

Configuration Guide

Senet_nodeRed Setup

Version 1.0

REVISION HISTORY

Version	Date	Notes	Contributor	Approver
1.0	4 Apr 2018	Initial Release	Seokwoo Yoon	Jonathan Kaye

CONTENTS

1	Overview.....	4
2	Data Flow Architecture.....	4
3	Gateway Factory Reset.....	4
4	Connect the Gateway	5
5	Log into the Gateway	6
5.1	Logging into Gateway Web Interface	6
5.2	Updating Gateway Firmware.....	6
5.3	LoRa Packet Forwarding Set Up.....	6
6	Senet Configuration.....	7
6.1	Senet Account Setup	7
6.2	Registering Your Gateway	8
6.3	Adding Devices an Application	9
7	RS1xx Configuration	10
8	Setting up MQTT Broker on Amazon AWS	11
8.1	Setting up AWS server	11
8.2	Installing MQTT Broker on AWS instance.....	14
9	MQTT Integration on Senet.....	14
10	node-RED Setup.....	16
10.1	Install node.js Package Manager and node.js	16
10.2	Install JSON Files	17
10.3	Installing node-RED	17
10.4	Installing node-RED-Dashboard.....	18
10.5	Updating RS1xx-Demo-Config File.....	19
10.6	Running node-RED with Default Settings	20
10.7	Establishing MQTT Broker Connections	20
10.8	Loading node-RED User Interface	22
11	Appendix.....	24
11.1	Troubleshooting Tips for Sentrius Gateway Connections	24
11.2	Troubleshooting Tips for Senet Connections	24
11.3	Troubleshooting Tips for node_RED (MQTT Broker Links) Connections.....	24

1 OVERVIEW

This guide provides instructions on configuring a Sentrius RG1xx gateway and Sentrius RS1xx sensor using the Senet server and node-RED web-based user interface.

Note: Step-by-step instructions, screen shots, and pictures are based on the Sentrius RG191 and Sentrius RS191, but the same is applicable for the Sentrius RG186 and Sentrius RS186. Differences are noted.

For more detailed information on how to use all Sentrius gateway and sensor features, please see the *Sentrius RG1xx User Guide*, available from documentation tab at: www.lairdtech.com/products/rg1xx-lora-gateway and the *Sentrius RS1xx User Guide*: www.lairdtech.com/products/RS1xx-LoRa-Sensors.

2 DATA FLOW ARCHITECTURE

The following block diagram (Figure 1) displays the overall architecture described in this demo. This is a high-level representation of how the sensor LoRaWAN data is transferred through the gateway (which plays a role of packet forwarder) to Senet server where the data packets are processed. They are then forwarded to the MQTT Broker. The node-RED application contains an MQTT client that receives sensor data and displays it on a browser-based user interface.

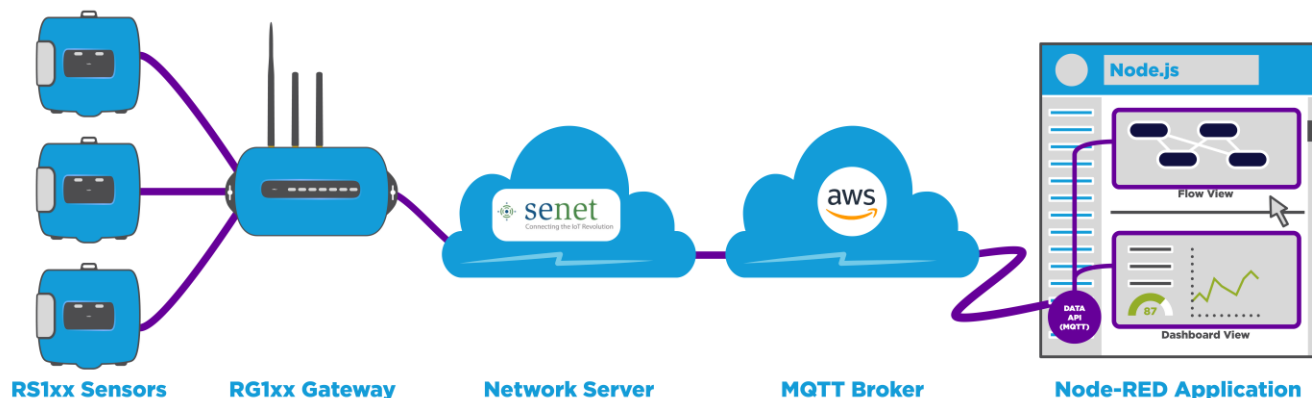


Figure 1: Data flow architecture

This guide shows a step-by-step approach to setting up a test LoRa network. It starts on the left side of the block diagram with the Gateway configuration and progresses toward the UI application on the right. Sensor devices are added when configuring the Senet backend.

3 GATEWAY FACTORY RESET

If setting up a previously configured gateway, we recommend that you clear any prior settings by resetting the gateway to its factory default values. To do this, complete the steps described in the *Factory Reset* section of the RG1xx User Guide. You can access this guide on the [RG1xx product page](#) from the Documentation tab.

4 CONNECT THE GATEWAY

To use the gateway, you must power it up and access the web interface via the Ethernet port. To do this, follow these steps:

1. Follow the label on the box and connect the three antennas. Refer to Antenna Configuration for additional information.
2. Connect the power supply (see #2 in Figure 2).
3. Connect the gateway to your router (#3 Figure 2) using the Ethernet cable (#1 in Figure 2).

Your gateway is now connected and ready.

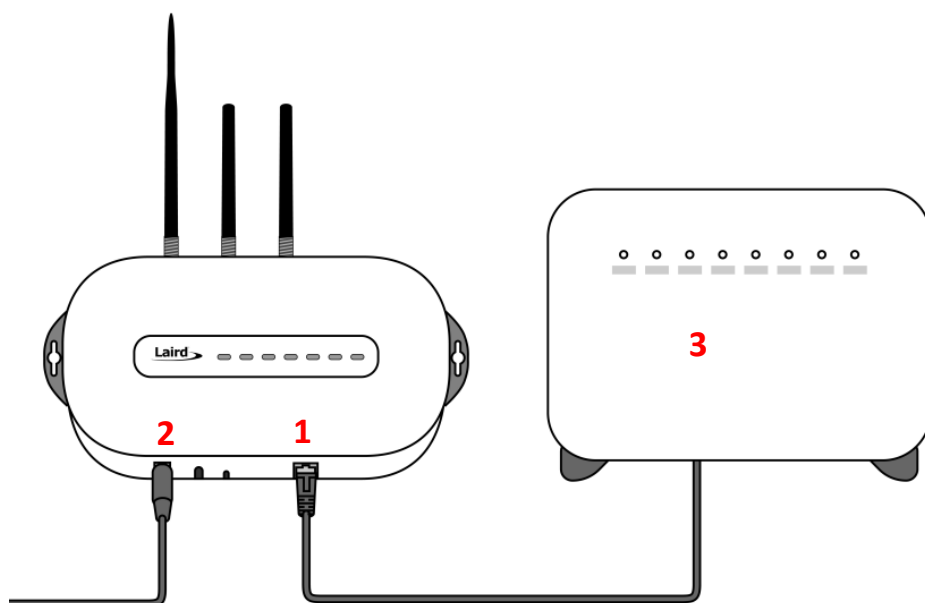


Figure 2: Connecting the gateway

Antenna Configuration

To configure the antenna properly, do the following:

1. Attach the two shorter antennas to the 2.4/5.5 GHz (Wi-Fi) ports.
2. Attach the third and longer antenna to the 868 MHz/900 MHz (LoRa) port.

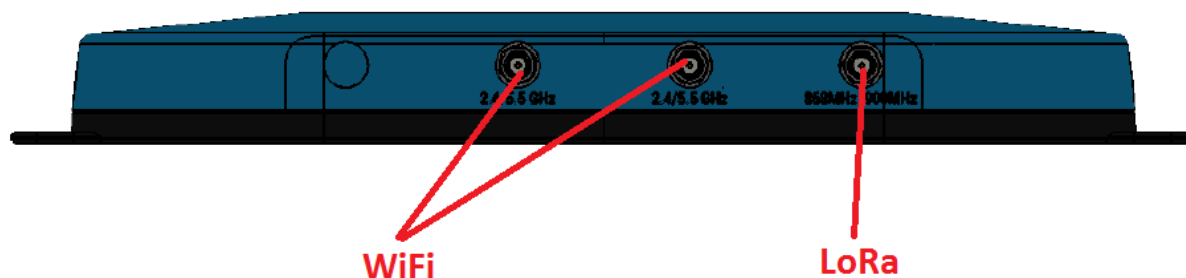


Figure 3: Antenna configuration

5 LOG INTO THE GATEWAY

5.1 Logging into Gateway Web Interface

To log into the gateway web interface, follow the steps from the *RG1xx User Guide's* Log into the Gateway section. This guide is accessible from the website's [RG1xx product page](#) on the Documentation tab.

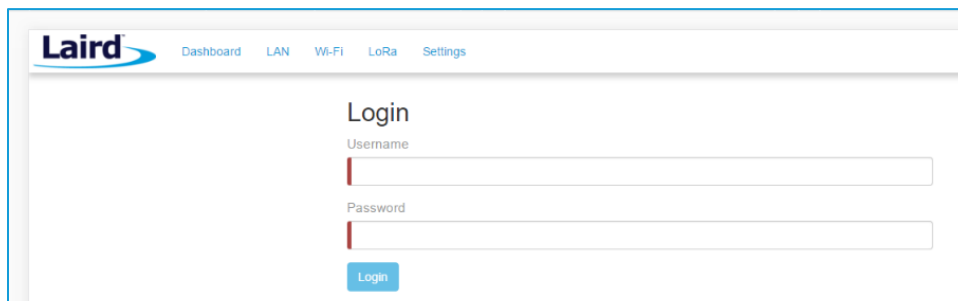


Figure 4: Gateway interface login screen

Note: For this guide, we assume the LAN or Wi-Fi connections are successfully set up and the user login credentials are updated. We recommend that you update the credentials as soon as possible to minimize a potential security risk. Update these credentials on the Settings page.

5.2 Updating Gateway Firmware

Update the firmware by clicking **Settings > Update Firmware**. (Figure 5)

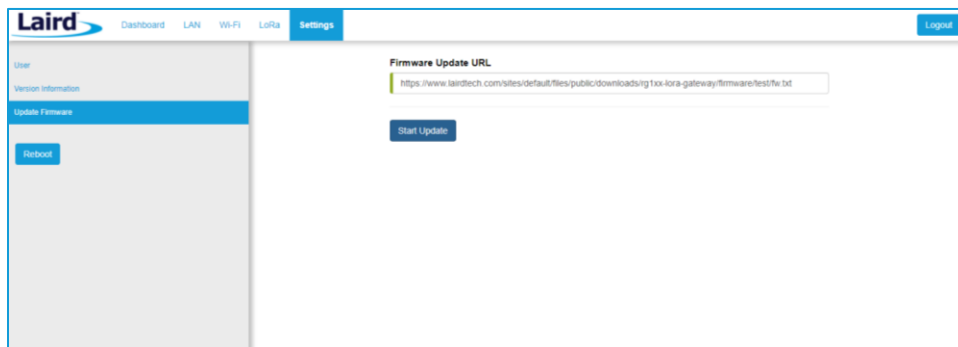


Figure 5: Updating gateway firmware

Enter the following URL: <https://www.lairdtech.com/products/rg1xx-lora-gateway/firmware/latest/fw.txt>

Note: It is recommended to have GA2 (v93.7.2.9) firmware or greater in order for this to function.

Once the gateway is finished updating the firmware, follow the on-screen instructions to reboot the device. This requires a login using your updated profile credentials.

5.3 LoRa Packet Forwarding Set Up

Simply select **Senet** on **LoRa > Preset**. The gateway connection status should change to true once the Preset is applied. (Figure 6)

The screenshot shows the 'Presets' configuration page in nodeRed. On the left, a sidebar lists 'Presets', 'Forwarder', 'Radios', 'Advanced', and 'Traffic'. The 'Presets' section is active, showing a table with columns 'Gateway Connected', 'Gateway EUI', 'Region Code', and 'Mode'. The 'Senet - US' preset is selected. A yellow warning box states: 'You may lose your LoRa settings when applying a preset!'. An 'Apply' button is visible. On the right, a sidebar shows the Senet logo and configuration details: Forwarder: semtech, Preset Server Address: collector.senetco.io, and Preset Upstream / Downstream Ports: 1700 / 1700.

Figure 6: Setting presets

Then, select **Forwarder** and confirm the followings.

- **collector.senetco.io** for Network Server Address
- **1700** for Port Up/Down

6 SENET CONFIGURATION

6.1 Senet Account Setup

Note: Before performing any actions, ensure that network port 1700 is open (LoRaWAN traffic port) as well as ports 1883 and 8883 (Senet MQTT traffic port). UDP traffic is sent over port 1700. TCP and TCP/TLS socket connections are made on ports 1883 and 8883. We recommend that you work with your IT group to provide access for this install.

To set up an account with Senet, follow these steps:

1. Go to <https://portal.senetco.io/#/home> and click **Create New Account** in the lower right of the window to start the process as shown in Figure 7. If you already have an account, click **Login** and skip to step four.

The screenshot shows the 'senet portal' account setup form. It includes fields for 'First Name', 'Last Name', 'Email Address', 'Username', 'Password', 'Confirm Password', and 'Company, inc.'. There is a checkbox for 'I agree to the Terms of Service' and a 'Create Account' button. A 'Login' link is also visible.

Figure 7: Senet account setup

Note: It is important that you have access to the email address entered in the account setup since Senet requires you to validate your email before proceeding with the rest of the setup.

2. Log in to Senet after the email is verified. (Figure 8)

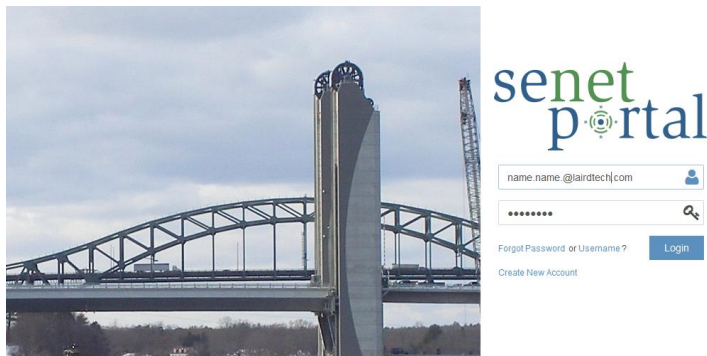


Figure 8: Senet login page

3. The dashboard appears as the home page once logged in.

6.2 Registering Your Gateway

To register your gateway with the Senet network, follow these steps:

1. Click **+** button on **Devices & Gateways** in Dashboard
2. Select **Gateway** (Figure 9)

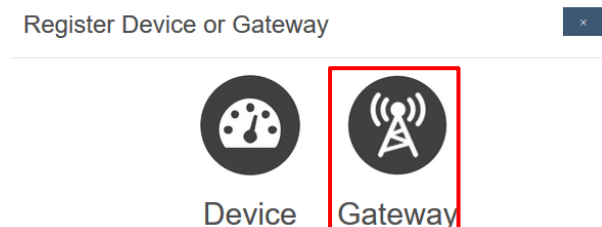


Figure 9: Senet gateway configuration page

3. Follow the prompts. When *Select Device Type* appears, choose **Semtech Packet Forwarder** (Figure 10).

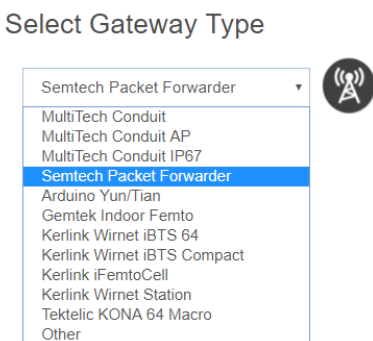


Figure 10: Semtech Packet Forwarder

- On the next page, register your gateway as shown in Figure 11. For *serial number*, enter the M2 EUI from the bottom of the RG1xx case (Figure 12), with colons removed. For *Gateway Vendor and Type*, enter **Laird**. For *Description*, enter your own description.

Register Gateway

Gateway Information

Please provide an alphanumeric serial number/identifier for this device.

In the Gateway Vendor and Type field provide details on the manufacturer of your Packet Forwarder.

Use the Description field to provide a short description used to identify this gateway.

For more information visit <http://docs.senetco.io>

Serial Number	XXXXXXXXXXXXX...
Gateway Vendor and Type	
Description	

< Back
Next >

Figure 11: Gateway Information

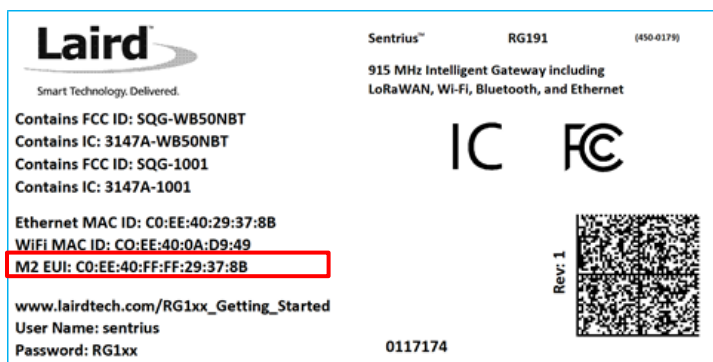


Figure 12: On a bottom label, M2 EUI corresponds to Serial number.

- Continue to follow the prompts and click **Register** to complete setup. Senet sends an approval email for gateway registration.

6.3 Adding Devices an Application

- click **+ button** on **Devices & Gateways** in Dashboard.
- Select **Device**. (Figure 13)

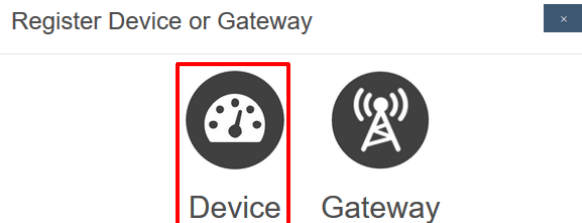
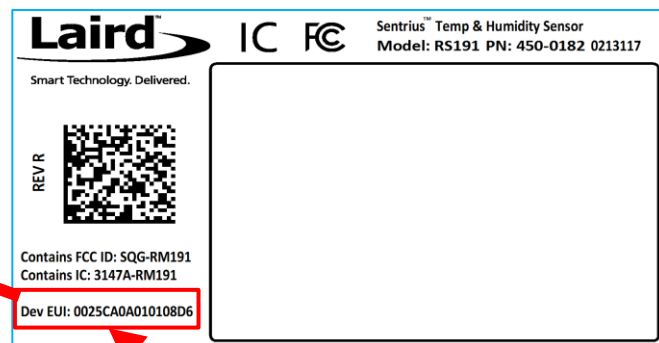


Figure 13: Device registration

3. Enter **Device EUI** from the RS191 label. Choose **Laird** for Device Type. Check the box for **Supports Senet Packets**. Then, click **Register New Device**.

Note: The Sensor Dev EUI can be found on the bottom of the sensor. This field is lower-case sensitive.



Device EUI from RS191 label

Figure 14: Register Device

4. Keep the values for **AppEUI** and **AppKey** shown on the next page. They are needed in RS1xx Configuration.

7 RS1XX CONFIGURATION

1. Download and run the **Sentrius sensor app** (available via [Apple Store](#), [Google Play](#) and [Microsoft Store](#) – only for Windows 10 PC).
2. The Sentrius app scans for Sentrius BLE devices. Click the Bluetooth icon () on the RS1xx face panel. This causes the sensor to start advertising, with **SS_T&H** as its device name.
3. The sensor's DevEUI, along with **SS_T&H**, should appear in the scan list. Connect to it.

- Click the gear icon for **LoRa Radio Settings and Info**. (Figure 15)

LoRa Radio

LoRa Radio Settings and Info



Figure 15: LoRa setting in Sentrius app

- Update **AppEUI** and **AppKey** to the values obtained from Senet. Set Channel Mask to **sub-band 1**. (Figure 16)

LoRa Configuration

DevEUI

0025CA0A010108D6

AppEUI

00250c0000010001

AppKey

???

Channel Mask

Sub-Band 1

Figure 16: LoRa Configuration page

Note: AppKey is a write only value, while DevEUI and AppEUI are read/write values. For that reason, AppKey does not appear here but it can be written.

8 SETTING UP MQTT BROKER ON AMAZON AWS

8.1 Setting up AWS server

The AWS Free Tier enables you to use a certain number of instances, depending on your region, for free for 12 months. The information on how many instances can be used for free is accessible in your AWS account after login.

Note: The AWS account used in this guide was created in Ohio, USA for this guide. Different regions may have different limit on the number of free instances, according to Amazon AWS policy.

- Create an AWS account at <https://aws.amazon.com/>. You must enter a credit card number during sign-up, but it will not be charged as long as you do not exceed the [AWS Free Tier Limit](#).
- After verification, choose **Free** for a Support Plan. (Figure 17)Figure 17: AWS Support Plan

Select a Support Plan

AWS offers a selection of support plans to meet your needs. Choose the support plan that best aligns with your AWS usage. [Learn more](#)

Basic Plan	Developer Plan	Business Plan
Free	From \$29/month	From \$100/month
<ul style="list-style-type: none"> Included with all accounts 24/7 self-service access to forums and resources Best practice checks to help improve security and performance Access to health status and notifications 	<ul style="list-style-type: none"> For early adoption, testing and development Email access to AWS Support during business hours 1 primary contact can open an unlimited number of support cases 12-hour response time for nonproduction systems 	<ul style="list-style-type: none"> For production workloads & business-critical dependencies 24/7 chat, phone, and email access to AWS Support Unlimited contacts can open an unlimited number of support cases 1-hour response time for production systems

Need Enterprise level support?
Contact your account manager for additional information on running business and mission critical-workloads on AWS (starting at \$15,000/month). [Learn more](#)

Figure 17: AWS Support Plan

- Click **Services** > **EC2**. Note that it may take up to 24 hours to fully activate the AWS account.

The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a 'Services' dropdown menu (highlighted with a red box), and 'Resource Groups'. Below the navigation bar, the 'History' section shows 'Console Home'. A search bar prompts the user to 'Find a service by name or feature (for example, EC2, S3 or VM, storage)'. The main content area displays a grid of service categories: Compute (with EC2 highlighted in a red box), Developer Tools, Machine Learning, and AR & VR. Each category lists several services, such as Lightsail, Elastic Container Service, Lambda, Batch, Elastic Beanstalk under Compute; CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, and X-Ray under Developer Tools; Amazon SageMaker, Amazon Comprehend, AWS DeepLens, Amazon Lex, Machine Learning, Amazon Polly, and Rekognition under Machine Learning; and Amazon Sumerian under AR & VR.

Figure 18: Amazon AWS Services

- Click **Launch Instance**.
- Locate **Ubuntu Server 16.04 LTS (HVM), SSD Volume Type (64 bit)** and click **Select**.

The screenshot shows the 'Launch Instance' wizard in the AWS Management Console. The 'Choose an Amazon Machine Image (AMI)' step is active. A search bar contains the text 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type'. Below the search bar, a list of AMIs is displayed, with the first entry, 'Ubuntu Server 16.04 LTS (HVM), SSD Volume Type', highlighted in a red box. To the right of this entry is a 'Select' button, also highlighted in a red box. Below the AMI list, there is a 'Free tier eligible' badge and information about the root device type (ebs) and virtualization type (hvm).

Figure 19: Ubuntu server on AWS

- The default instance should be **t2.micro**. If not, select it. Then, click **Review and Launch**.

	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
--	-----------------	--------------------------------	---	---	----------	---	-----------------	-----

Figure 20: t2.micro instance

- On the Review page, click **Edit security groups**.
- Optional: **Security group name** and **Description** can be changed to be specific for your purpose.
- Port 22 is enabled by default for SSH. Click Add Rule, and add the ports shown in Figure 21, and click **Review and Launch**.

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	Access via SSH
HTTP	TCP	80	Custom 0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop
Custom TCP R	TCP	1883	Custom 0.0.0.0/0	MQTT w/o SSL
Custom TCP R	TCP	8883	Custom 0.0.0.0/0	MQTT w/ SSL
Custom TCP R	TCP	1880	Custom 0.0.0.0/0	node-RED traffic

Add Rule

Figure 21: Configure Security Group

Note: The address 0.0.0.0/0 was used for testing purpose. For better security, you may consider adding the IP address of the computer you're using (example: <IP_ADDRESS>/32).

- Click **Launch**.
- AWS can now create a key to securely connect the AWS instance via SSH. In the pop-up window, select **Create a new key pair** and provide a **name for key pair**. **Download Key Pair** and **Launch Instances**.

Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Create a new key pair

Key pair name

Key pair name

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel Launch Instances

Figure 22: Create a new key pair

- Your AWS instance is now running.

Note: To avoid being charged, make sure no instance is running after the 12 month trial period.

8.2 Installing MQTT Broker on AWS instance

1. You must SSH into the AWS EC2 instance and install the MQTT broker. To SSH to it with the generated key pair above, complete the following:
 - a. For Linux, open a terminal and type:

```
ssh -i <keyfile> ubuntu@<instance hostname or IP address>
```
 - b. For Windows, follow steps provided in [this link](#).

Note: The IP address for your AWS instance is in **Services > EC2 > running instances > IPv4 Public IP**.

2. Once logged in, to ensure APT package manager has the latest sources, type:

```
sudo apt-get update
```

3. To retrieve and install the broker and clients, type:

```
sudo apt-get install mosquitto mosquitto-clients
```

4. To create a username and password, type:

```
sudo mosquitto_passwd -c /etc/mosquitto/passwd <user_name>.
```

When prompted, enter a strong password and confirm it.

5. Force mosquitto to use the login and password by creating a configuration file. And for extra security, disallow anonymous users by setting "allow_anonymous" to "false," as follows:

```
printf "allow_anonymous false\npassword_file\n/etc/mosquitto/passwd\nlistener 1883 0.0.0.0\nprotocol mqtt\n" >\n/tmp/listeners.conf
```

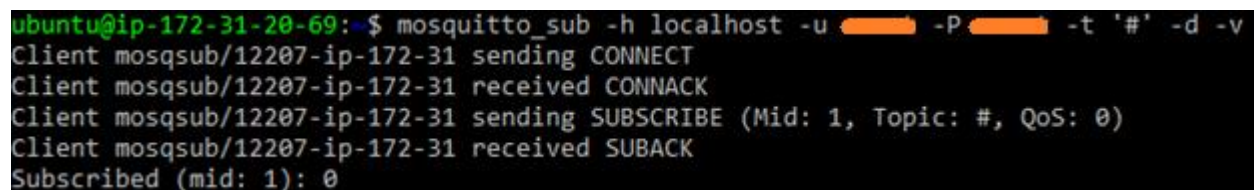
```
sudo cp /tmp/listeners.conf /etc/mosquitto/conf.d/
```

6. Mosquitto must be restarted to apply changes. To restart, type:

```
sudo systemctl restart mosquitto
```

7. Test for connecting to the MQTT broker. Type:

```
mosquitto_sub -h localhost -u senet -P <PASSWORD> -t '#' -d -v
```



The screenshot shows a terminal window with the command `mosquitto_sub -h localhost -u senet -P [redacted] -t '#' -d -v` being executed. The output shows the client connecting, receiving a CONNACK, sending a SUBSCRIBE, receiving a SUBACK, and finally being subscribed to the topic '#'. The IP address of the instance is visible as 172-31-20-69.

Figure 23: Successfully connected to MQTT broker locally

9 MQTT INTEGRATION ON SENET

Now that all the Keys/EUIs match between RS1xx sensor and Senet, the sensor should be able to join the Senet network. Senet allows users to add a notification target to a LoRa device after it has joined (activated). The notification target should be set to be the MQTT broker created in section 8: [Setting up MQTT Broker on Amazon AWS](#).

1. Power cycle the RS1xx by taking out battery and re-inserting it.
2. Verify that the end-device is updated with Joined status on the <https://portal.senetco.io/#/home> dashboard. You may need to refresh the page.
3. Click the hamburger menu at the upper right corner of the device widget, and then click the gear icon. (Figure 24)

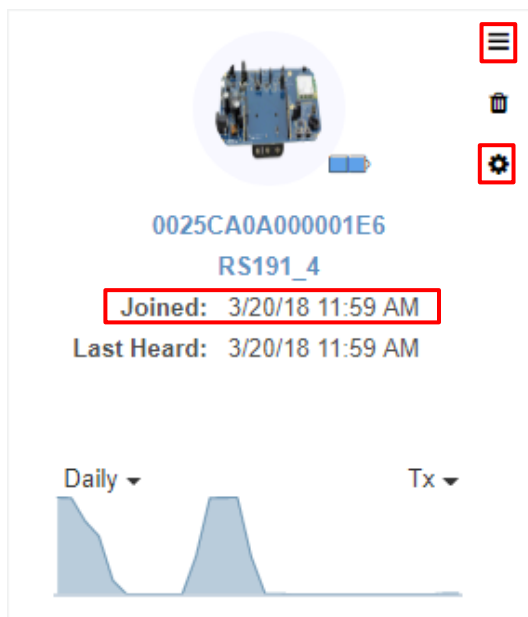


Figure 24: RS1xx activated on Senet

4. Click **Notification Target** tab and choose **MQTT** in **Forward To**. (Figure 25)

The screenshot shows the 'Edit Device' form with the 'Notification Target' tab selected. The 'Status' is 'Enabled'. The 'Forward To' dropdown is set to 'MQTT'. The 'Include RF Data' checkbox is checked, and the 'Include Duplicate Uplinks' checkbox is unchecked. The 'SSL/TLS' checkbox is unchecked. The 'Client ID' is 'RS191_4', the 'Broker Address' is a redacted field, and the 'Broker Port' is '1883'. The 'Publish Topic' and 'Subscribe Topic' are both 'application/sene'. The 'Authentication' section has 'User Name' and 'Password' fields, both of which are redacted. At the bottom, there are 'Save' and 'Cancel' buttons.

Figure 25: MQTT integration on Notification Target

5. Fill in the information for MQTT configuration. (Figure 25)
 - a. If your MQTT broker is set up with SSL support and you want to use it, check the box for SSL.

- b. Client ID: any name that preferably describes your device
- c. Broker Address: web address or IP address where your MQTT broker resides
- d. Broker Port: 8883 for SSL support and 1883 without SSL
- e. Public Topic: application/senet/node/<Client ID>/rx
- f. Subscribe Topic: application/senet/node/<Client ID>/tx
- g. User name/password: credential for MQTT broker

10 NODE-RED SETUP

node-RED is a flow-based programming environment which is used to wire devices models, APIs, and online services together. Flow-based programming is a way of describing an application's behavior as a network of *nodes*. Each node receives data, does something with that data, and then passes the data onward. This network of nodes is set up using the node-RED palette web interface.

node-RED uses *node.js* to host the flow-based programming palette and to create a user web interface. Reference <https://nodered.org/> for more information on the node-RED tool.

10.1 Install node.js Package Manager and node.js

Install the *node.js* package manager (NPM) and *node.js* locally on your PC by going to the *node.js* website: <https://nodejs.org/en/>. Both *node.js* and the NPM are installed in the same install (Figure 26). We recommend that you install the latest version.

Note: This guide is based off working versions of Node-Red: Version 0.17.5 and Node.js: Version 8.9.1

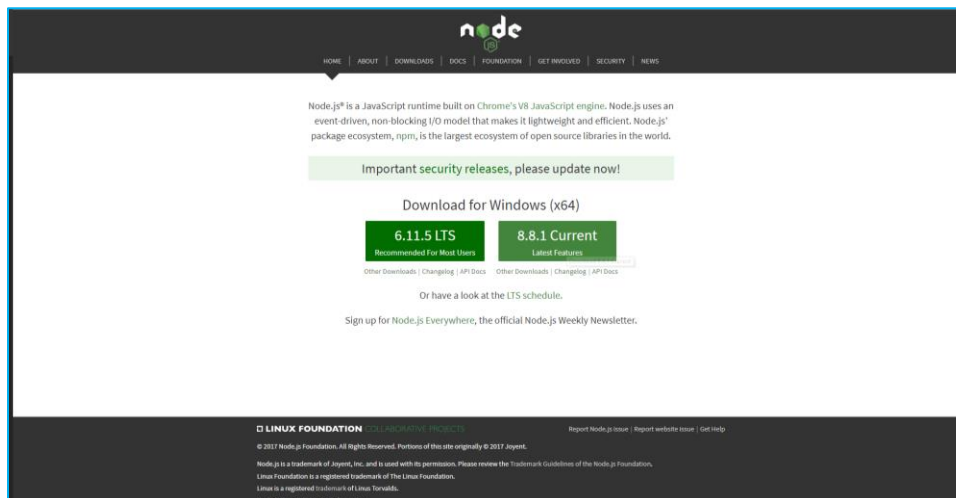


Figure 26: node.js and node.js package manager installation

Note: If there are errors encountered while running the latest version of the installer, use the **Recommended for Most Users** version of node.js.

10.2 Install JSON Files

Before running the install, you must save the JSON config and flow file in the same directory from which node-RED is run. In Windows, this is typically in your USER file location on the C: drive (e.g.

C:\Users\User_Name\nodered-demo). Placing the files in this USER file location ensures that the correct privileges are assigned to these files; otherwise there may be issues during the install of node-RED packages.

Download the JSON files off the website – <https://www.lairdtech.com/products/rs1xx-lora-sensors>

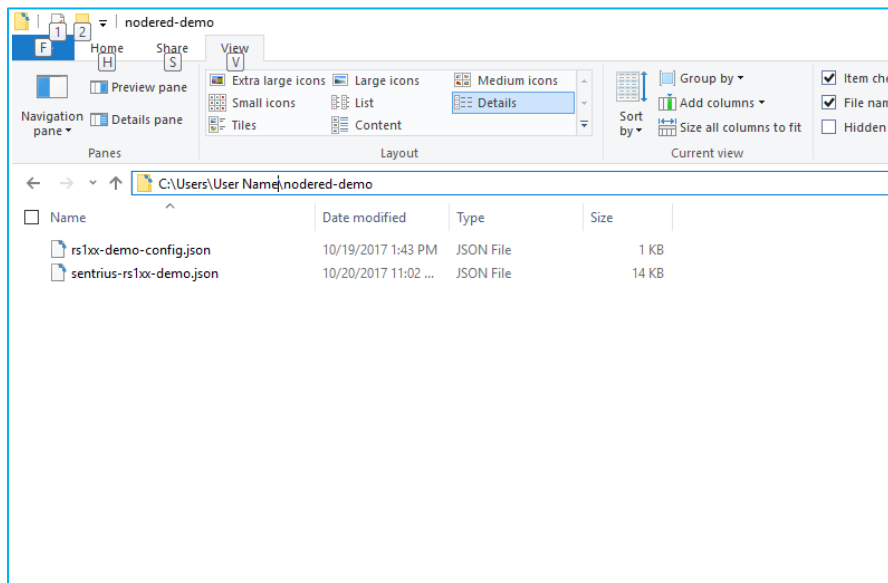


Figure 27: JSON node-RED configuration files

Note: The following files are contained in the node-RED-demo folder:

- **rs1xx-demo-config.json – Config File**
- **sentrus-rs1xx-demo.json – Flow File**

The *rs1xx-demo-config.json* file is a template for listing the devices in the Device ID dropdown menu and for changing from Celsius to Fahrenheit for the node-RED user interface. The *sentrus-rs1xx-demo.json* is a template for a node-RED flow which processes the MQTT data and sets up the user interface.

10.3 Installing node-RED

Once the nodered-demo folder is saved in the User folder within the C drive (or other folder with appropriate permissions), you can install node-RED using a terminal window. In Windows 10, press **Shift** and right-click on the mouse while in the nodered-demo folder. This brings up the PowerShell Window option. Depending on the Windows version, a Command Prompt or a PowerShell Window will appear. (Figure 28).

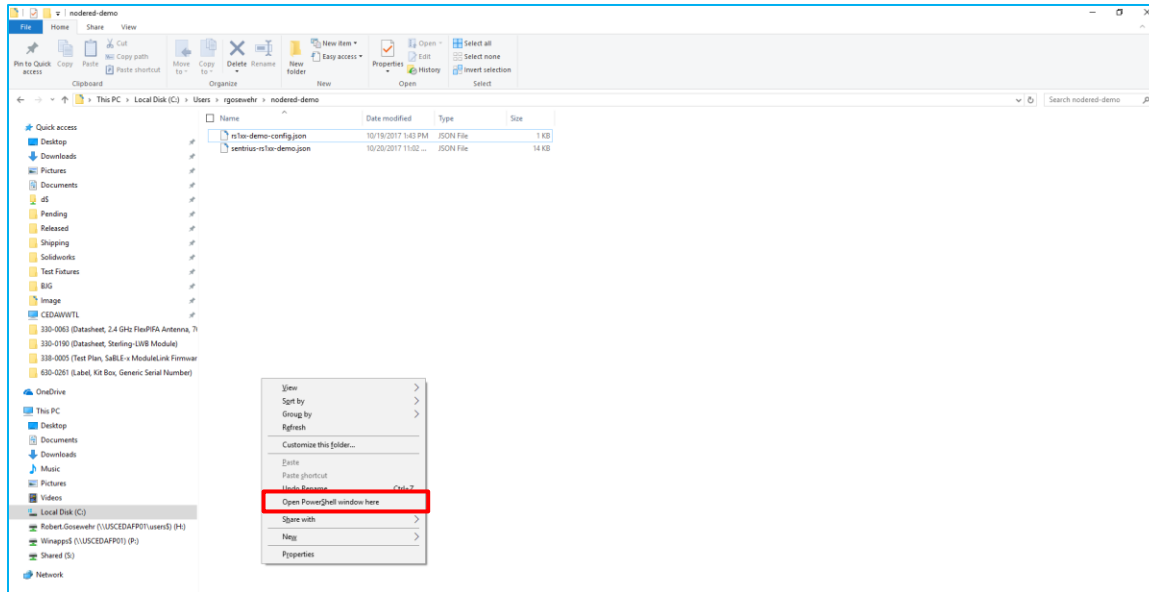


Figure 28: Opening PowerShell



Figure 29: node-Red install command prompt screen

To install node-RED using the Node.js Package Manager (NPM), enter the following:

```
npm install -g node-red
```

Reference the node-RED website for information on alternate ways to install node-RED:

<https://nodered.org/docs/getting-started/installation>.

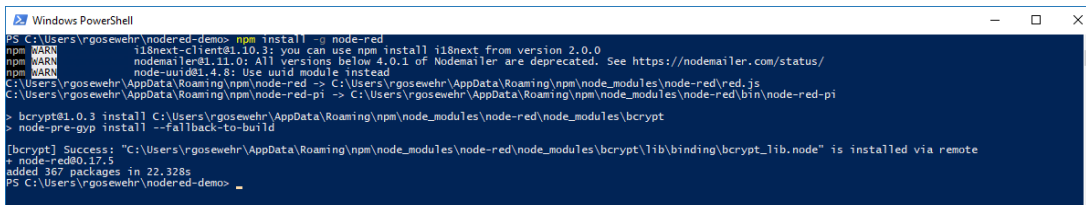


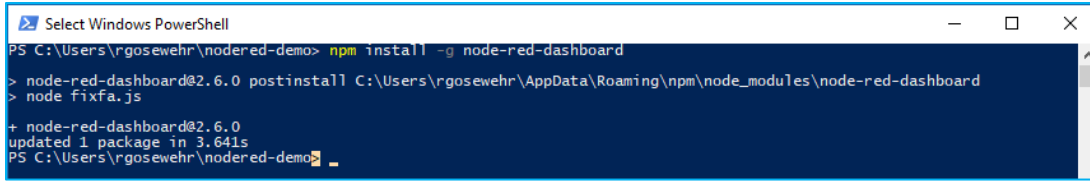
Figure 30: Successful node-Red install

node-Red is now installed on your PC.

10.4 Installing node-RED-Dashboard

To install the node-RED-dashboard, enter the following as shown in Figure 31:

```
npm install -g node-red-dashboard
```



```
Select Windows PowerShell
PS C:\Users\rgosewehr\nodered-demo> npm install -g node-red-dashboard
> node-red-dashboard@2.6.0 postinstall C:\Users\rgosewehr\AppData\Roaming\npm\node_modules\node-red-dashboard
> node fixfa.js
+ node-red-dashboard@2.6.0
  updated 1 package in 3.641s
PS C:\Users\rgosewehr\nodered-demo>
```

Figure 31: node-RED Dashboard

10.5 Updating RS1xx-Demo-Config File

To populate the Device ID drop-down menu in NodeRed with the devices registered with Senet, you must edit the *rs1xx-demo-config.json* file. This file is read by the node-RED flow during the start-up of the user web interface.

Note: JSON is an object representation in a human readable text format. Any text editor can be used to modify the file. We recommend Notepad++ (<https://notepad-plus-plus.org/>). Install the latest version.

Open *rs1xx-demo-config.json* with a text editor and follow the instruction in the below note.

Note: The content of the file is an array of device IDs plus a variable to either display the temperature readings in Celsius or Fahrenheit. The file is in JSON format and the JSON object must look like this:

```
{ "dev_ids": ["<enter a device id here>", "<enter a device id here>"],  
  "degreesFahrenheit": false }
```

For every device you want to add to the list, replace: "<enter a device id here>" with a device ID. The file must be named *rs1xx-demo-config.json*. The following is an example of a valid config file content:

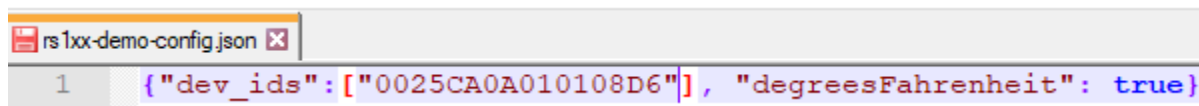
```
{ "dev_ids": [ " 0025CA0000000002", "0025CA0000000004"], "degreesFahrenheit": false }
```

Even if there is only one device, the array format (square brackets) is used.

The device ID is case sensitive and must exactly match the device ID in the Senet portal.

To change the temperature readings from Celsius to Fahrenheit, set the variable from false to true or vice versa depending on the what is desired. False is the Default Setting.

When you're done editing, save the file.



```
rs1xx-demo-config.json
1 { "dev_ids": [ "0025CA0A010108D6" ], "degreesFahrenheit": true }
```

Figure 32: *rs1xx-demo-config.json* in a text editor

The dashboard module adds a set of nodes to the node-RED palette that provides a quick and effortless way to create a live data dashboard.

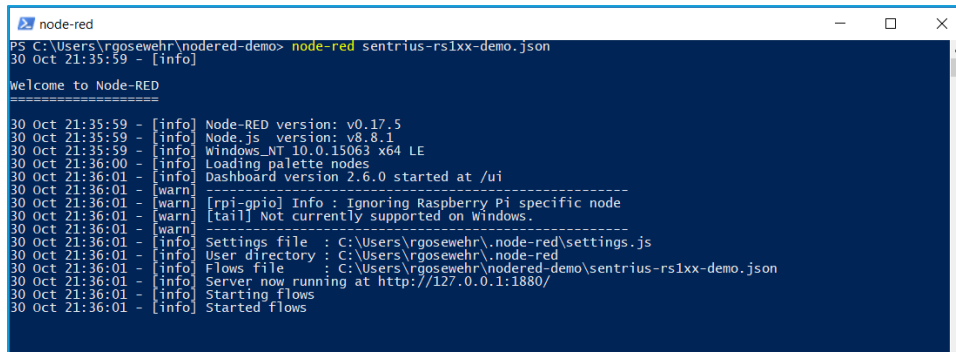
10.6 Running node-RED with Default Settings

In the prompt screen, enter the following command as shown in Figure 33:

```
node-red sentrius-rs1xx-demo.json
```

node-RED generates the dashboard based on the Flow file: **sentrius-rs1xx-demo.json**.

To close node-RED, press **CTRL-C**.



```
node-red
PS C:\Users\rgosewehr\nodered-demo> node-red sentrius-rs1xx-demo.json
30 Oct 21:35:59 - [info]
Welcome to Node-RED
=====
30 Oct 21:35:59 - [info] Node-RED version: v0.17.5
30 Oct 21:35:59 - [info] Node.js version: v8.8.1
30 Oct 21:35:59 - [info] Windows_NT 10.0.15063 x64 LE
30 Oct 21:36:00 - [info] Loading palette nodes
30 Oct 21:36:01 - [info] Dashboard version 2.6.0 started at /ui
30 Oct 21:36:01 - [warn] [rpi-gpio] Info : Ignoring Raspberry Pi specific node
30 Oct 21:36:01 - [warn] [tail] Not currently supported on Windows.
30 Oct 21:36:01 - [info] Settings file : C:\Users\rgosewehr\node-red\settings.js
30 Oct 21:36:01 - [info] User directory : C:\Users\rgosewehr\node-red
30 Oct 21:36:01 - [info] Flows file : C:\Users\rgosewehr\nodered-demo\sentrius-rs1xx-demo.json
30 Oct 21:36:01 - [info] Server now running at http://127.0.0.1:1880/
30 Oct 21:36:01 - [info] Starting flows
30 Oct 21:36:01 - [info] Started flows
```

Figure 33: node-red-sentrius-rs1xx-demo.js

Open a Chrome web browser to the address: <http://localhost:1880>. This launches the node-RED Integrated Development Environment (IDE). (Figure 34)

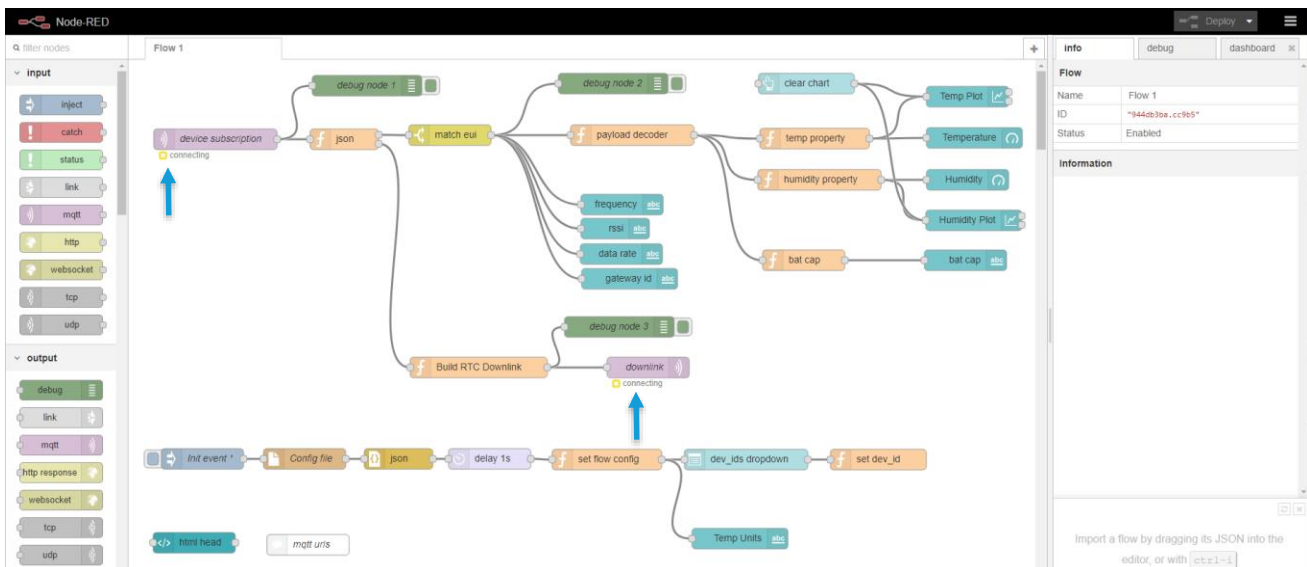


Figure 34: Default Node-RED Dashboard

10.7 Establishing MQTT Broker Connections

When you run the node-RED application for the first time, notice that the MQTT client links are not connected, as indicated by the yellow connecting or red disconnected symbol below them.

To establish an MQTT connection with the broker running on your AWS EC2 instance, you must set the MQTT Broker configuration. The Broker configuration is the information required to connect to an MQTT Broker using

the node-RED flow. The connection allows the nodes to receive sensor device's data from the Senet servers and display the information in the dashboard.

To modify the nodes, follow these steps:

1. Click the hamburger menu (☰) in the top right corner, click **Configure Nodes**. (Figure 35)

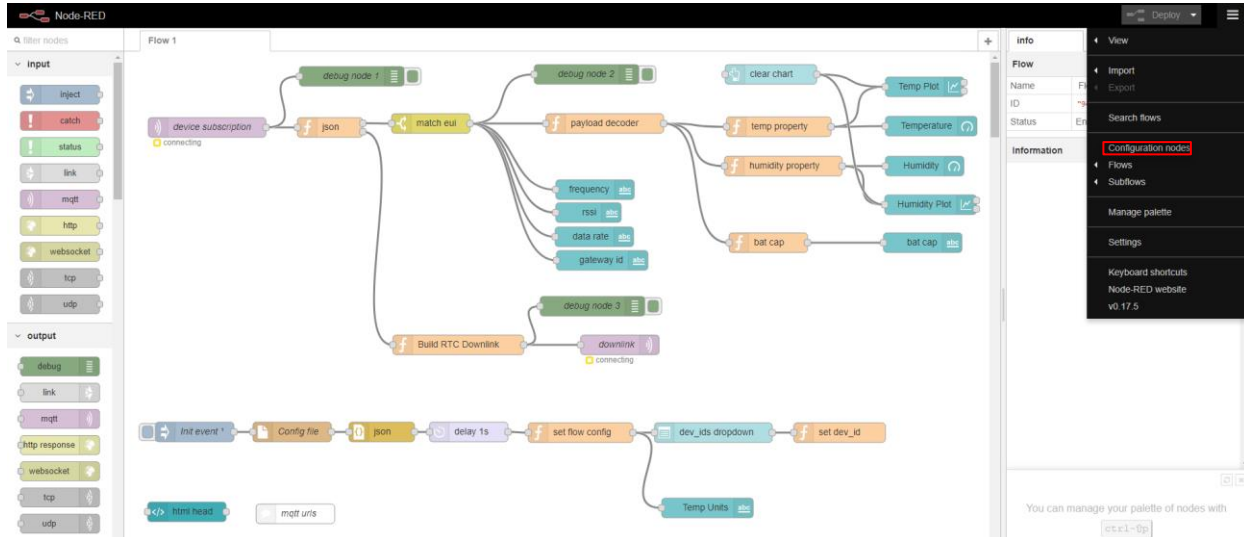


Figure 35: Configuring nodes

2. Double-click the MQTT broker configuration. This brings up the customizable settings for this node. (Figure 36) On the Connection tab, type the MQTT broker's address in Server and Port. (usually port 1883 is used for TCP and port 8883 is used for TCP/TLS)

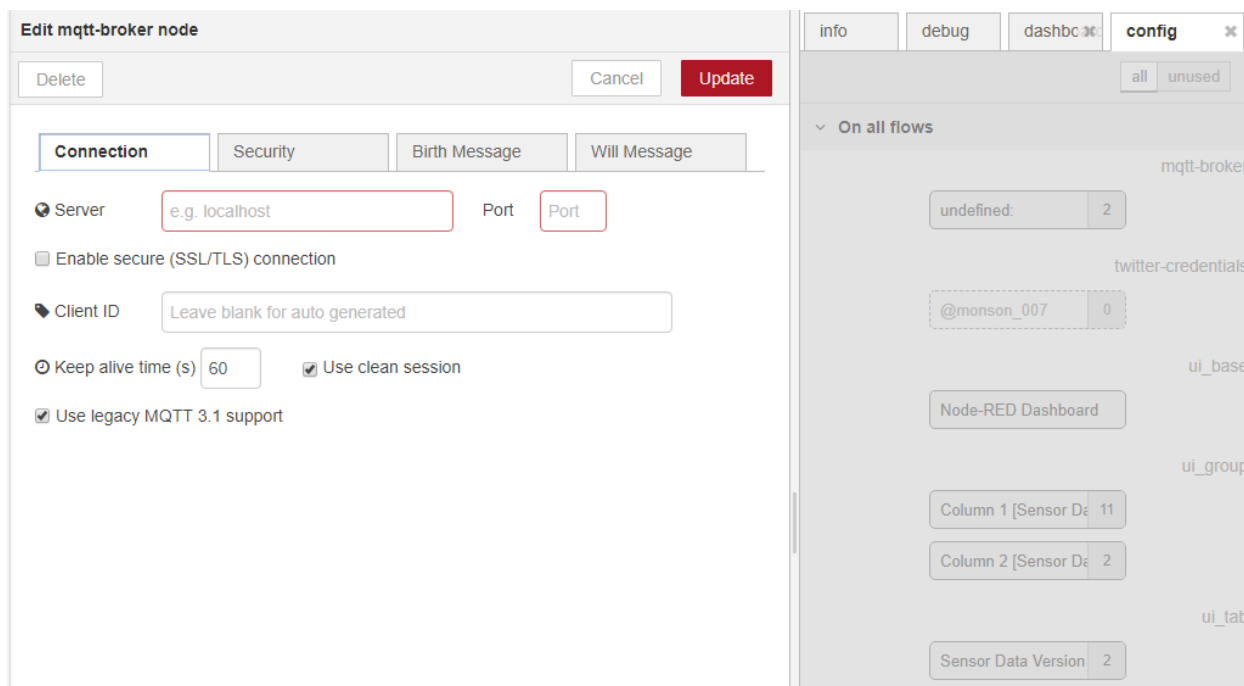



Figure 36: Configuring MQTT Broker

3. Click the **Security Tab**. Enter user name and password on the MQTT broker.

- Click **Update** to save parameters.
- Click **Deploy**  to update the node-RED application on the local node-RED server.

Note: Whenever a change is made to the flow, you must click on the **Deploy** button to save the changes to the server. If customizing the Flow, utilize the palette which contains a list of possible nodes located on the left-hand side of the screen.

Once the MQTT client configuration is set and a MQTT connection is established, the communication links display a connected status. Note that all the MQTT nodes on the flow share the same MQTT connection under-the-hood.

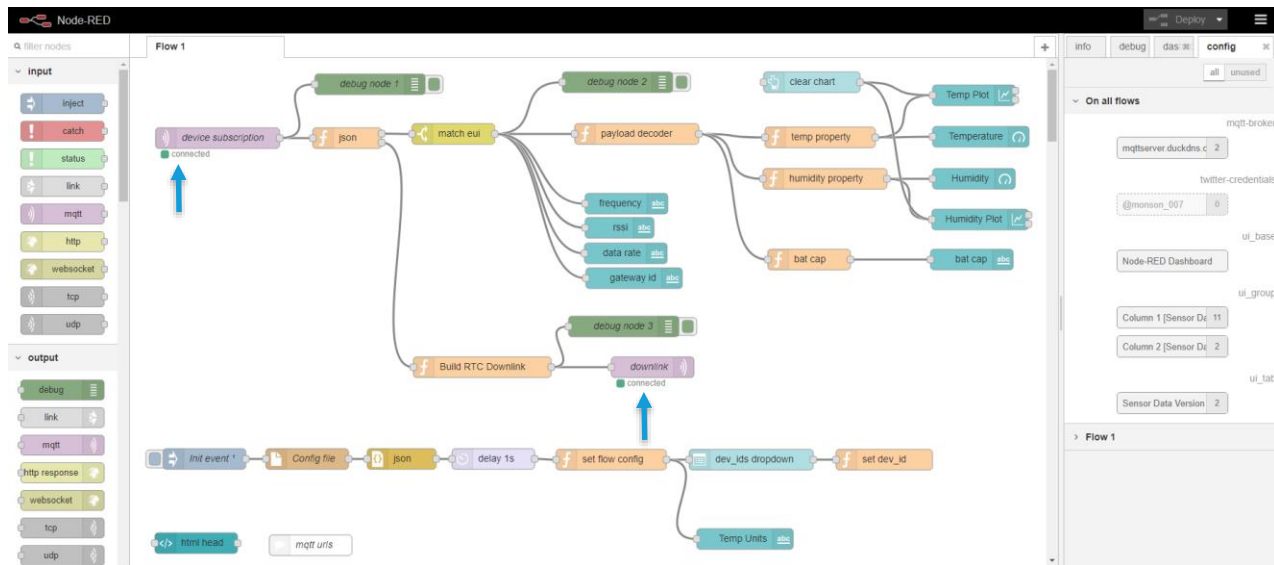


Figure 37: Good connection status

10.8 Loading node-RED User Interface

To view the node-RED user interface, open a Chrome web browser and enter the URL: <http://localhost:1880/ui>.

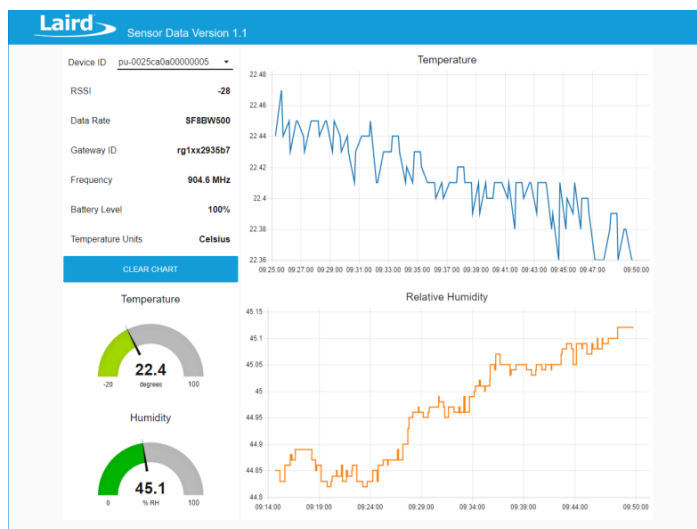


Figure 38: node-RED user interface

Note: To display the User Interface, run the node-RED flow example by entering: **node-red sentrius-rs1xx-demo.json** on the command line in a terminal window (Figure 33Error! Reference source not found.). This starts a local host web server which provides the user interface web page and establishes the MQTT connection to the Senet server.

To pick a different device in the drop-down menu, click the Device ID.

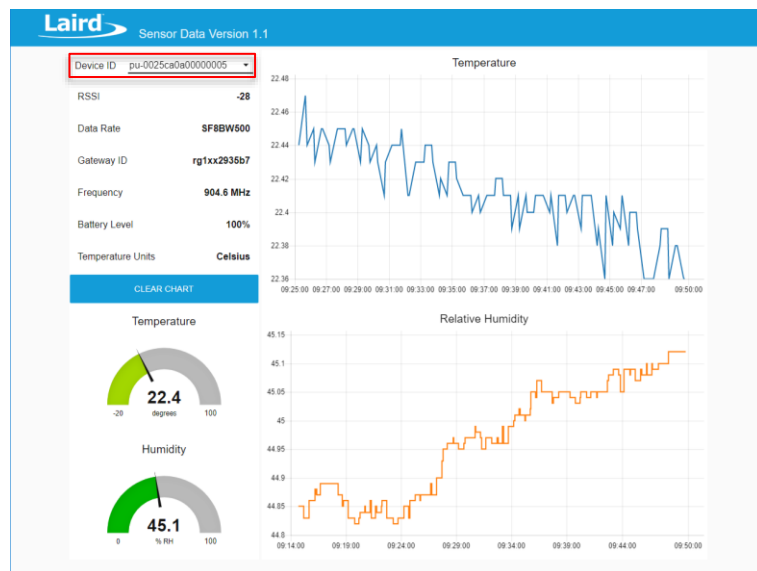


Figure 39: Device ID Drop-Down Menu

Note: Please contact a Laird sales representative if you need help setting up your application to work with the Senet servers and the customizable node-RED user interface.

11 APPENDIX

11.1 Troubleshooting Tips for Sentries Gateway Connections

11.1.1 LoRaWAN Network Ports Open?

- For the gateway to get access to the necessary LoRaWAN network server, port 1700 must be open for bi-directional UDP traffic:
collector.senetco.io: **Port 1700 UDP**
- This is the most frequently encountered problem when running this demo.

11.2 Troubleshooting Tips for Senet Connections

11.2.1 Common Issues

- Most connection problems are due to inaccuracies between what LoRa Configuration settings (Dev EUI, App EUI, and App Key) are loaded in the RS1xx device and the settings on the Senet server. Make sure the RS1xx settings are accurate and match what is displayed in the Senet network.

11.2.2 Device Network Time

- The RS1xx must be provided with network time as part of the activation process for the device to begin sending data. The node-Red application sends network time to the sensor if it is running when the sensor joins the network.
 - If the node-RED application is not running, the RS1xx device joins the network (activation), but does not begin sending data. If the sensor is activated but has not received network time, the data view (on the Senet device page) shows that the device received one or more activations but no proper uplink or downlink data.

11.2.3 Connection Issues Between Registered Devices and Senet Servers

- In the Senet dashboard, each device has status. Joined value is how long ago the sensor joined the Senet network and Last heard value is likely the last time data arrived from the sensor.

Joined: 51 minutes ago
Last Heard: 51 minutes ago

Figure 40: Device's status

- You may need to Refresh the Page to see the status change.
- Make sure the Activation Method is set to Over-the-Air-Activation (OTAA)

11.3 Troubleshooting Tips for node_RED (MQTT Broker Links) Connections

11.3.1 MQTT Network Port Open?

- For node-red to connect with the MQTT Broker (broker links in the node-RED palette), the network interface needs to open a TCP socket:
 - For using without TLS, **Port 1883 MQTT**
 - For using TLS, **Port 8883 MQTT**
- The MQTT broker connection can be debugged independently using tool such as the Mosquito Client (<https://mosquitto.org/>) or MQTT.fx (<http://mqttfx.jensd.de/>)

11.3.2 Device IDs Correct on User Web Interface?

- The available Device IDs are read from the file **rs1xx-demo-config.json** at startup. Make sure that this config file is in the same directory as the **sentrius-rs1xx-demo.json** flow file and that node-RED is started from this directory with the command : **node-red sentrius-rs1xx-demo.json** If your Device ID is not shown in the dropdown menu, double check the config file.

11.3.3 MQTT Broker Nodes Connection

- The two MQTT nodes (**Device Subscription and Downlink**) should show a connected status. These all use the same configuration node (MQTT Broker)
 - Click the hamburger menu and navigate to **Configuration Nodes > MQTT-Broker**. (Figure 36). Make sure the username and the password are typed correctly.
- MQTT troubleshooting can be done with one of the free MQTT applications such as MQTT-SPY.
- Debug nodes are shown in green on the flow. A Debug node is active when the tab on the right has a green dot in the center. The output of active debug nodes is shown in the debug window. Enable the debug window in node-RED by clicking on the debug tab in the right-hand pane of the flow editor. When active, debug node 1 shows all device data sent from the Senet server for the Application ID (set in the Configuration Node). Debug node 2 displays data.