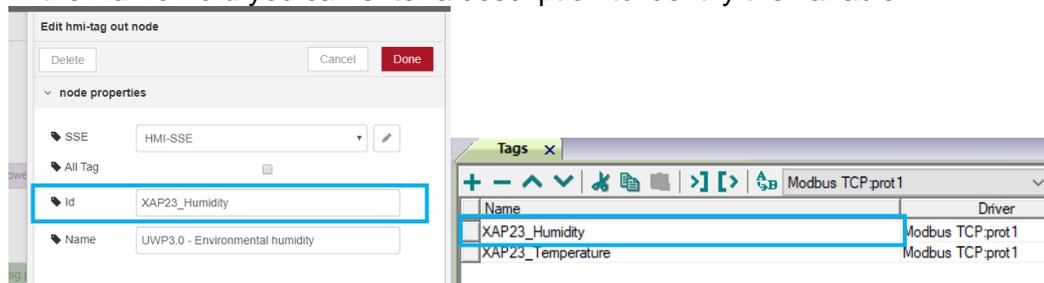


The Node-RED flow related to the imported file is shown:



Double click on the **Read Tag node** to enter the settings page:  
 Set in the **Id** field the same Tag name that is shown in Wizard Studio in **Tags** tab.  
 In the **Name** field you can enter a description to identify the variable.

4

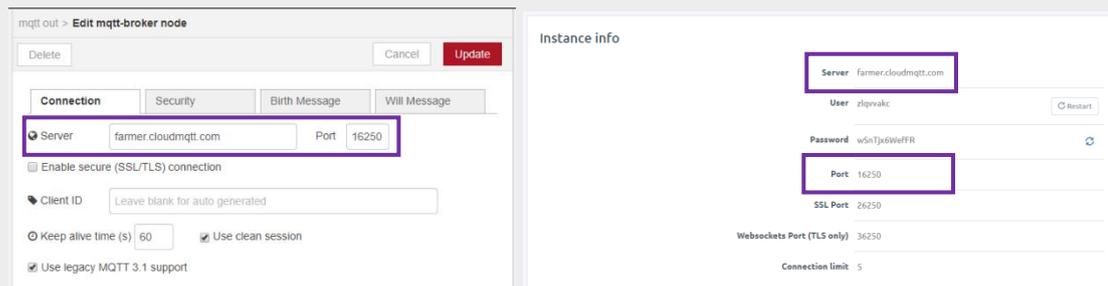


Click on **Done** button to store the changes.

Double click on the **CloudMQTT server node** to enter the settings page.  
 This node must be set with the CloudMQTT server information. To get these information access to the CloudMQTT and access to the **Instance** settings page:

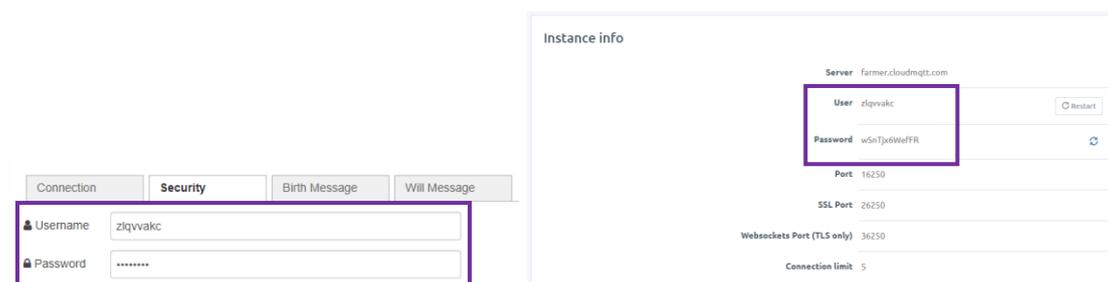
In the **Server** field enter the value present in the Server field.  
 In the **Port** field enter the value present in the Port field.

5



Click the **Connection** tab to enter:  
 in the **Server** field the value present in the Server field.  
 in the **Port** field the value present in the Port field.

6



Click on **Update/Done** button to store the changes.

Click to **Deploy** button to compile the project:



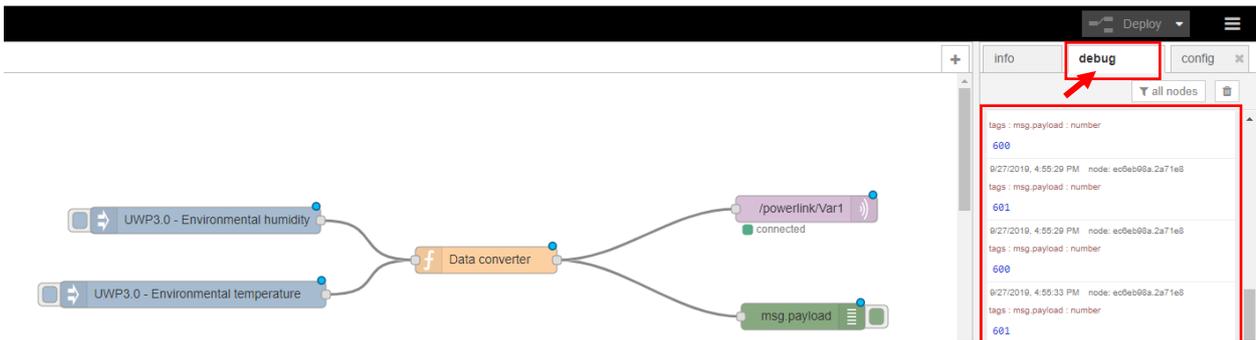
7 The status indicator is shown below the CloudMQTT server node



8 Click to **Debug tag** to see the updated value that are sent to MQTT broker (CloudMQTT) in real-time.

### Example

In the example shown below the project is running on Node-RED. The **UWP3.0 – Humidity** and **UWP3.0 – Temperature** tags represent two variables read from Wizard Studio via Modbus TCP protocol: their value is sent to CloudMQTT free portal every time their values changes.



The screenshot shows a Node-RED workflow. On the left, two input nodes are labeled 'UWP3.0 - Environmental humidity' and 'UWP3.0 - Environmental temperature'. These connect to a central 'Data converter' node. The output of the 'Data converter' node is split into two paths: one leading to a 'powerlink/Var1' MQTT node (with a 'connected' status indicator) and another leading to a 'msg.payload' node. On the right, the 'debug' console is open, displaying a log of messages with the following structure:

```

tags : msg.payload : number
600
9/27/2019, 4:55:29 PM node: ec04eb08a.2a71e0
tags : msg.payload : number
601
9/27/2019, 4:55:29 PM node: ec04eb08a.2a71e0
tags : msg.payload : number
600
9/27/2019, 4:55:33 PM node: ec04eb08a.2a71e0
tags : msg.payload : number
601
    
```