

Practical IoT (Internet of Things)

BSCIS – DCIS, PIEAS

Lab 05 - Setting Up an IoT Server with Node-RED, Mosquitto, and Dashboard

Objective

- Deploy a local IoT server with Node-RED (workflow automation).
- Install Mosquitto (MQTT broker).
- Setup Grafana/Node-RED Dashboard for visualization.

Prerequisites

Virtual Machine Software:

- VirtualBox, VMware, or Hyper-V. (We will be using VirtualBox)
- Virtual Box Download Page: <https://www.virtualbox.org/wiki/Downloads>

OS Image:

- Raspberry Pi Desktop (We can use any Linux Distribution for this server, but this manual is based on the Raspberry Pi Desktop).
- RPi Desktop Download Page: <https://www.raspberrypi.com/software/raspberry-pi-desktop/>

Hardware:

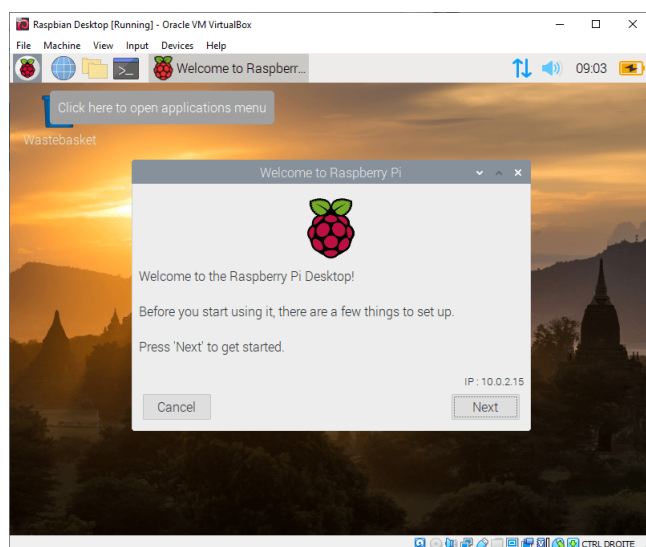
- Wemos D1 Mini (ESP8266) with any sensor (e.g., DHT11 or simple a push button).

Step 1: Set Up the Virtual Machine

Create a New VM:

- Install the virtual box on your machine / laptop. The link is already given in prerequisite section.
- Download the Raspberry Pi Desktop Image and create a new virtual machine with this image. A video guide for setup is also provided on canvas. The text instructions to install the virtual machine could be found at:

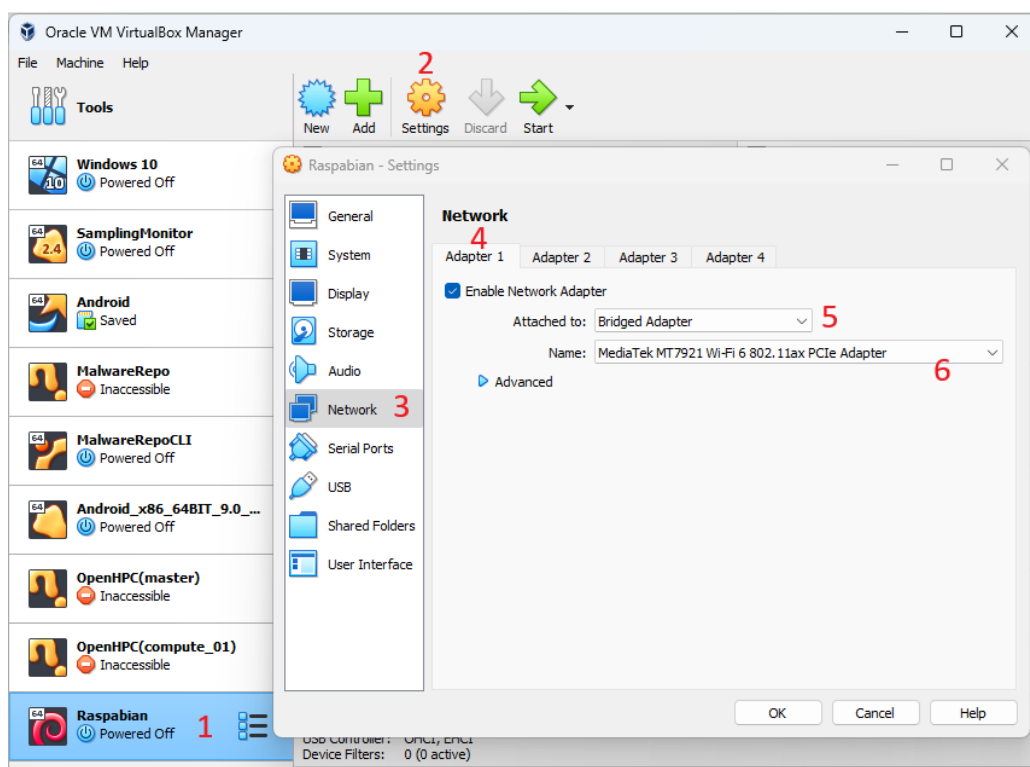
<https://roboticsbackend.com/install-raspbian-desktop-on-a-virtual-machine-virtualbox>



Configure Network Settings:

Set the network adapter to "Bridged" mode to allow the VM to have its own IP address on the local network.

1. Open virtual Box and Select the VM you just created.
2. Click settings gear icon, the VM Settings Dialog Box opens.
3. Select the Network Tab on Left list in settings.
4. Select the Adapter you want to configure (Most Cases – Adapter 1).
5. Enable Network Adapter and Select Bridged Adapter.
6. Select Your Wifi/Wired Adapter which would be used to reach the outer world.
7. Click OK to save the settings.
8. Powerup / Reset the virtual machine and verify that you are getting correct IP inside the Guest OS. The Network should be same as your Laptop is connected to but different IP.



Step 2: Install Mosquitto MQTT Broker

Installing Mosquitto Broker:

- Detailed Guide: <https://randomnerdtutorials.com/how-to-install-mosquitto-broker-on-raspberry-pi/>
- You need to follow "Installing Mosquitto Broker on Raspberry Pi OS" section of guide to accomplish this part.
- Commands Summary (to be executed on Linux terminal):

```
sudo apt update && sudo apt upgrade  
sudo apt install -y mosquitto mosquitto-clients  
sudo systemctl enable mosquitto.service  
mosquitto -v
```

Making Broker Public / Anonymous:

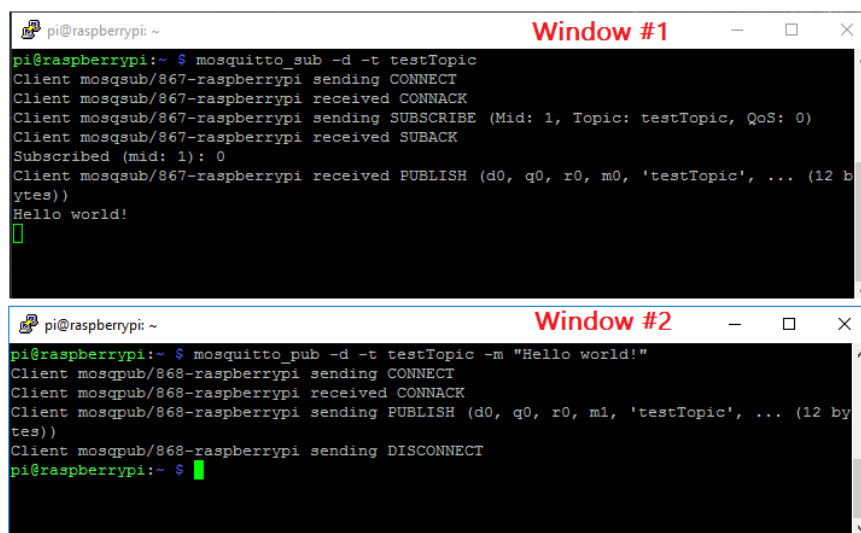
- Detailed Guide: <https://randomnerdtutorials.com/how-to-install-mosquitto-broker-on-raspberry-pi/>
- You need to follow “Mosquitto Broker Enable Remote Access (No Authentication)” section of guide to accomplish this part.
- Commands Summary (to be executed on Linux terminal):

```
sudo nano /etc/mosquitto/mosquitto.conf  
    listener 1883  
    allow_anonymous true  
sudo systemctl restart mosquitto
```
- Get your VM IP and test your broker with MQTT Explorer.
- Before proceeding further Make sure your broker is successfully able to connect with MQTT Explorer. You need to Use Your VM IP and Broker URL

Securing Mosquitto Broker (recommended but optional for this lab):

- Detailed Guide: <https://randomnerdtutorials.com/how-to-install-mosquitto-broker-on-raspberry-pi/>
- You need to follow “Mosquitto Broker Enable Remote Access (Authentication: user and password)” section of guide to accomplish this part.
- Commands Summary (to be executed on Linux terminal):

```
sudo mosquitto_passwd -c /etc/mosquitto/passwd YOUR_USERNAME  
sudo nano /etc/mosquitto/mosquitto.conf  
    per_listener_settings true  
    listener 1883  
    allow_anonymous false  
    password_file /etc/mosquitto/passwd  
sudo systemctl restart mosquitto  
sudo systemctl status mosquitto
```
- Get your VM IP and test your broker with MQTT Explorer.
- Now you need to supply your username and password to connect the broker (same which was used to secure broker).
- Before proceeding further Make sure your broker is successfully able to connect with MQTT Explorer. By supplying username and password.



The image shows two terminal windows, Window #1 and Window #2, running on a Raspberry Pi. Window #1 shows a subscriber command and its output, while Window #2 shows a publisher command and its output.

```
Window #1
pi@raspberrypi: ~
pi@raspberrypi:~$ mosquitto_sub -d -t testTopic
Client mosqsub/867-raspberrypi sending CONNECT
Client mosqsub/867-raspberrypi received CONNECT
Client mosqsub/867-raspberrypi sending SUBSCRIBE (Mid: 1, Topic: testTopic, QoS: 0)
Client mosqsub/867-raspberrypi received SUBACK
Subscribed (mid: 1): 0
Client mosqsub/867-raspberrypi received PUBLISH (d0, q0, r0, m0, 'testTopic', ... (12 bytes))
Hello world!

```

```
Window #2
pi@raspberrypi: ~
pi@raspberrypi:~$ mosquitto_pub -d -t testTopic -m "Hello world!"
Client mosqpub/868-raspberrypi sending CONNECT
Client mosqpub/868-raspberrypi received CONNECT
Client mosqpub/868-raspberrypi sending PUBLISH (d0, q0, r0, m1, 'testTopic', ... (12 bytes))
Client mosqpub/868-raspberrypi sending DISCONNECT
pi@raspberrypi:~$

```

Step 3: Install Node-RED

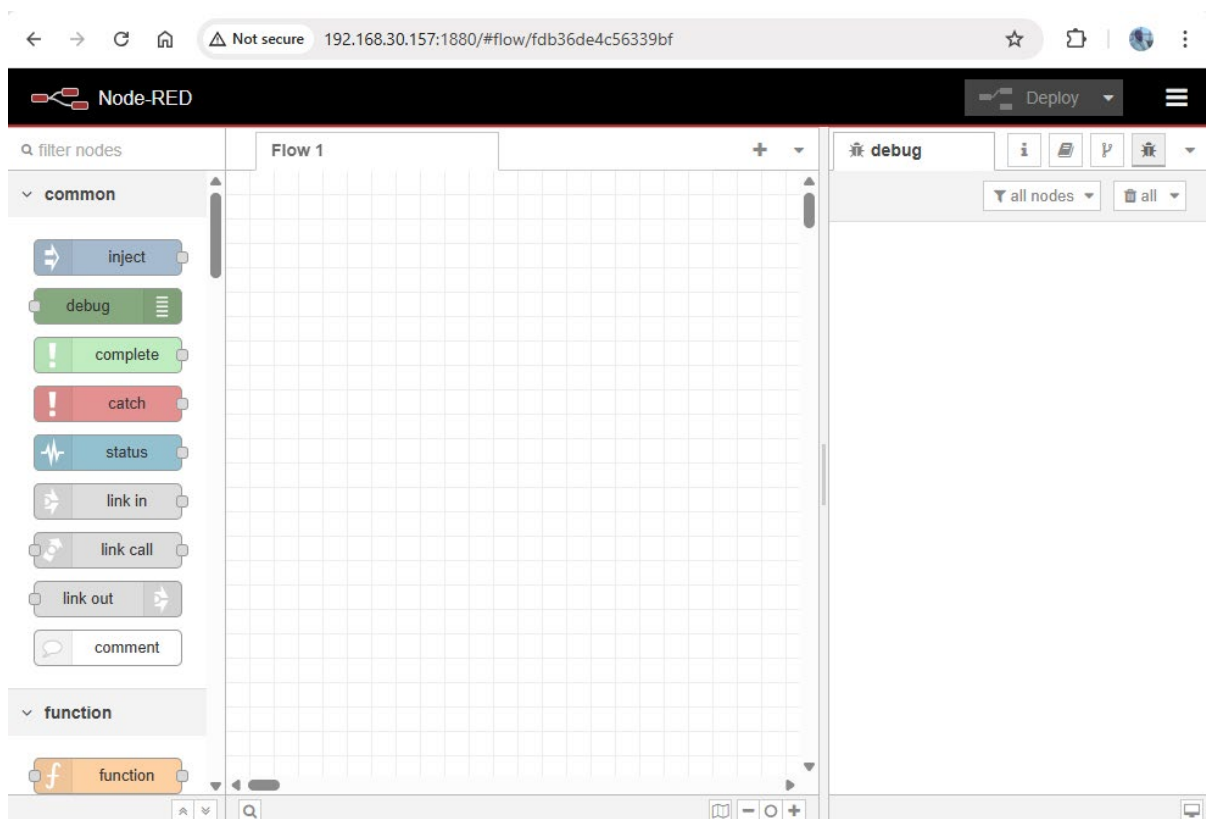
Installing Node-RED:

- Detailed Guide: <https://nodered.org/docs/getting-started/raspberrypi>
- You need to follow above complete guide to setup Node-RED on your VM. It may take 15-20 minutes to complete the installation. As there are many steps involved including installation of Node.js and other pre-requisites.
- Commands Summary (to be executed on Linux terminal):

```
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)
node-red-pi --max-old-space-size=256
sudo systemctl enable nodered.service
sudo systemctl disable nodered.service
```
- After installation is complete, you can access Node-RED editor at: http://vm_ipaddress:1880

Securing Node-RED (recommended but optional for this lab):

- Detailed Guide: <https://nodered.org/docs/user-guide/runtime/securing-node-red>
- You need to follow “Enabling HTTPS Access” and “Editor & Admin API security” sections of above guide to secure Node-RED on your VM based on some Username/password.
- OAuth/OpenID based authentications are best for multi user environment and production use. **For this lab skip OAuth/OpenID based authentications.**
- After installation is complete, you can access Node-RED editor at: http://vm_ipaddress:1880
- Now you need to specify username / password to access the editor.



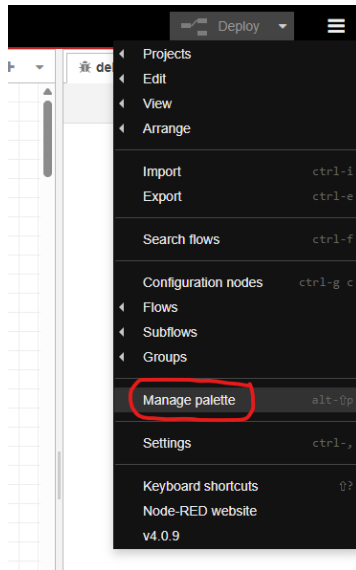
Step 4: Install Node-RED Dashboard

Access Node-RED Interface:

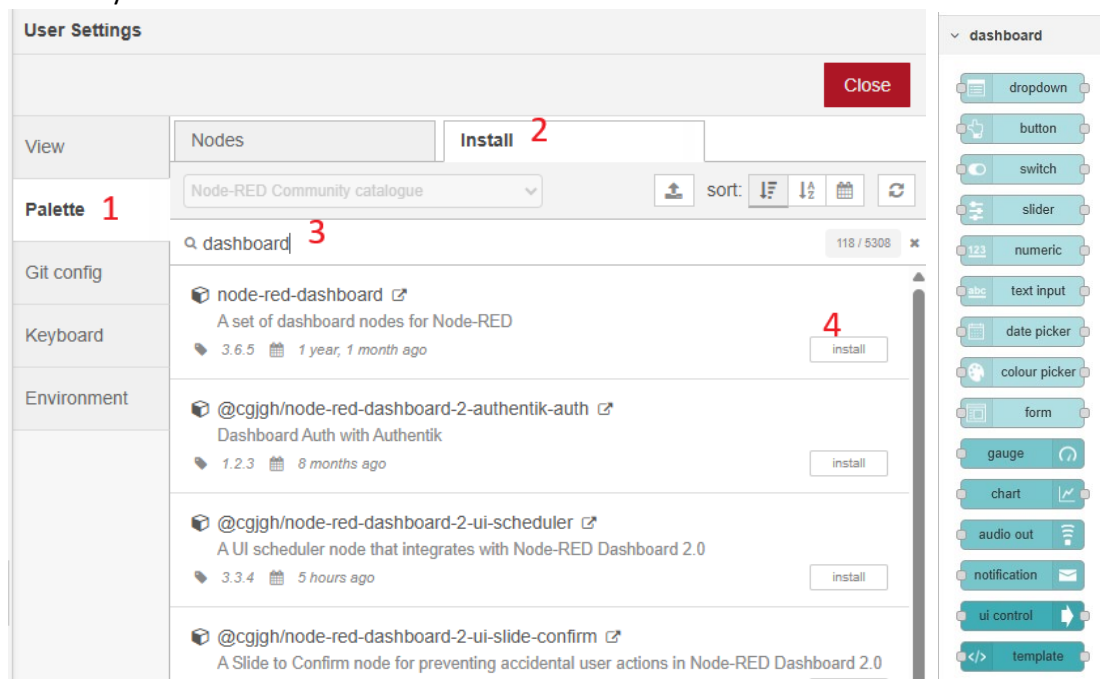
- Open a web browser and navigate to <http://<VM IP Address>:1880>.

Install Dashboard Nodes:

- Click on the menu (three horizontal lines) in the top-right corner.
- Select "Manage palette."



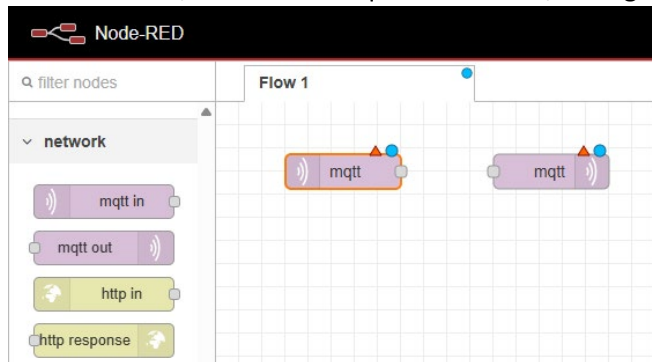
- Go to the "Install" tab and search for node-red-dashboard.
- Click "Install" to add dashboard nodes to Node-RED. Please note there are many community dashboards available. For this lab we are using only node-red-dashboard.
- Confirm your installation action and dashboard nodes will be available in few seconds.



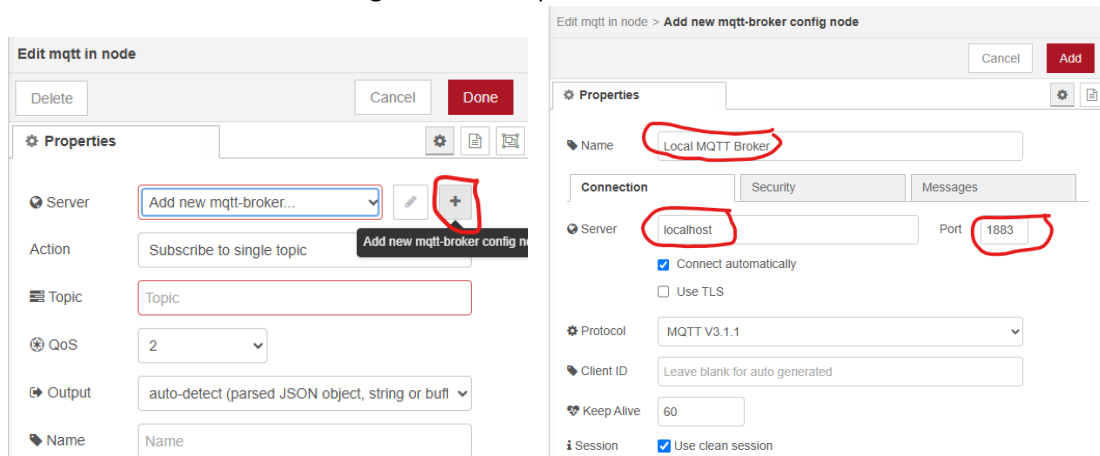
Step 5: Create a Simple Flow to Test the Setup

Add MQTT Nodes:

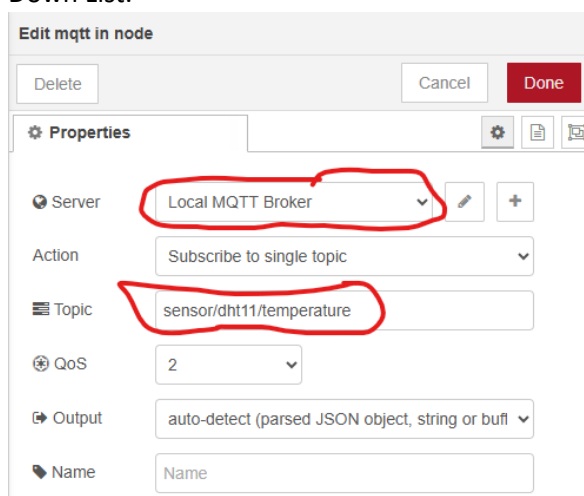
- Drag an "mqtt in" node and an "mqtt out" node onto the workspace. Little red triangle on nodes tell that, these node require attention / configuration.



- Double Click on nodes to configure the Mosquitto broker at localhost : 1883.

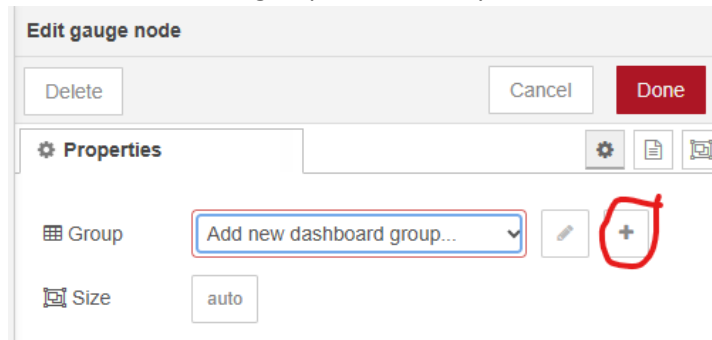


- Finally Click Add to Add New MQTT Broker. The newly add broker node will be available on all flows of Node-RED.
- You also need to specify MQTT topic e.g. "" to subscribe the "Mqtt in" node and click Done.
- For further nodes you only need to select previously added Mosquitto Broker from Drop Down List.



Add Dashboard Nodes:

- Drag a "gauge" node from the dashboard section onto the workspace.
- Connect the "mqtt in" node to the "gauge" node.
- Configure the Gauge Node:
 - Add new dashboard group if not already added.



Edit gauge node

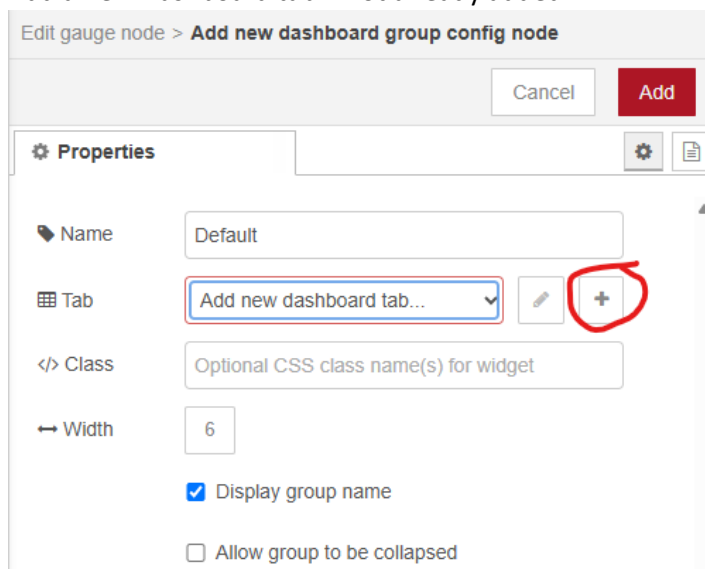
Delete Cancel Done

Properties

Group Add new dashboard group... +

Size auto

- Add a new Dashboard tab if not already added



Edit gauge node > Add new dashboard group config node

Cancel Add

Properties

Name Default

Tab Add new dashboard tab... +

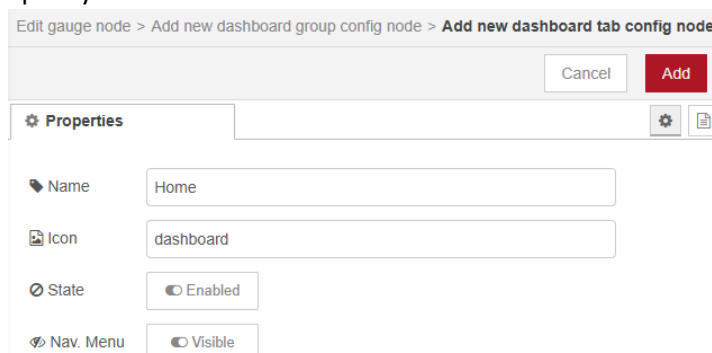
Class Optional CSS class name(s) for widget

Width 6

☒ Display group name

☐ Allow group to be collapsed

- Specify the Tab name and icon and click Add. You can also leave default.



Edit gauge node > Add new dashboard group config node > Add new dashboard tab config node

Cancel Add

Properties


Name Home

Icon dashboard

State Enabled

Nav. Menu Visible

- Finalize the gauge settings. Specify the units and range of values then click done.



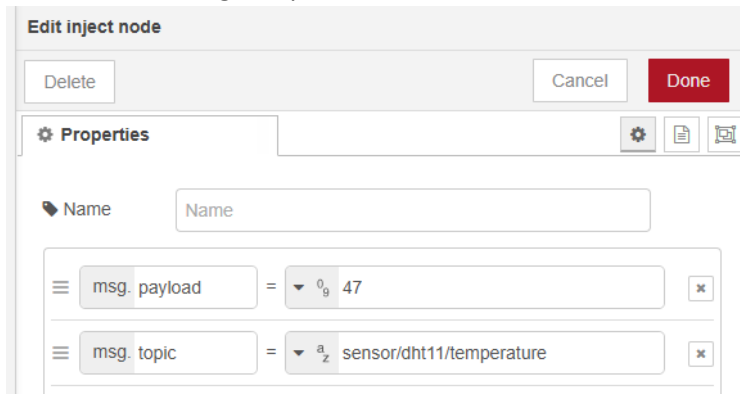
Units C

Range min 0 max 100

Colour gradient

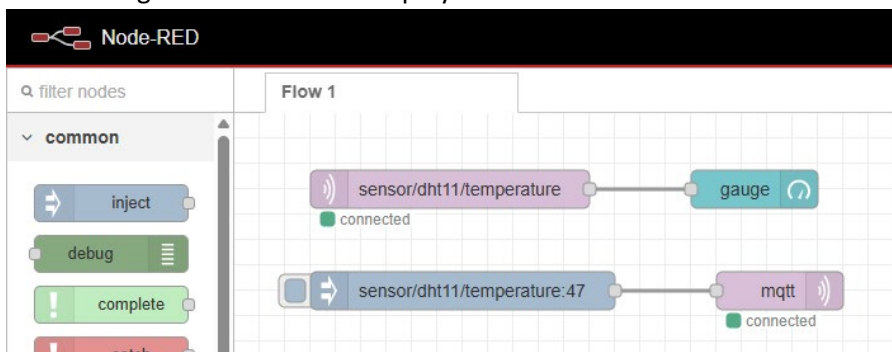
Add Inject Node:

- Drag a "inject" node from the common section onto the workspace.
- Connect the inject node to the "mqtt out" node.
- Configure the Inject node to send a number as payload and topic to "mqtt out" node. Click Done when editing complete.

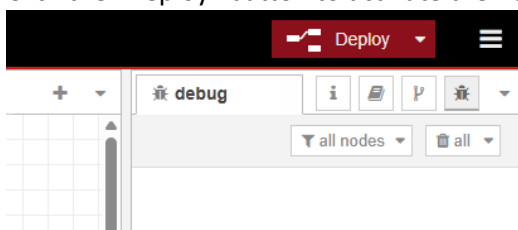


Deploy the Flow:

- Final settings of nodes before deploy will be like this:

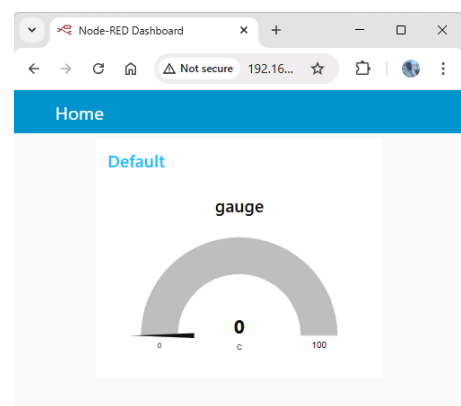


- Click the "Deploy" button to activate the flow.

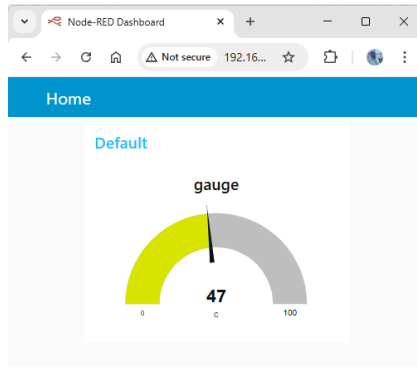


Step 6: Access the Dashboard

- Open Browser on laptop and navigate to http://<VM_IP_Address>:1880/ui to view and interact with your dashboard.
- Publish test messages (via inject node) to the MQTT topic to see the gauge update on the dashboard.



- See the updated value on the gauge as it was published through the inject button.



- Now you can publish values to this topic through your ESP8266 controller and DHT-11 sensor. The values will reflect on the gauge.

Troubleshooting Tips:

- Ensure that the VM's firewall allows traffic on ports 1880 (Node-RED) and 1883 (Mosquitto).
- If encountering connection issues between Node-RED and Mosquitto, verify that both services are running and correctly configured.
- For external devices to connect to the Mosquitto broker, ensure that `allow_anonymous` is set to true in the Mosquitto configuration.

Submission Requirements:

Submit a single PDF containing:

- **Screenshots:** VM setup, Node-RED flows, Node-Red dashboards, Node Configurations.
- **Codes:** Wemos code and Node-RED flow JSON. (You can export any Node-RED flow as json).
- **Analysis:** Explain how MQTT, Node-RED, and dashboards interact.
- **Short Description:**
 - The steps you followed.
 - How you verified the functionality.
 - Any challenges you encountered and how you addressed them.

Grading Rubric:

Criterion	Points	Description
VM Setup and Configuration	30	VM is correctly configured with appropriate resources, along with network settings.
Installation of Mosquitto Broker	25	Mosquitto is installed and configured flawlessly.
Node-RED Installation and Flows	25	Node-RED is installed correctly.
Dashboard Implementation	20	Dashboard is intuitive, user-friendly, and aesthetically pleasing.
Total	100	