

Configuration Guide

TTN_nodeRed Setup

Version 1.0

REVISION HISTORY

Version	Date	Notes	Contributor	Approver
1.0	05 Dec 2017	Initial Release	Robert Gosewehr	Jonathan Kaye

CONTENTS

1	Overview.....	4
2	Data Flow Architecture.....	4
3	Gateway Factory Reset.....	4
4	Creating the Things Network (TTN) Account.....	5
5	Connect the Gateway	11
6	Log into the Gateway	12
7	Adding Devices to the Gateway in TTN	14
8	node-RED Setup.....	20
8.1	Install node.js Package Manager and node.js	20
8.2	Install JSON Files	21
8.3	Installing node-RED	22
8.4	Updating RS1xx-Demo-Config File.....	23
8.5	Installing node-RED-Dashboard.....	24
8.6	Running node-RED with Default Settings	24
8.7	Establishing MQTT Broker Connections	25
8.8	Loading node-RED User Interface	28
9	Appendix.....	30
9.1	Troubleshooting Tips for Sentries Gateway Connections	30
9.2	Troubleshooting Tips for The Things Network (TTN) Connections	30
9.3	Troubleshooting Tips for node_RED (MQTT Broker Links) Connections.....	32

1 OVERVIEW

This guide provides instructions on configuring a Sentrius RG1xx gateway and Sentrius RS1xx sensor(s) using the Things Network (TTN) 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. If there are differences, they 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 using the TTN packet forwarder to the Things Network servers where the data packets are processed. They are then forwarded to the MQTT Broker (Data API). The node-RED application contains an MQTT client that receives sensor data and displays it on a browser-based user interface.

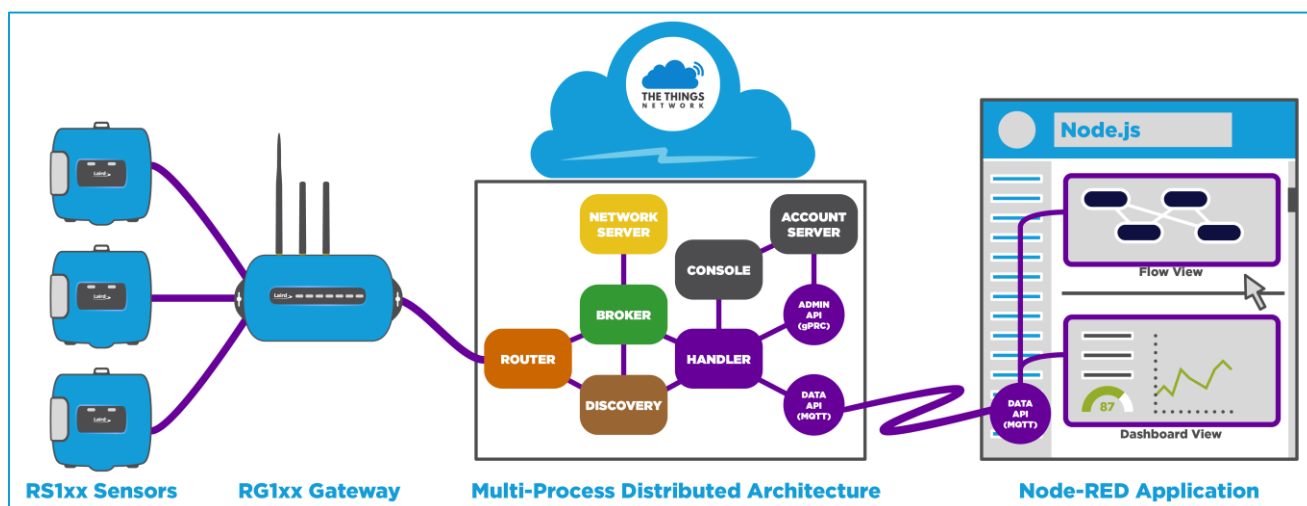


Figure 1: Data flow architecture

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 TTN 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 CREATING THE THINGS NETWORK (TTN) ACCOUNT

4.1 Account Setup

Before performing any actions, ensure that Network Ports 1900, 1901 (these are the LoRaWAN traffic ports) and Port 1883 (TTN_MQTT traffic) are open on the network. You must establish an external TCP socket and HTTP/2 connection for this to work properly. We recommend that you work with your IT to provide access for this install.

To set up an account with **The Things Network (TTN)**, follow these steps:

1. Go to <https://www.thethingsnetwork.org/> and click **Sign-Up** in the upper right of the window to start the process or, if you already have an account, **Login** (skip to step four).

Figure 2: TTN account setup

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

2. Log in to TTN after the email is verified.

Figure 3: TTN login page

3. On the main page, click your profile name and picture.
4. Click **Console** in the drop-down selection.

4.2 Registering Your Gateway

Note: Registering your gateway on the TTN network and configuring the Gateway LoRa connection on the web interface should be performed at the same time since you need the TTN-generated gateway key to finish the TTN packet forwarder setup using the gateway's web interface.

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

1. Click **Gateways** on the console screen (Figure 4).

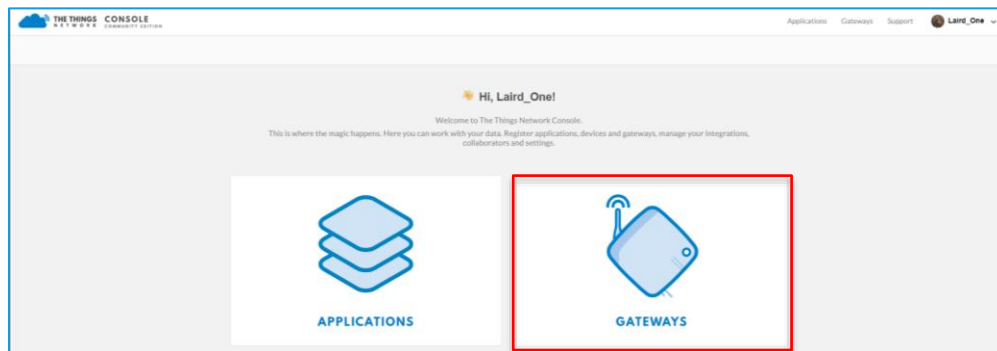


Figure 4: TTN console screen (Gateways)

2. Click **register gateway** (Figure 5).

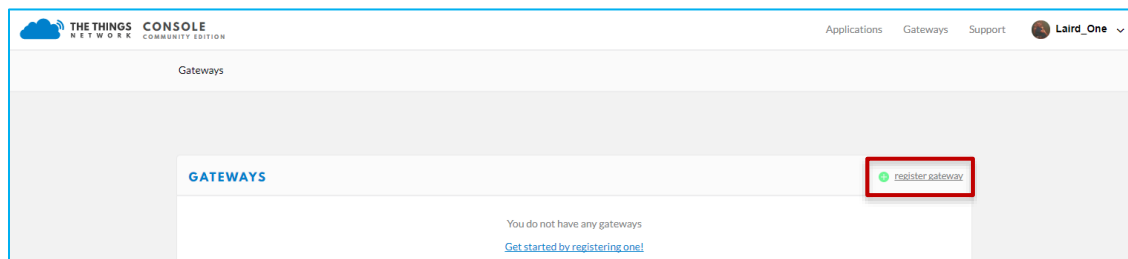


Figure 5: Click Register Gateway

3. Enter the gateway ID (Figure 6). You can find your gateway ID either on the label on the bottom of the gateway (Figure 7) or on the Sentiur RG1xx web interface (see Figure 21).

Note: If using the bottom gateway label for reference, enter rg1xx plus the last three bytes of the M2 EUI. For example: M2 EUI is C0:EE:40:FF:FF:29:37:8B, the gateway ID would be **rg1xx29378B**.

The screenshot shows the 'REGISTER GATEWAY' form in the TTN Console. The form includes fields for 'Gateway ID' (with a placeholder example), a radio button for 'I'm using the legacy packet forwarder', a 'Description' field, a 'Frequency Plan' dropdown (currently set to 'no selection'), a 'Router' field, and a 'Location' map. The 'Gateway ID' field is highlighted with a red border.

Figure 6: Gateway registration form

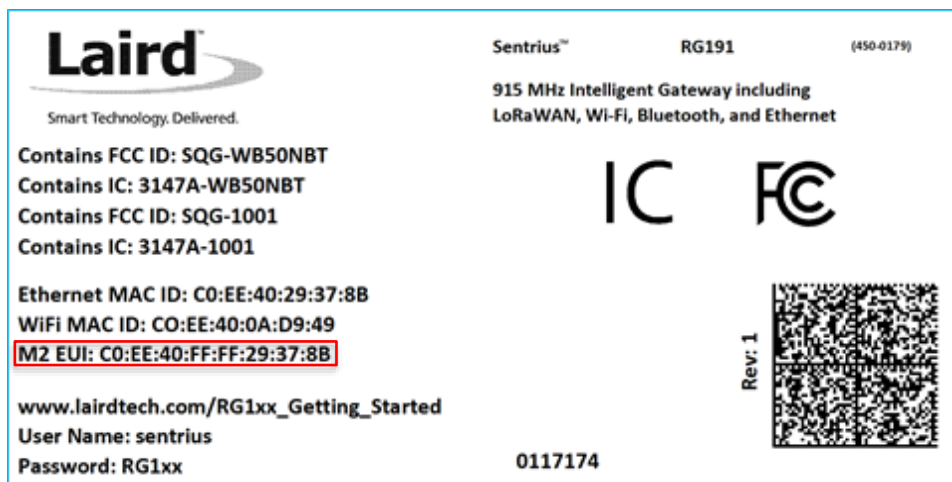


Figure 7: Gateway ID on a bottom label

Note: The Gateway ID field (text string format) must be lower case and has limitations on what characters can be used. The Gateway ID must be unique among all gateways on the TTN server, not just this account.

The screenshot shows the 'Gateway ID' field in the registration form. It includes a red border around the input field and a red warning icon with the text: 'Gateway ID can only exist of lowercase alphanumeric characters, nonconsecutive - and _ and it cannot start or end with a - or a _'.

Figure 8: Gateway ID restrictions

Note: Because we use the TTN Forwarder, do *not* mark the Legacy Packet Forwarder.

☐ **I'm using the legacy packet forwarder**
Select this if you are using the legacy [Semtech packet forwarder](#).

Figure 9: Packet forwarder selection

4. Enter a unique description for the gateway ([Figure 10](#)).

Description
A human-readable description of the gateway

Figure 10: Description

5. Select the appropriate frequency plan from the drop-down menu ([Figure 11](#)). This is based on the region in which the gateway is operating.

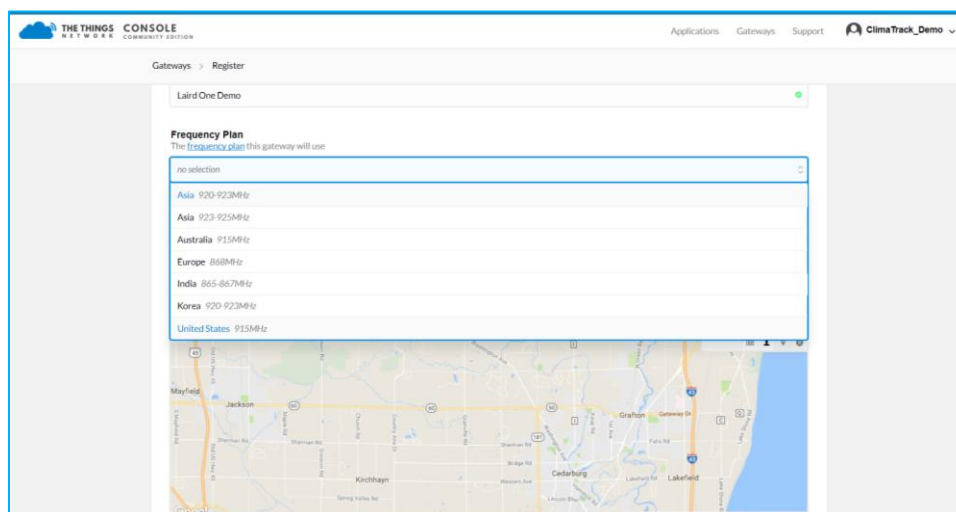


Figure 11: Frequency plan selection

6. Select the appropriate router from the drop-down menu (Figure 12).

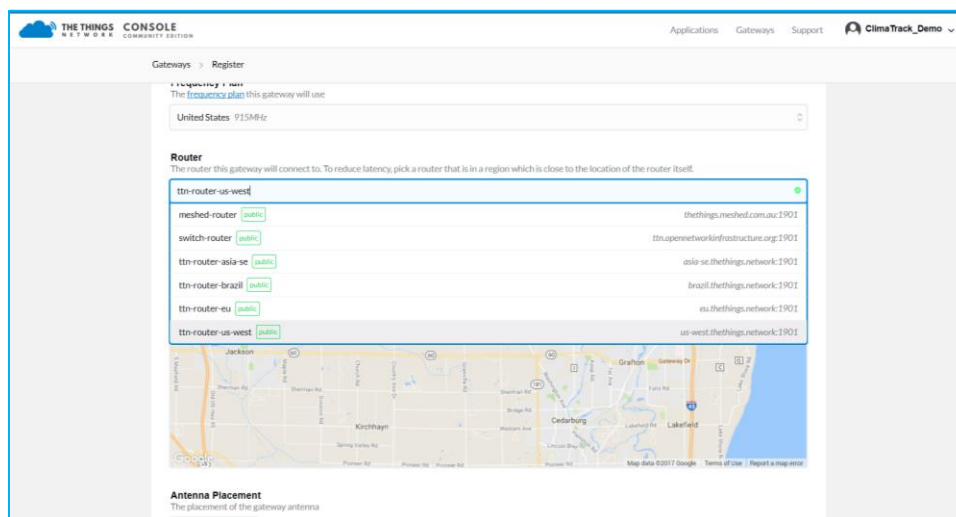


Figure 12: Router selection

The router selection is based on the physical location of the gateway. Selecting the correct router reduces latency.

7. (Optional) Set the location of the gateway.

Note: This gateway location step is optional; however, setting the location helps determine which gateway was used to forward the LoRaWAN packets if the sensors are visible to multiple gateways.

- a. To set a location in the map area, double-click the desired location in the map area to place the marker (Figure 13).

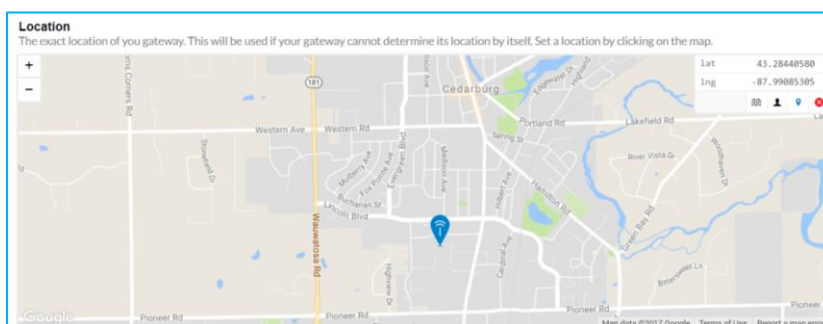


Figure 13: Marker location

- b. Click on the location where the gateway antenna will be placed.
- c. Register the gateway with TTN.
- d. Once the gateway is registered with TTN, go to the Gateway Overview page. There is a Gateway Key assigned to the gateway that needs to be entered in the Gateway Key field located on the gateway's web interface (see Figure 22 for additional information) when setting up the TTN Forwarder.

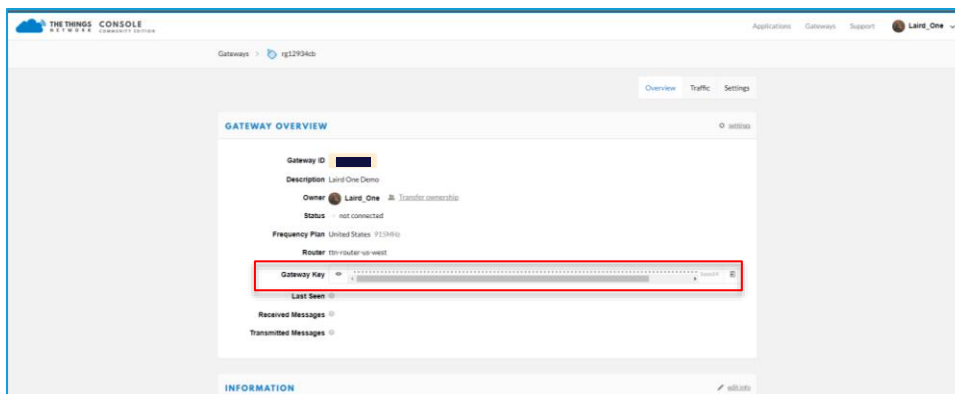


Figure 14: Gateway key

Note: For the TTN servers to authorize the gateway and to establish a connection, the gateway ID and gateway key in the TTN account must be copied exactly into the gateway's web configuration page.

We recommend that you turn off the automatic updates in the Gateway settings (Figure 15).

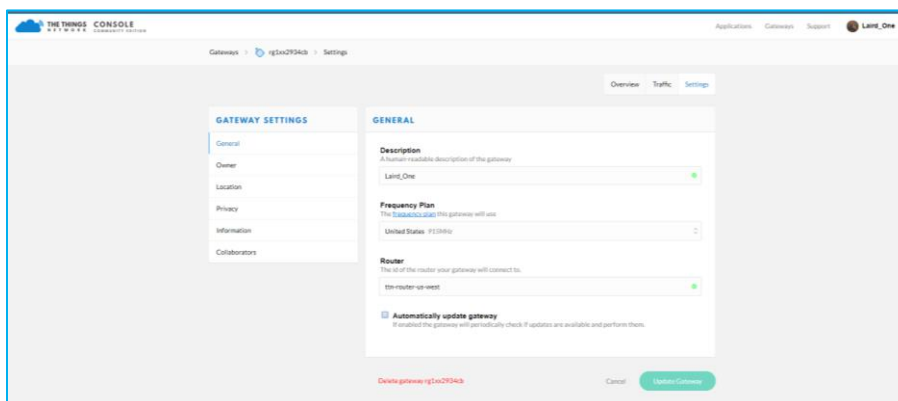


Figure 15: Automatic updates off

5 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 16](#)).
3. Connect the gateway to your router (#3 in [Figure 16](#)) using the Ethernet cable (#1 in [Figure 16](#)).

Your gateway is now connected and ready.

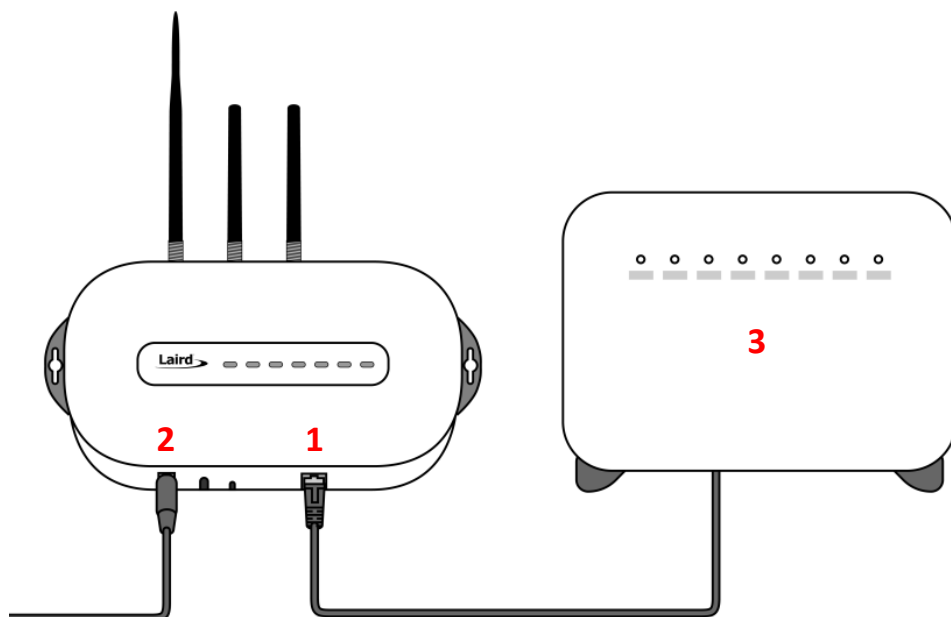


Figure 16: Connecting the gateway

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.

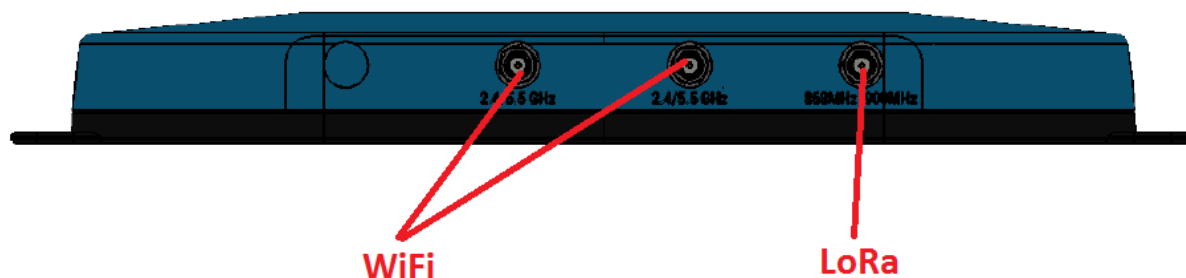


Figure 17: Antenna configuration

6 LOG INTO THE GATEWAY

6.1 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.

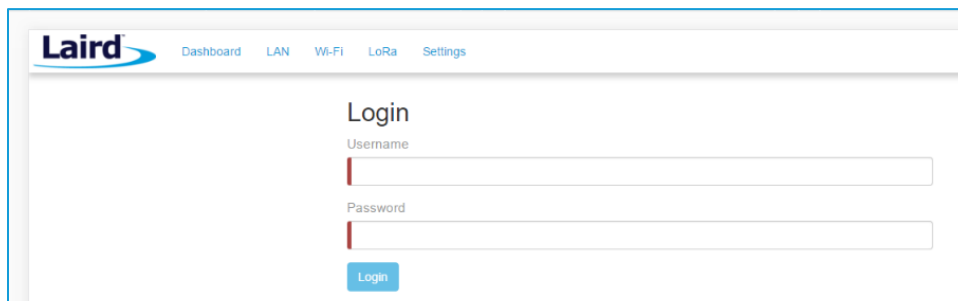


Figure 18: Gateway interface login screen

Note: For this guide, we assume the LAN and 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.

6.2 Updating Gateway Firmware

To configure the TTN forwarder, the RG1xx gateway must be updated with the latest firmware. Update the firmware by clicking **Settings > Update Firmware**.

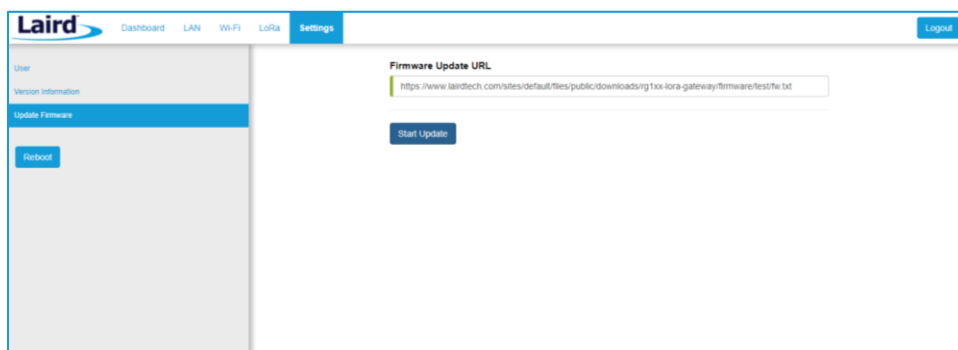


Figure 19: Updating gateway firmware

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

Note: You need GA2 FW 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.

Note: If the gateway firmware is updated after the TTN Packet Forwarder is already set up, you must set the TTN pre-sets (Figure 20). The default Forwarder setting is automatically set after an update and needs to be changed.

6.3 LoRa Packet Forwarding Set Up

Before you set up the packet forwarder, we recommend that you set the Things Network – US default settings (located in Presets accessible from the left column) for a gateway operating in the U.S. at 915 MHz. These presets configure the forwarder and the channel plan on the gateway.

The gateway connection status should change to true once the Preset is applied.

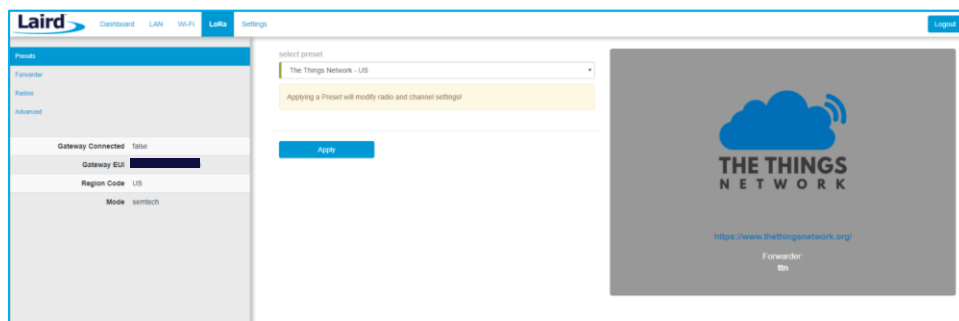


Figure 20: Setting presets

To set up the TTN packet forwarder, follow these steps:

1. Select **Forwarder** (Figure 21).
2. Select **TTN Forwarder** from the Mode drop-down menu.
3. Enter the Gateway ID which can be found by the Gateway ID in the left column or on the gateway label (see Figure 7 for additional information).

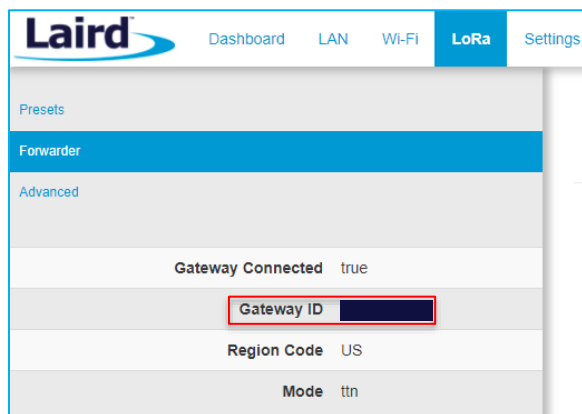


Figure 21: Gateway ID

The gateway key is generated by the TTN server during the gateway registration process. It can be copied and pasted from the Gateway Overview Page (see Figure 14 for additional information).

4. Enter the generated TTN gateway key.
5. Click **Update**. A confirmation note appears in the left column (Figure 22).

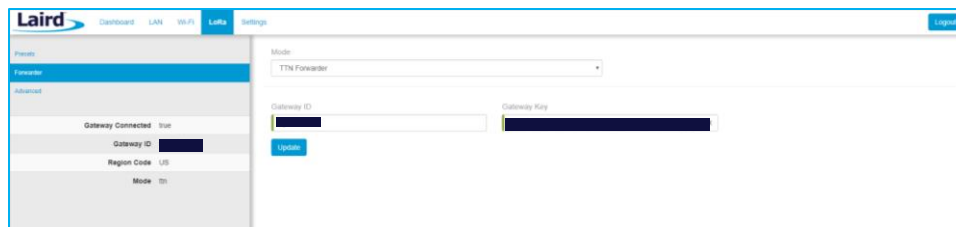


Figure 22: Gateway key entry

- Click the Dashboard to verify your connection status (Figure 23).

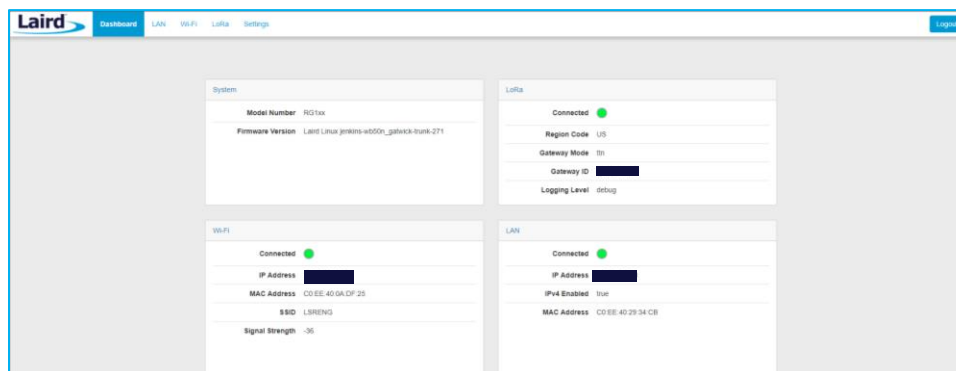


Figure 23: Gateway connection status

7 ADDING DEVICES TO THE GATEWAY IN TTN

7.1 Creating an Application

Now that the gateway is connected to the TTN network and gateway web interface, the next step is to add an application. The TTN server uses applications to create groups of devices.

Gateways are associated with user accounts but not applications. All gateways connected to TTN servers forward all LoRaWAN data traffic to the TTN message router. The TTN network server filters LoRa traffic by application ID so that data is routed to the correct user/application and users are only able to access data from devices registered to their account.

To add an application, follow these steps:

- Click **Applications** located on the Console page (Figure 24).

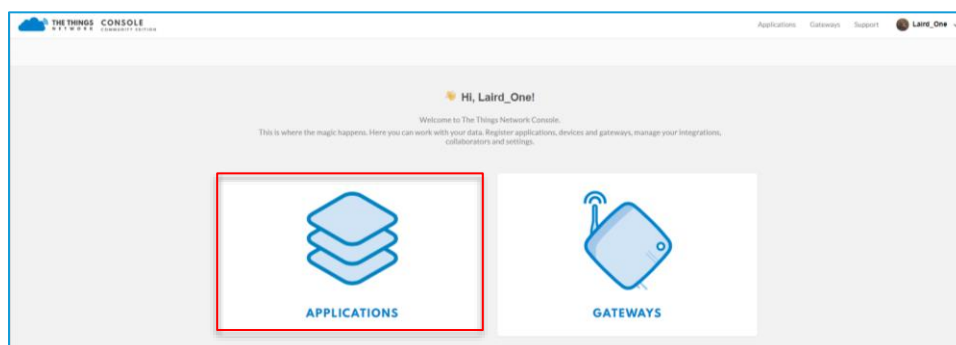


Figure 24: TTN console screen (Applications)

2. Click **add application** (Figure 25).

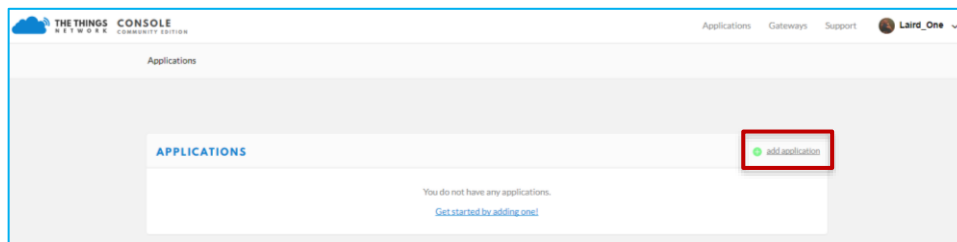


Figure 25: Click add application

The screenshot shows the 'ADD APPLICATION' form. It has four main sections: 'Application ID' with a text input field and a description 'The unique identifier of your application on the network'; 'Description' with a text input field containing 'Eg. My sensor network application' and a green checkmark; 'Application EUI' with a text input field containing 'EUI issued by The Things Network' and a description 'An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.'; and 'Handler registration' with a dropdown menu showing 'ttn-handler-eu' and a green checkmark. At the bottom right, there are 'Cancel' and 'Add application' buttons.

Figure 26: Application form

3. Enter an application ID and description. Note the application ID; it is used to set up the node-RED interface (Figure 48).

Note: The Application ID field (text string format) has limitations on what characters can be used and can only be named once.



Figure 27: Application ID restrictions

Note: The Application EUI field is auto-generated by the TTN network. This is required when setting up Laird RS1xx devices that are associated to this application ID (Figure 28).

DEVICE OVERVIEW

Application ID **laird-one-demo-app**

Device ID dev-0025ca0a0000000f

Activation Method **OTAA**

Device EUI <> 00 25 CA 0A 00 00 00 0F

Application EUI <> 70 B3 D5 7E D0 00 7D 81

App Key <>

Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0

Figure 28: Device overview screen

4. Enter the Handler Registration. This is the same as the router selection in the previous Gateway registration (see Figure 12 for more information).

7.2 Adding a Device to the Application

To add a RS1xx device to the Application ID recently established, follow these steps:

1. Click **Register Device** under Devices in the application overview page (Figure 29).

Overview **Devices** Payload Formats Integrations Data Settings

REGISTER DEVICE [bulk import devices](#)

Device ID
This is the unique identifier for the device in this app. The device ID will be immutable.

Device EUI
The device EUI is the unique identifier for this device on the network. You can change the EUI later.

App Key
The App Key will be used to secure the communication between you device and the network.

App EUI
70 B3 D5 7E D0 00 7D 81

Cancel Register

Figure 29: Register device form

2. Enter the **Device ID**. This ID must be unique on the user's account.

We recommend using the convention dev (for device), followed by the Sentries Sensor Dev EUI. For instance, if the Sentries Sensor device has a Dev EUI of *0025ca000000000f* then the Device ID is *dev-0025ca000000000f*.

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

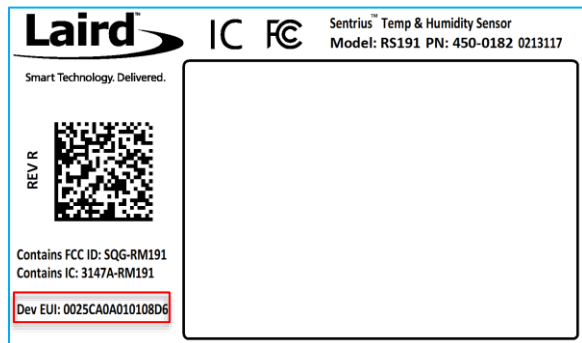


Figure 30: Sensor Dev EUI

3. Enter the Device EUI. This is the Sensor Dev EUI on the label. To ensure uniqueness, the sensors extended MAC address is used. .

The App key field is auto-generated by the TTN network.

The App EUI field was previously auto-generated by the TTN network when the application ID was created (Figure 26). The application EUI is associated with the application ID and used by the TTN servers to associate the device with the application ID.

4. Register the device.

7.3 Updating Device Settings (Dev EUI_App EUI_App Key)

Each registered device is associated with an application EUI and application key pair. The device must be updated with the newly-generated app EUI and app key. The application EUI is generated for an application ID and is used to route traffic. The application key is unique per device and is used during activation to generate the LoRa session keys. You may have multiple applications with devices registered to each application. Applications are defined by the application EUI/ID.

The application EUI and application key for a device can be found in the list of registered devices on the TTN Applications page.

To update the settings for the registered device, follow these steps:

1. Click the registered device on the Applications page (Figure 31).

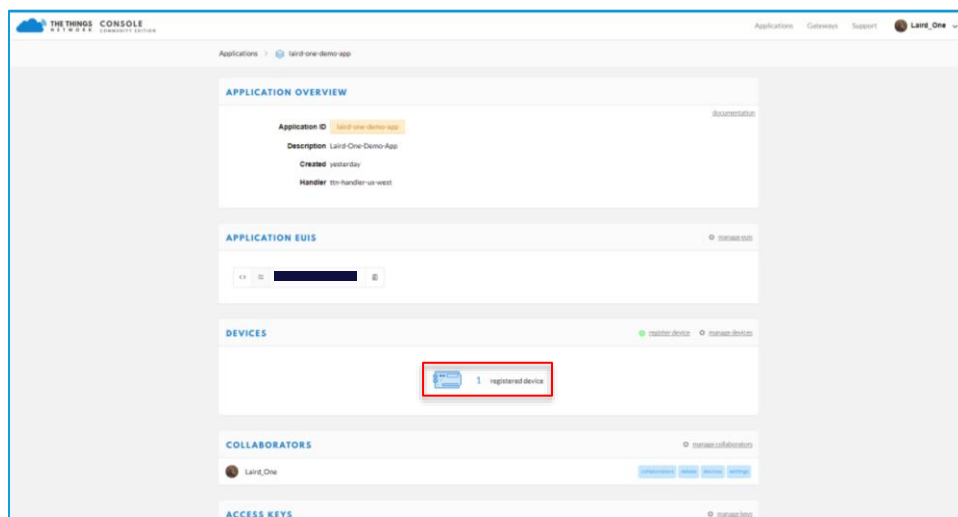


Figure 31: Registered device

2. Click the device to be updated (Figure 32).

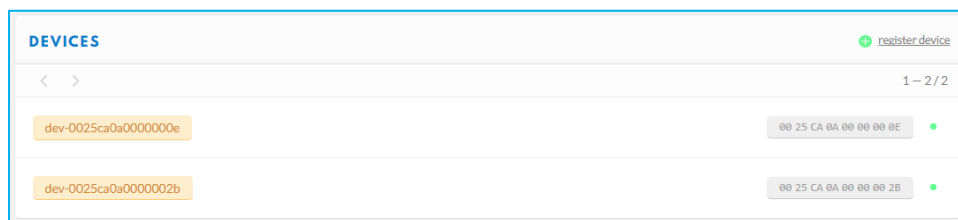


Figure 32: Device list

The screenshot shows the 'DEVICE OVERVIEW' section of a TTN interface. It contains the following fields and controls:

- Application ID:** laird-one-demo-app
- Device ID:** dev-0025ca0a0000000f
- Activation Method:** OTAA
- Device EUI:** A text field with a redacted value and a copy icon.
- Application EUI:** A text field with a redacted value and a copy icon, highlighted with a red box.
- App Key:** A text field with a redacted value and a copy icon, highlighted with a red box.
- Status:** never seen
- Frames up:** 0, with a link to 'reset frame counters'.
- Frames down:** 0

Below the 'DEVICE OVERVIEW' section is the 'DOWNLINK' section, which includes:

- Scheduling:** Buttons for 'replace', 'first', and 'last'.
- FPort:** A text field with the value '1'.
- Confirmed:** A checkbox.
- Payload:** Buttons for 'bytes' and 'fields', and a text field with '0 bytes'.

Figure 33: Application EUI and App Key

To update the App EUI and App key on each registered device, the Sentrius mobile phone app must be installed on an iOS or Android device and used to modify these LoRa configuration settings on the RS1xx Sensor. Reference the Mobile Application section of the *Sentrius RS1xx User Guide* for details on installing the phone application and updating these LoRa configurations. This guide is available from the [Sentrius RS1xx product page](#) of the Laird website on the Documentation tab.

Once the device is updated, it attempts to connect to the TTN server via the LoRa Gateway. When it has connected, the status on the TTN Device Overview page changes from *Never Seen* to a green connected Status. This is confirmation that the device is communicating with the TTN servers through a registered gateway.

DEVICE OVERVIEW

Application ID `laird-one-demo-app`

Device ID `dev-0025ca0a0000000f`

Activation Method `OTAA`

Device EUI `<>`

Application EUI `<>`

App Key `<>`

Device Address `<>` `26 02 2B A5`

Network Session Key `<>`

App Session Key `<>`

Status ● 3 seconds ago

Frames up 0 [reset frame counters](#)

Frames down 0

Figure 34: Device Connection Status

Note: The channel mask should be set to sub-band 2. This is set as a default value in the sensor, however if this is not set change the value.

8 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.

8.1 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 35). 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

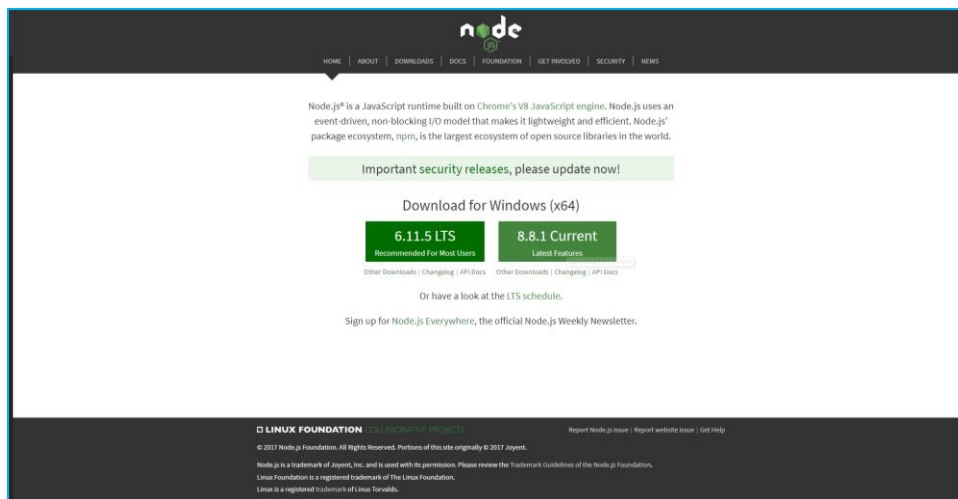


Figure 35: node.js and node.js package manager installation

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

8.2 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>

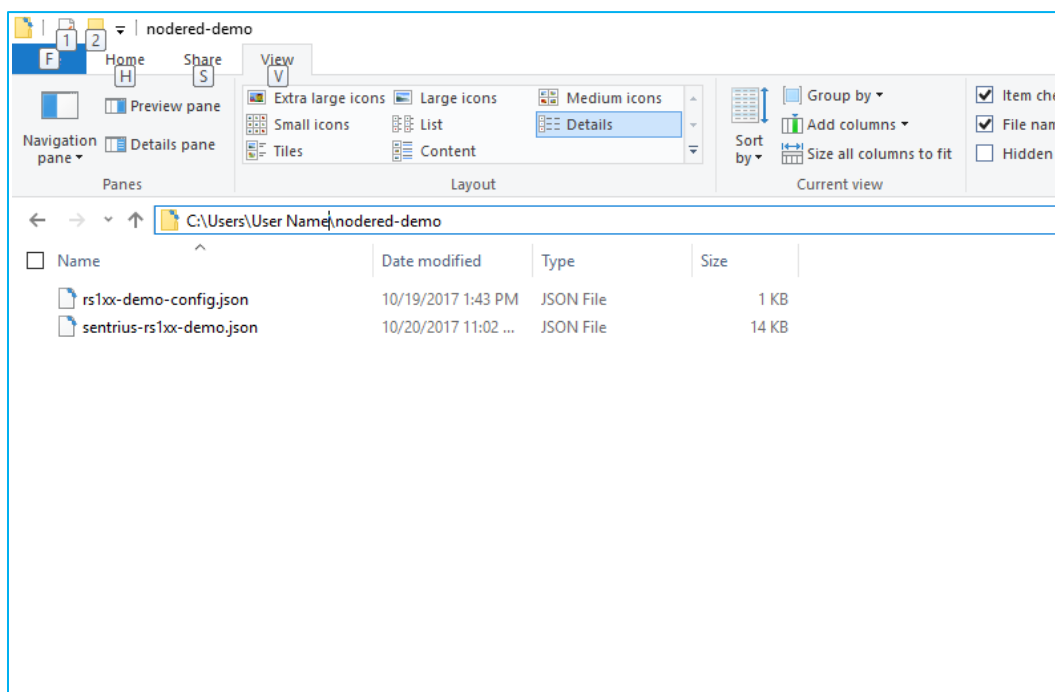


Figure 36: JSON node-RED configuration files

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

- **rs1xx-demo-config.json – Config File**
- **sentrius-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 *sentrius-rs1xx-demo.json* is a template for a node-RED flow which processes the MQTT data and sets up the user interface.

8.3 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 37).

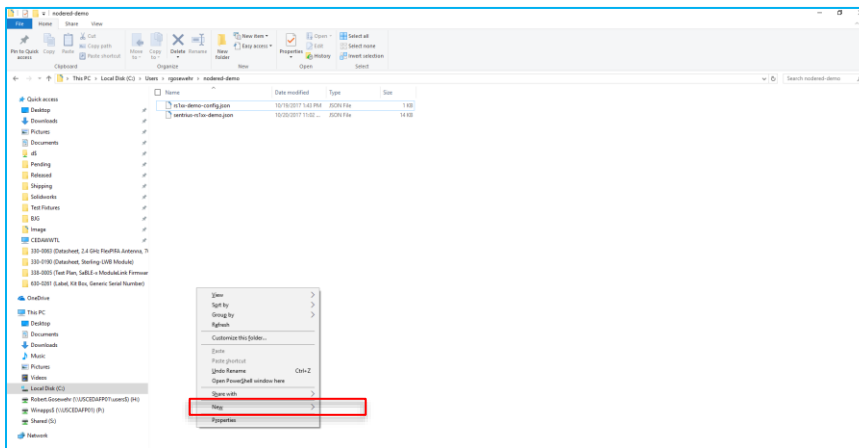


Figure 37: Opening command prompt screen



Figure 38: node-Red install command prompt screen

Type the `npm install -g node-red` command to install node-RED using the Node.js Package Manager (NPM). Reference the node-RED website for information on alternate ways to install node-RED:
<https://nodered.org/docs/getting-started/installation>.

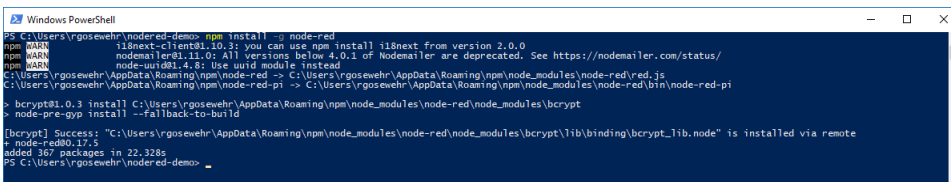


Figure 39: Successful node-Red install

node-Red is now installed on your PC.

8.4 Updating RS1xx-Demo-Config File

To populate the Device ID drop-down menu on the node-RED app user interface with the devices registered within the TTN servers, 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.

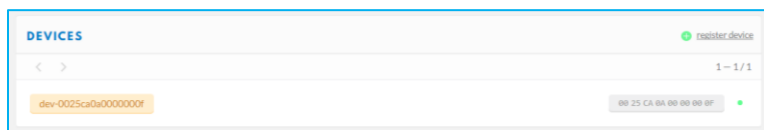


Figure 40: List of Devices from TTN

Note: Reference the Device List in the Registered Device section from the appropriate Applications ID page to see a list of available devices to list in the drop-down menu (see [Figure 32](#) for additional information).

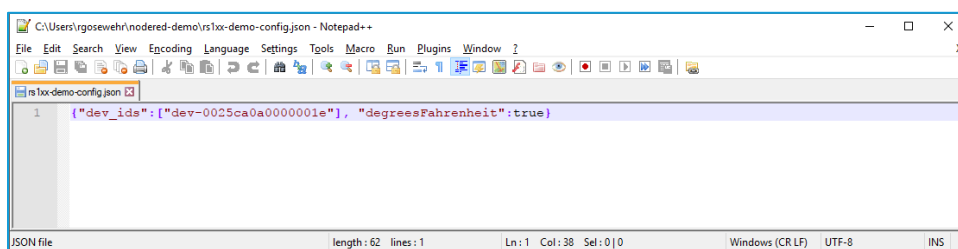


Figure 41: rs1xx-demo-config.json Editor

When you're done editing, save the file.

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:

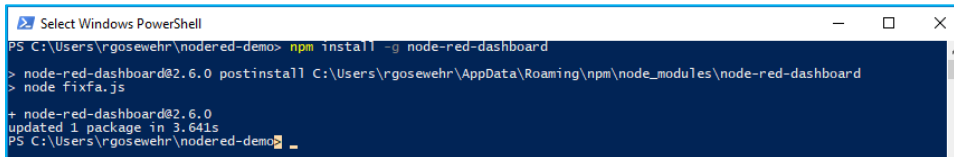
```
{"dev_ids":["dev-0025ca0000000002","dev-0025ca0000000004"], "degreesFahrenheit":false}
```

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

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.

8.5 Installing node-RED-Dashboard

To install the node-RED-dashboard, enter the command **npm install -g node-red-dashboard**.



```
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 42: node-RED Dashboard

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.

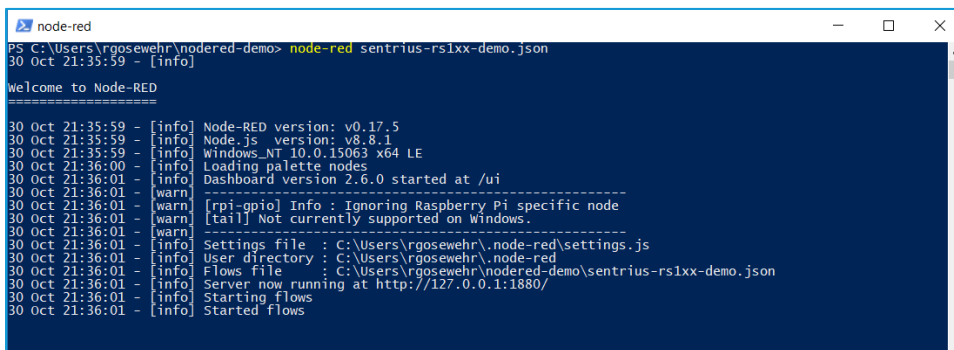
8.6 Running node-RED with Default Settings

In the prompt screen, enter the following command:

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**.



```
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]
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 43: 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).

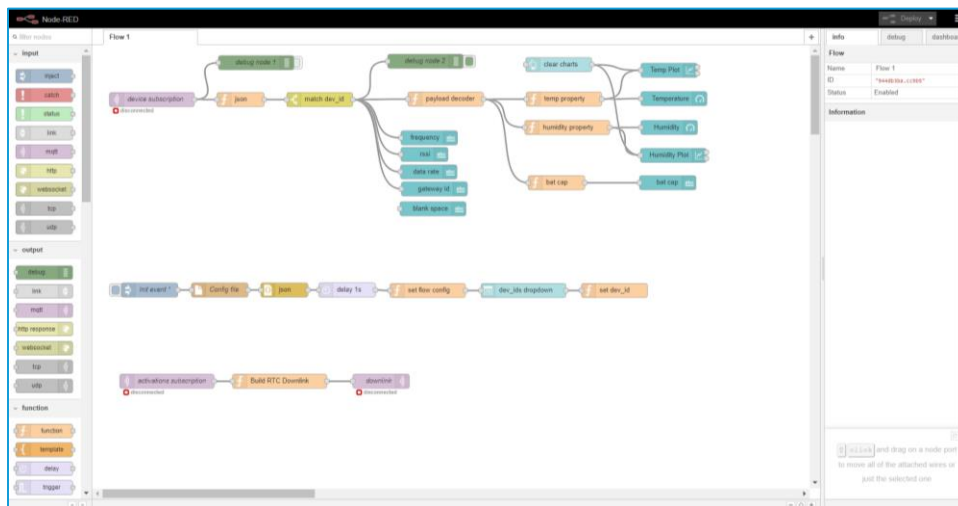


Figure 44: Default Node-RED Dashboard

8.7 Establishing MQTT Broker Connections

When you run the node-RED application for the first time, notice that the MQTT client links are disconnected, as indicated by the red disconnected symbol below them.

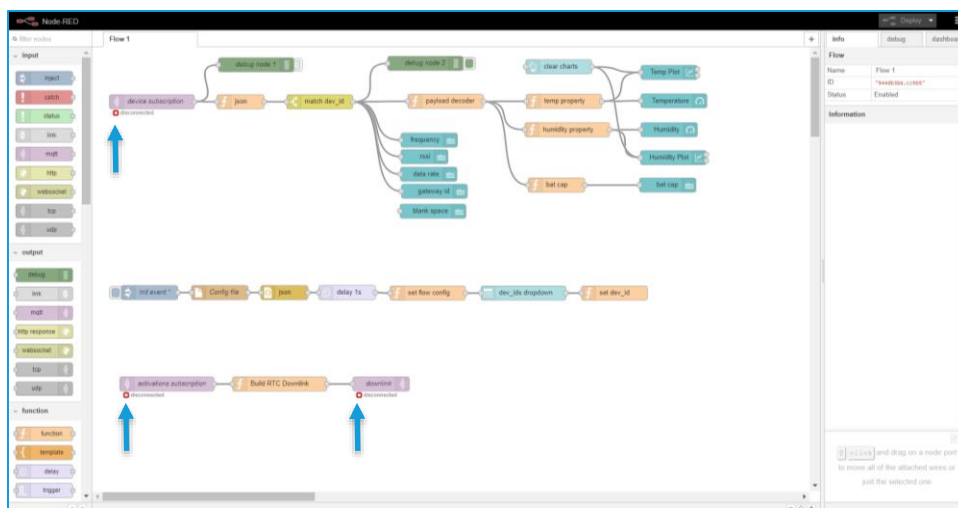



Figure 45: Broken MQTT broker links

To establish an MQTT connection with the TTN Server, 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 data from the TTN servers and display the information in the dashboard.

To modify the nodes, follow these steps:

1. From the “Hamburger Menu”  in the top right corner, click **Configure Nodes**.

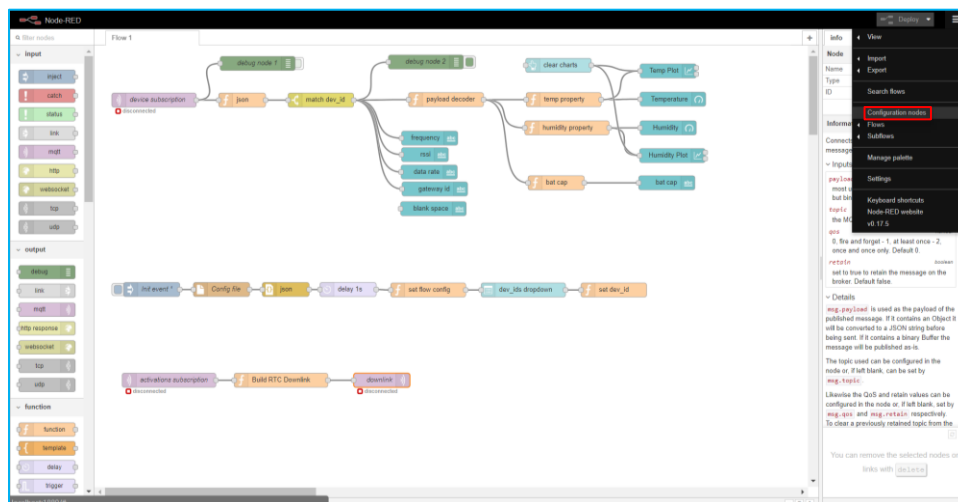


Figure 46: Configuring nodes

2. Double-click the MQTT Broker configuration. This brings up the customizable settings for this node (Figure 47).

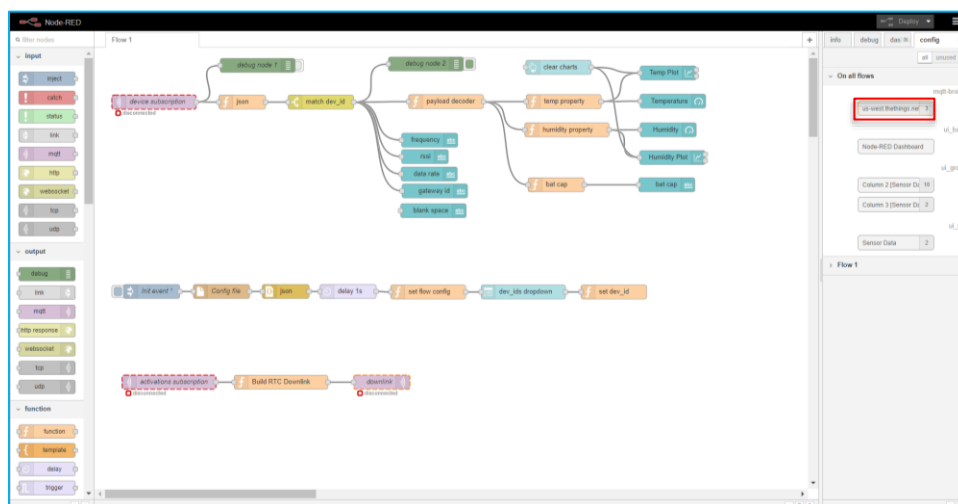


Figure 47: Configuring MQTT Broker nodes

3. Click the **Security Tab**. You can find the user name and password on the TTN network.
4. From **Applications** under the **Console** page, click the **Applications ID** under which the devices are registered. The user name is the **Application ID** and the password is the **Access Key Default Key** at the bottom of the page.
5. Click **Update** to save parameters.

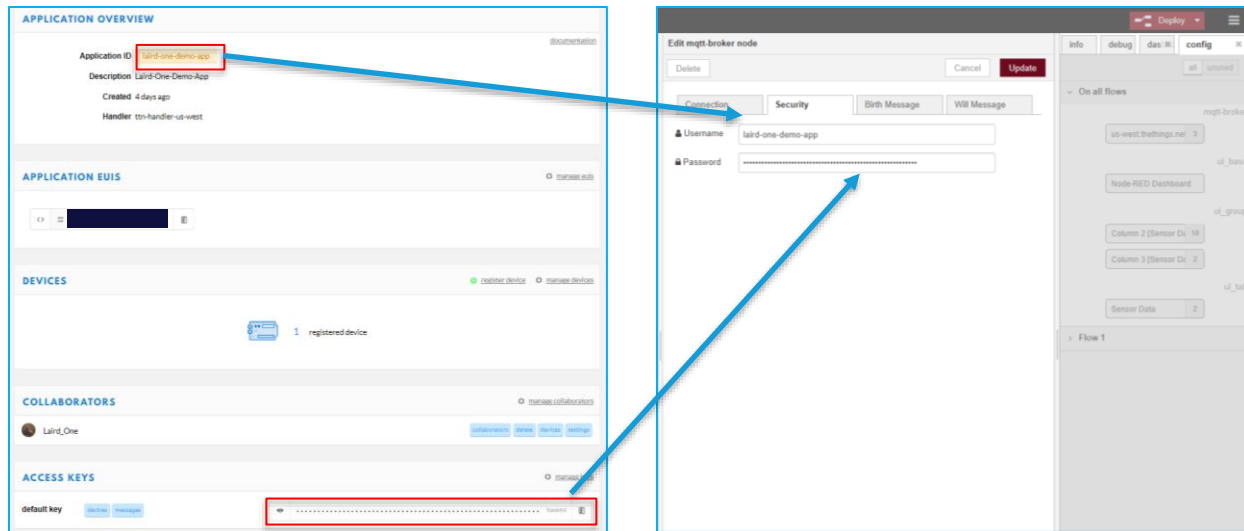



Figure 48: User name and password

Note: When entering the user name, copy the Applications ID from the TTN network *exactly*.

- 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.

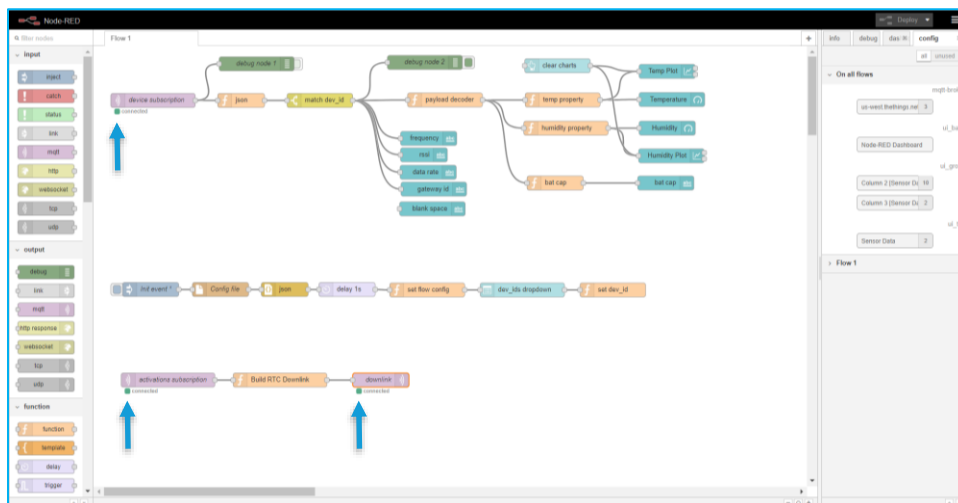


Figure 49: Good connection status

8.8 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>.

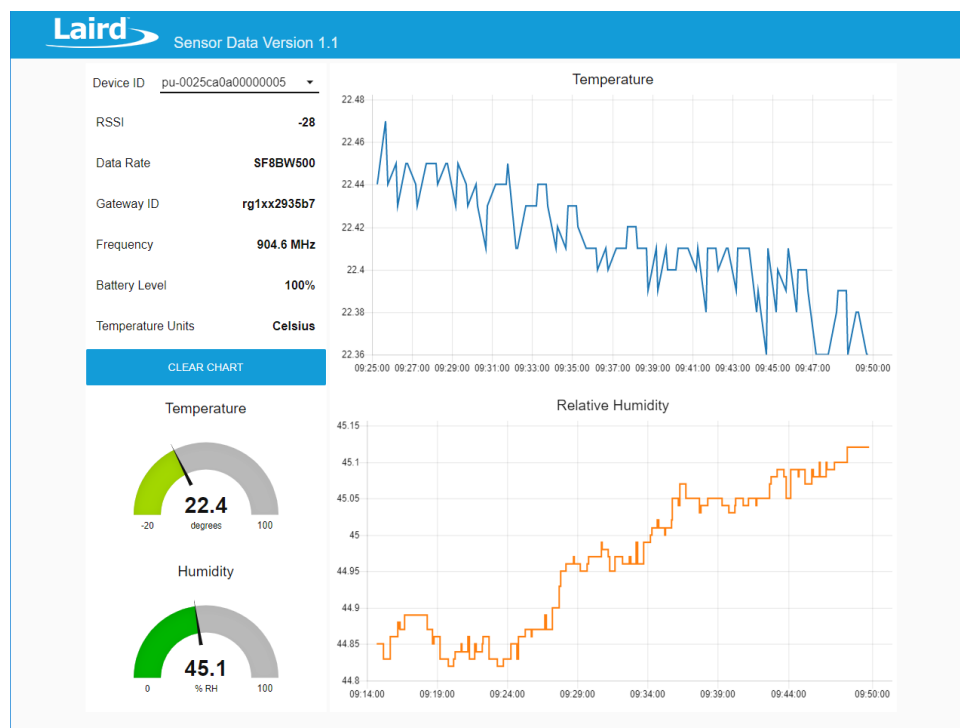


Figure 50: node-RED user interface

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 43). This starts a local host web server which provides the user interface web page and establishes the MQTT connection to the TTN server.

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

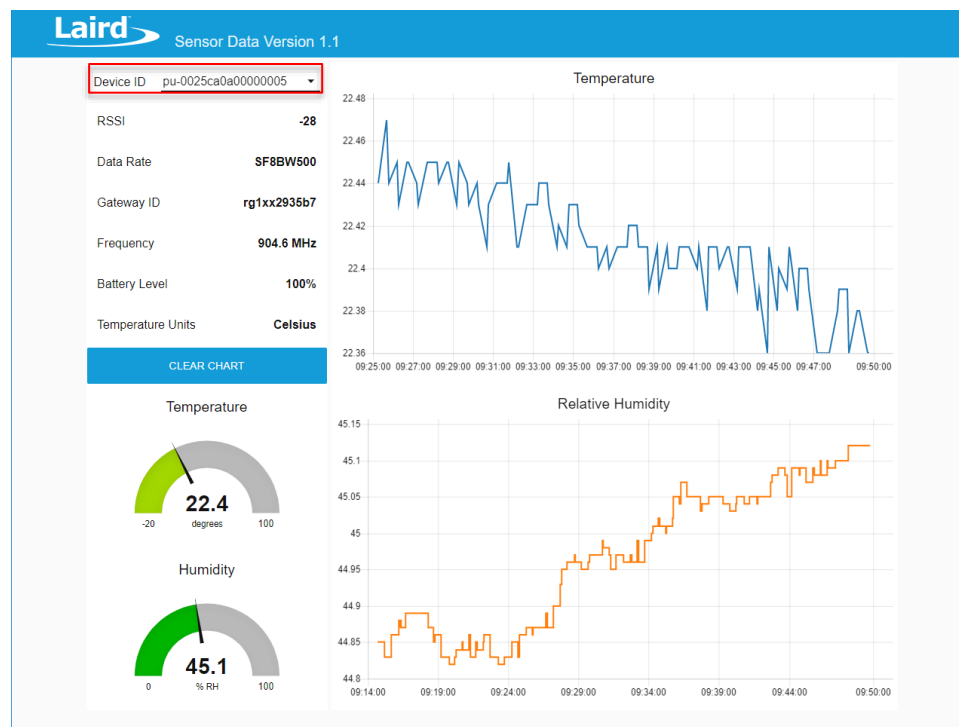


Figure 51: Device ID Drop-Down Menu

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

9 APPENDIX

Problems? Issues with connecting you gateway to the network? Learn more about viable solutions to your issues from previous encounters below in our *Troubleshooting Tips Guide*.

9.1 Troubleshooting Tips for Sentriss Gateway Connections

9.1.1 LoRaWAN Network Ports Open?

- For the gateway to get access to the necessary LoRaWAN data servers, these Network Ports must have an opened TCP socket:
 - discovery.the.thingsnetwork.org: **Port 1900 HTTP/2**
 - router.us.thethings.network: **Port 1901 HTTP/2**
- The most frequent problem encountered when running this demo is that the required IP ports are not open or have traffic restrictions in place on these ports.

9.2 Troubleshooting Tips for The Things Network (TTN) Connections

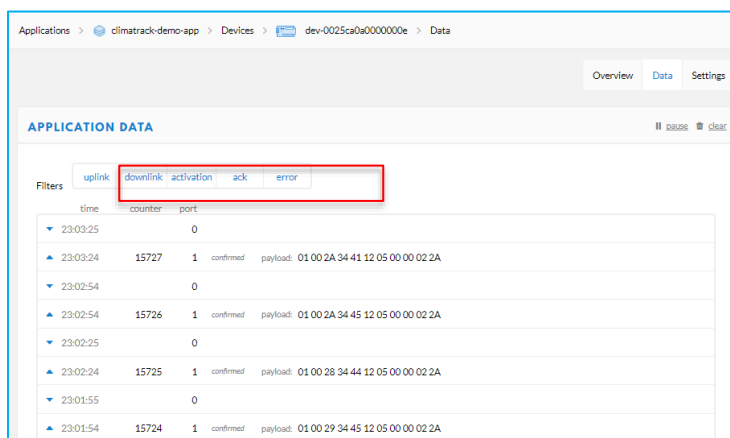
9.2.1 Common Issues

- 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 TTN server. Make sure the RS1xx settings are accurate and match what is displayed in the TTN network.

9.2.2 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, going to the data view (on the TTN device page) shows that the device received one or more activations but no uplink or downlink data.

Note: Selecting Uplink, downlink, activation, ack and error filters is possible by click on the option above the data table on the device data view.



The screenshot shows the TTN device data view for a specific device. At the top, there are tabs for 'Overview', 'Data', and 'Settings'. Below the tabs, there is a section titled 'APPLICATION DATA' with a 'filters' button. Under the filters, there are buttons for 'uplink', 'downlink', 'activation', 'ack', and 'error'. The 'downlink' button is highlighted with a red box. Below the filters, there is a table with columns for 'time', 'counter', 'port', and 'payload'. The table contains several rows of data, including timestamps, counters, and payloads.

time	counter	port	payload
23:03:25	0		
23:03:24	15727	1 confirmed	payload: 01 00 2A 34 41 12 05 00 00 02 2A
23:02:54	0		
23:02:54	15726	1 confirmed	payload: 01 00 2A 34 45 12 05 00 00 02 2A
23:02:25	0		
23:02:24	15725	1 confirmed	payload: 01 00 28 34 44 12 05 00 00 02 2A
23:01:55	0		
23:01:54	15724	1 confirmed	payload: 01 00 29 34 45 12 05 00 00 02 2A

Figure 52: Device network time

9.2.3 Connection Issues Between Registered Devices and the Things Network Servers

- Go to the TTN console and then Applications > <my Application ID> > Devices <my Device ID> >. You should see the following window.

The screenshot shows the 'DEVICE OVERVIEW' page in the TTN console. It contains the following information:

- Application ID: laird-one-demo-app
- Device ID: dev-0025ca0a0000000f
- Activation Method: OTAA
- Device EUI: 00 25 CA 0A 00 00 00 0F
- Application EUI: 70 B3 D5 7E D0 00 7D 81
- App Key: [Redacted]
- Device Address: 26 02 2B A5
- Network Session Key: [Redacted]
- App Session Key: [Redacted]
- Status: 3 seconds ago
- Frames up: 0 (with a link to reset frame counters)
- Frames down: 0

Figure 53: Device overview

- The status value is how long ago the sensor was last seen by the TTN servers. If the last seen value is never, the device never established a connection with TTN. The problem is usually that any of the following do not match the server: the App EUI, the Device EUI, or the Application Key. If the status is x number of time ago, and the App EUI, Device EUI, and Application Key have not changed, then the problem is likely elsewhere.
 - Carefully check the three keys (Device EUI/App EUI/App Key) on the Sensor device utilizing the Sensor Phone App. The Sentries RS1xx guide available from the [Sentries RS1xx product page](#) of the Laird website can help with explaining how to use the phone app.
 - 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)
- The App Key is unique for each Sentries Device and is used to derive the session keys. The App Key is not shown in the Phone App for security reasons.
 - The App Key must be reprogrammed, it cannot be verified with the phone app. If the other two values match the TTN server then the problem is often the App Key. Before reprogramming the App Key, it may be worth power cycling the sensor, wait 15 seconds and check the device web page on TTN to see if the Status have changed from never.
 - If the status still has not changed, the next step is to reprogram the App Key. The App Key is a long string that must match exactly. It may be helpful to email or otherwise copy it to the phone and paste it into the Sentries application.

9.3 Troubleshooting Tips for node_RED (MQTT Broker Links) Connections

9.3.1 MQTT Network Port Open?

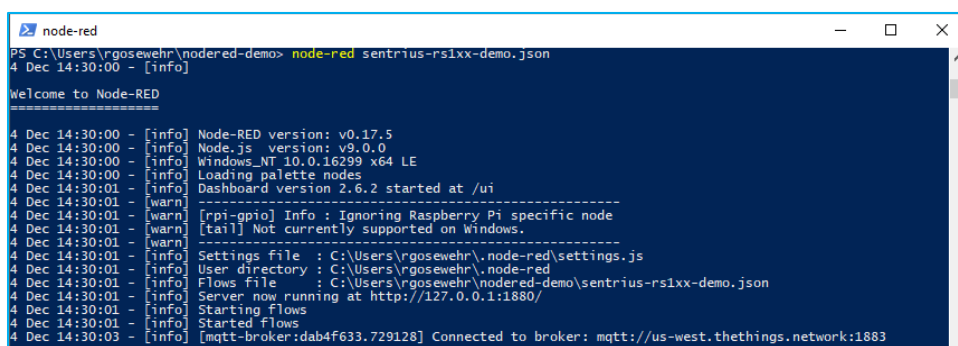
- For the devices to connect to the Things Network Servers and communicate with the MQTT Broker Links in the node-RED dashboard, the Network Port needs to have an opened TCP socket:
 - us-west.thethings.network (us-west portion of URL can vary depending on the TTN server being used): **Port 1883 MQTT**
- The most frequent problem Users encounter when running this demo, is that the required IP ports are not open or have traffic restrictions in place on these ports.

9.3.2 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.

9.3.3 MQTT Broker Nodes Connection

- The three MQTT nodes (**Device Subscription, Activations Subscription and Downlink**) should show a connected status. These all use the same configuration node (MQTT Broker)
 - Access the Hamburger Menu > Configuration Nodes > MQTT-Broker. (Figure 48). Make sure the username is the **TTN Application ID** and the password is the **Application Default Access key**.
- 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. The debug window in node-RED can be enabled by clicking on the debug tab in the right-hand pane of the flow editor. When active, debug node 1 will show all device data sent from the TTN server for the Application ID (set in the Configuration Node). Debug node 2 will display data.
 - Below shows a properly connected Terminal Window running node-RED with the predefined Flow settings.



```
node-red
PS C:\Users\rgosewehr\nodered-demo> node-red sentrius-rs1xx-demo.json
4 Dec 14:30:00 - [info]
Welcome to Node-RED
-----
4 Dec 14:30:00 - [info] Node-RED version: v0.17.5
4 Dec 14:30:00 - [info] Node.js version: v9.0.0
4 Dec 14:30:00 - [info] Windows_NT 10.0.16299 x64 LE
4 Dec 14:30:00 - [info] Loading palette nodes
4 Dec 14:30:01 - [info] Dashboard version 2.6.2 started at /ui
4 Dec 14:30:01 - [warn]
-----
4 Dec 14:30:01 - [warn] [rpi-gpio] Info : Ignoring Raspberry Pi specific node
4 Dec 14:30:01 - [warn] [tail] Not currently supported on Windows.
4 Dec 14:30:01 - [warn]
-----
4 Dec 14:30:01 - [info] Settings file : C:\Users\rgosewehr\node-red\settings.js
4 Dec 14:30:01 - [info] User directory : C:\Users\rgosewehr\node-red
4 Dec 14:30:01 - [info] Flows file : C:\Users\rgosewehr\nodered-demo\sentrius-rs1xx-demo.json
4 Dec 14:30:01 - [info] Server now running at http://127.0.0.1:1880/
4 Dec 14:30:01 - [info] Starting flows
4 Dec 14:30:01 - [info] Started flows
4 Dec 14:30:03 - [info] [mqtt-broker:dab4f633.729128] Connected to broker: mqtt://us-west.thethings.network:1883
```

Figure 54: Properly-connected terminal window