Texas Instruments Sensor2Cloud Linux – Getting Started Guide

Introduction

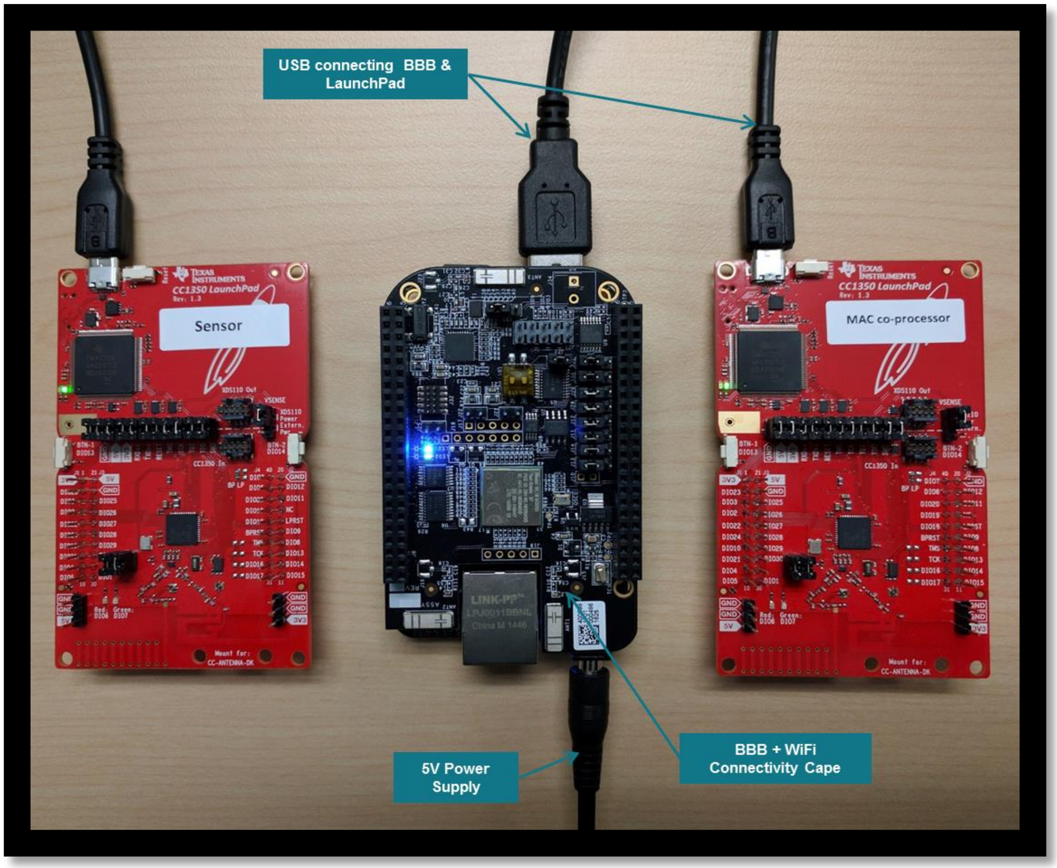
This guide provides quick steps to run the out of box **Sensor2Cloud** example. This kit enables you to view and control sensors on a Sub1-GHz network from the cloud. All the devices in the kit are pre-programed with required software. The kit includes two methods of operation: a cloud connected gateway and a local gateway. The cloud connected gateway allows you to leverage cloud services such as Amazon Web Services (AWS) and IBM Cloud. The local gateway allows you to run a gateway within a local IP network for privately managed networks that do not require external connectivity. It's important to note that the design allows you to quickly add **any** desired cloud service.

Kit Contents

- 5V Power Supply
- BeagleBone Black
- BeagleBone Wireless Connectivity Cape
- FTDI Cable
- Pre-flashed MicroSD
- 2 x TI SimpleLink™ CC1350 LaunchPad™
- 2 x Micro-USB Type B Cable

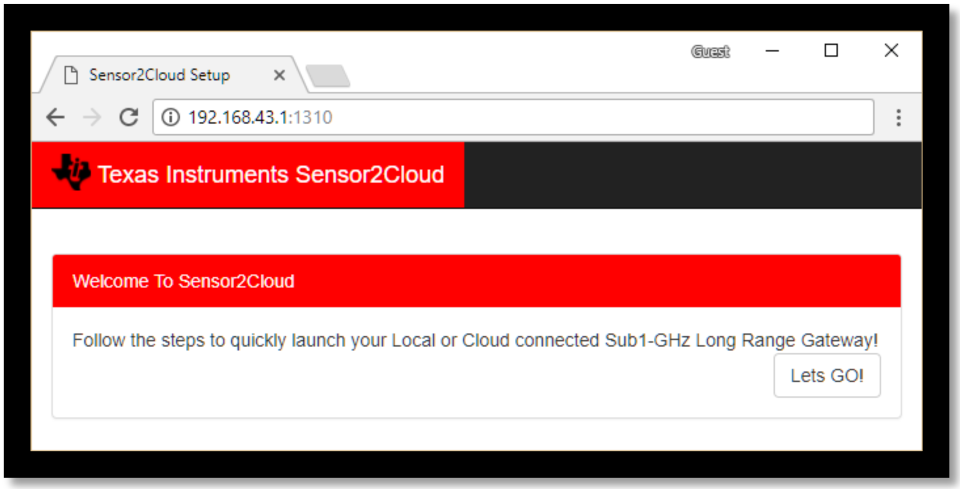
Hardware Setup

The BeagleBone Black (BBB) comes equipped with a BeagleBone Wireless Connectivity Cape and pre-inserted microSD card with necessary software. Connect the BBB to the CC1350 LaunchPad labeled ***MAC co-processor*** as shown below. Connect the 5V power supply to the BBB and an electrical outlet. The CC1350 LaunchPad labeled ***Sensor*** should be connected using the provided USB to any USB power source. A blue LED on the BBB and a green LED on each of the CC1350 LaunchPads should illuminate.



Running the Demo

The BBB operates out the box as a Wi-Fi access point with SSID: ***Sitara AP***. Connect to the network with a PC by using the password: ***sensor2cloud***. Once connected, open a browser and type in the address bar: **http://*192.168.43.1:1310***. This will open the main Sensor2Cloud portal hosted on the BBB.

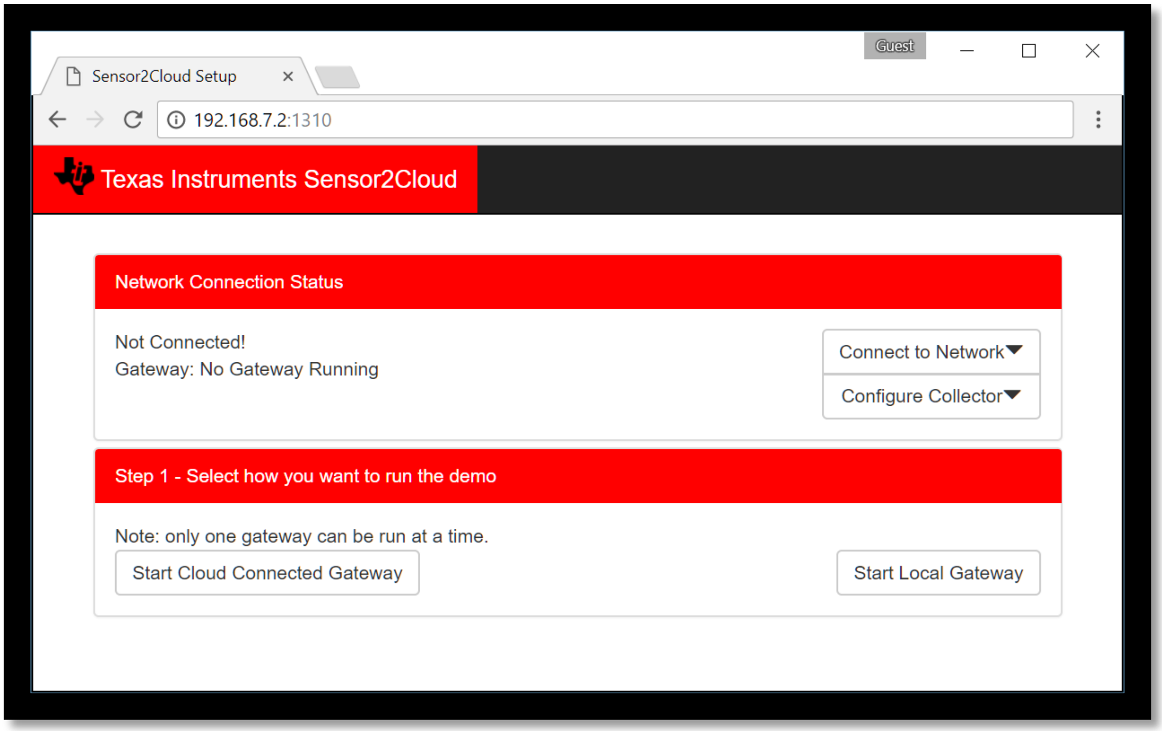


Welcome To Sensor2Cloud

Follow the steps to quickly launch your Local or Cloud connected Sub1-GHz Long Range Gateway!

Select the method of operation for the demo:

1. **Cloud Connected Gateway**
   - Monitor and control the Sub1-GHz network using AWS or IBM Cloud Service.
2. **Local Gateway**
   - Monitor and control the Sub1-GHz network using a local IP network for privately managed networks that do not need external connectivity.



Network Connection Status

Not Connected!
Gateway: No Gateway Running

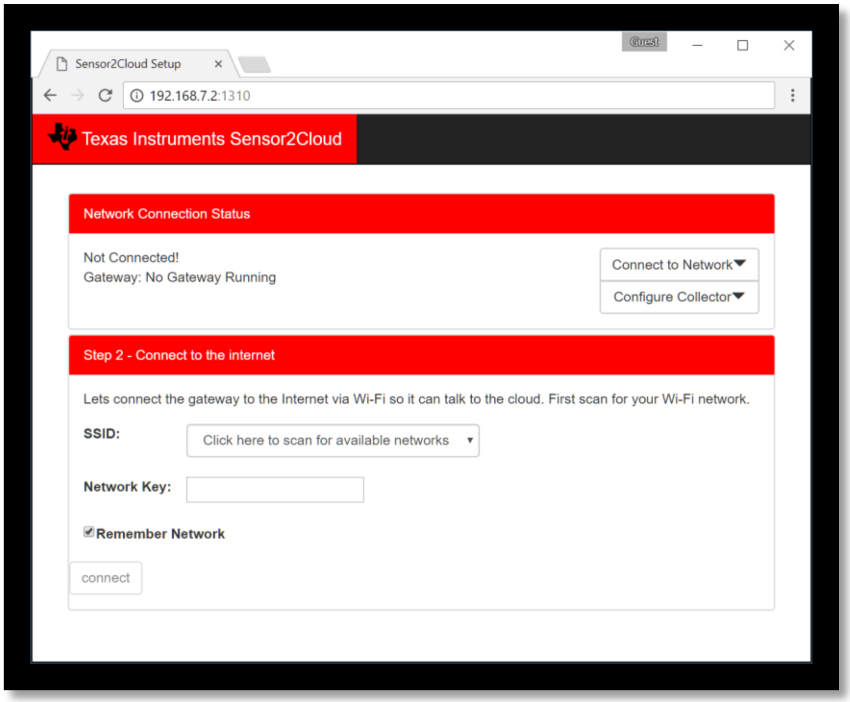
Step 1 - Select how you want to run the demo

Note: only one gateway can be run at a time.

Running Cloud Connected Gateway

To run the cloud connected gateway, the BBB must be connected to the internet.

1. Select the ***scan for available networks*** drop down to begin a 10 second network scan. Selecting **Remember Network** will cause the BBB to automatically connect to the selected network when possible. This is true even after the device has been power cycled.



Sensor2Cloud Setup

192.168.7.2:1310

Texas Instruments Sensor2Cloud

Network Connection Status

Not Connected!
Gateway: No Gateway Running

Step 2 - Connect to the internet

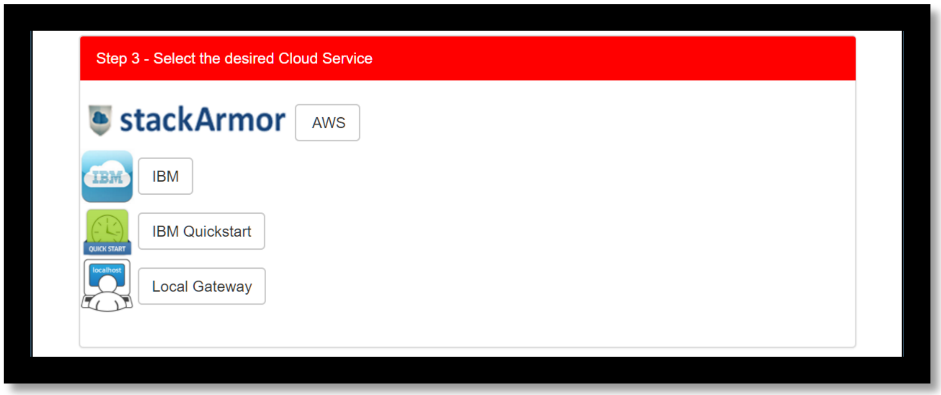
Lets connect the gateway to the Internet via Wi-Fi so it can talk to the cloud. First scan for your Wi-Fi network.

SSID: Click here to scan for available networks ▾

Network Key:

☑ **Remember Network**

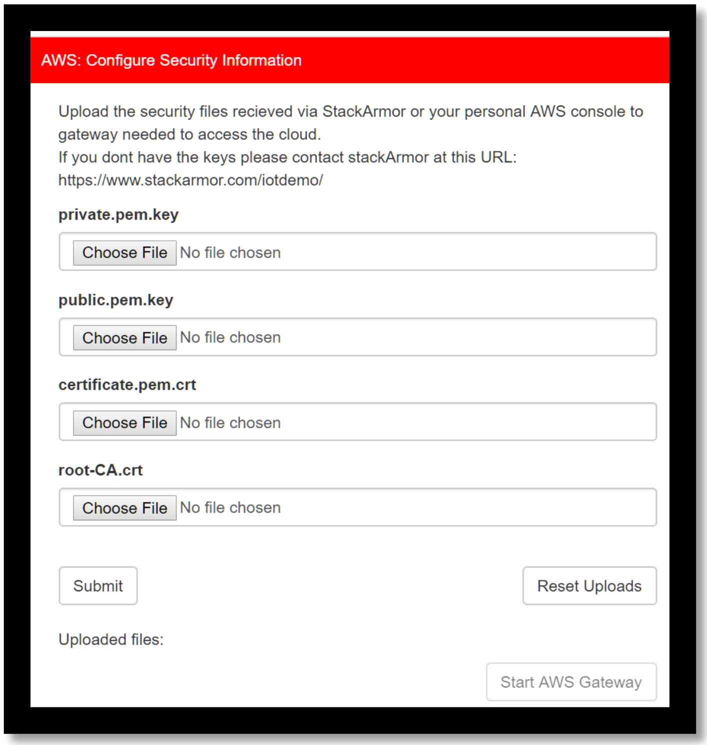
2. After successfully connecting to the internet, select the desired cloud service. Note that the ***Local Gateway*** option may be selected despite it not requiring an internet connection. Amazon Web Services (AWS) is provided through stackArmor.



Step 3 - Select the desired Cloud Service

Using Amazon Web Services (AWS)

1. Request credentials from stackArmor.
   - If approved, stackArmor will provide the security credentials necessary for the steps below.
2. Use ***Choose File*** to upload the security credentials received from stackArmor. Click ***Submit*** to upload the selected files.



AWS: Configure Security Information

Upload the security files recieved via StackArmor or your personal AWS console to gateway needed to access the cloud.
If you dont have the keys please contact stackArmor at this URL:
https://www.stackarmor.com/iotdemo/

private.pem.key
Choose File No file chosen

public.pem.key
Choose File No file chosen

certificate.pem.crt
Choose File No file chosen

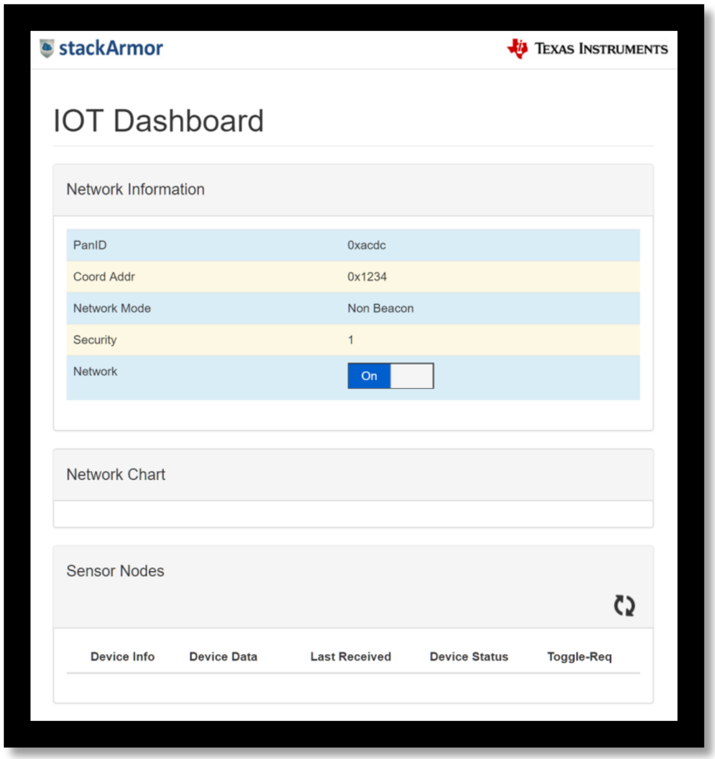
root-CA.crt
Choose File No file chosen

Submit Reset Uploads

Uploaded files:

Start AWS Gateway

3. After successfully uploading the security files, select ***Start AWS Gateway***.
   - If prompted, enter the username and password provided via email from stackArmor.
4. Set the ***Network*** to ***On*** to allow the sensor(s) to join.



IOT Dashboard

Network Information

PanID	0xacdc
Coord Addr	0x1234
Network Mode	Non Beacon
Security	1
Network	On

Network Chart

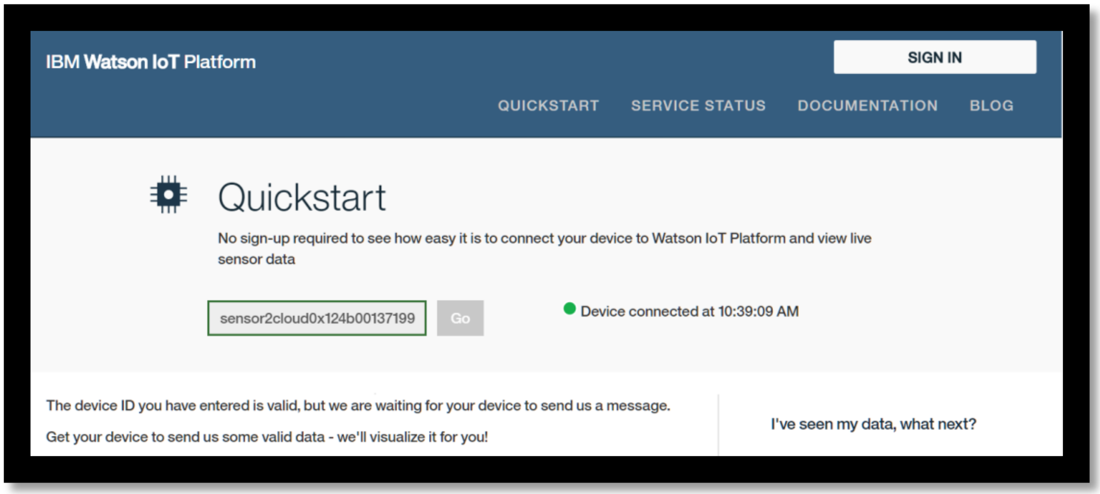
Sensor Nodes

Device Info	Device Data	Last Received	Device Status	Toggle-Req

5. Power on the sensor(s), if not done so already.
    - The sensor(s) will automatically join the network and appear in the ***Network Chart*** and ***Sensor Nodes*** table.

Using IBM Quickstart

Selecting ***IBM Quickstart*** will launch an IBM Watson IoT Platform.



Quickstart

No sign-up required to see how easy it is to connect your device to Watson IoT Platform and view live sensor data

sensor2cloud0x124b00137199

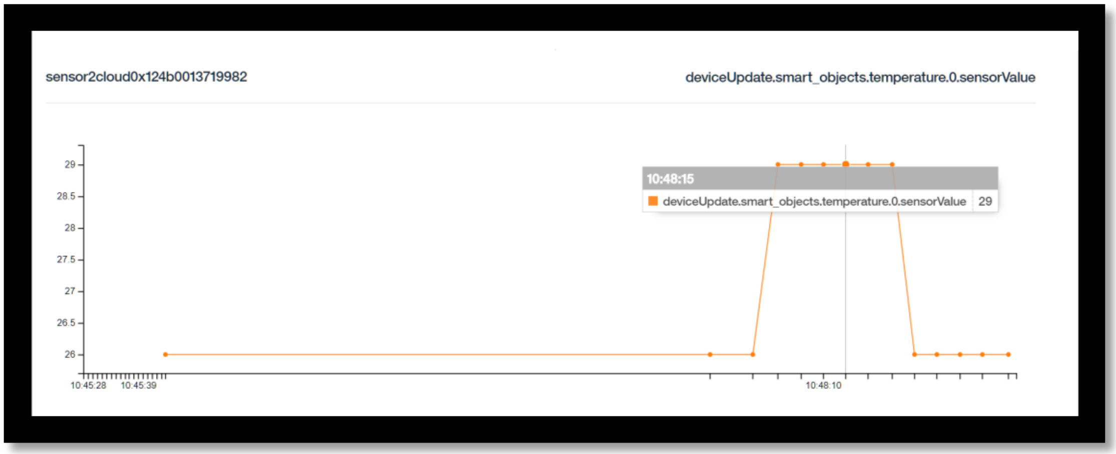
● Device connected at 10:39:09 AM

The device ID you have entered is valid, but we are waiting for your device to send us a message.

Get your device to send us some valid data - we'll visualize it for you!

I've seen my data, what next?

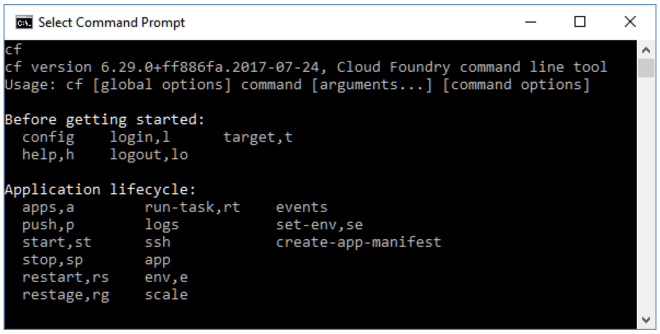
1. Power on the sensor(s), if not done so already.
    - The webpage will automatically update with the sensor data.
2. Select a sensor data point to see its value graphed. In the figure below, ***smart_objects.temperature.0.sensorValue*** is selected.


sensor2cloud0x124b0013719982
deviceUpdate.smart_objects.temperature.0.sensorValue
29
28.5
28
27.5
27
26.5
26
10:45:28
10:45:39
10:48:10
10:48:15
deviceUpdate.smart_objects.temperature.0.sensorValue
29

Using IBM Cloud Service

1. Download and install Cloud Foundry CLI.

   1.1. Verify installation by typing the command, ***cf***, on the system terminal.



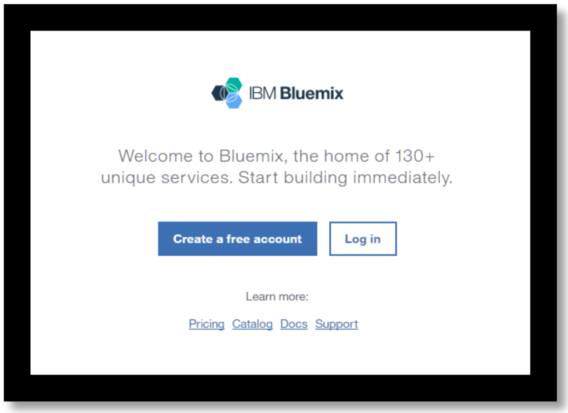
```
cf
cf version 6.29.0+ff886fa.2017-07-24, Cloud Foundry command line tool
Usage: cf [global options] command [arguments...] [command options]

Before getting started:
  config    login,l      target,t
  help,h    logout,lo

Application lifecycle:
  apps,a        run-task,rt    events
  push,p        logs           set-env,se
  start,st      ssh            create-app-manifest
  stop,sp       app
  restart,rs    env,e
  restage,rg    scale
```

2. If you do not already have one, you will need to open and configure an IBM® Bluemix® Account.

   2.1. Create an IBM Bluemix Account and register a 30-day trial account.

   2.2. Confirm the Bluemix account using the link provided by IBM through email.

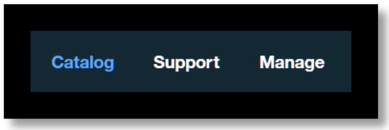
   2.3. Log into the Bluemix account.



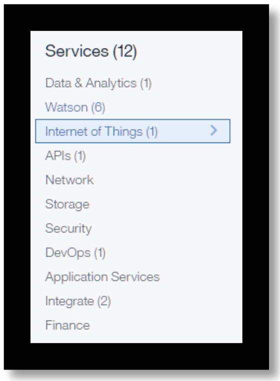
Welcome to Bluemix, the home of 130+ unique services. Start building immediately.

3. Create a service.

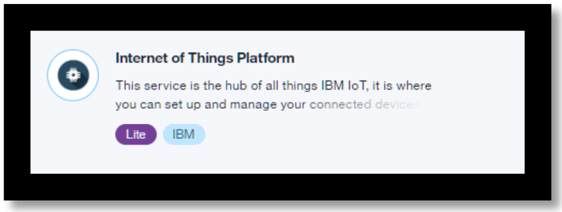
   3.1. From the IBM BlueMix Dashboard, select ***Catalog*** from the upper right hand corner.



   3.2. On the left hand menu, select ***Internet of Things***.



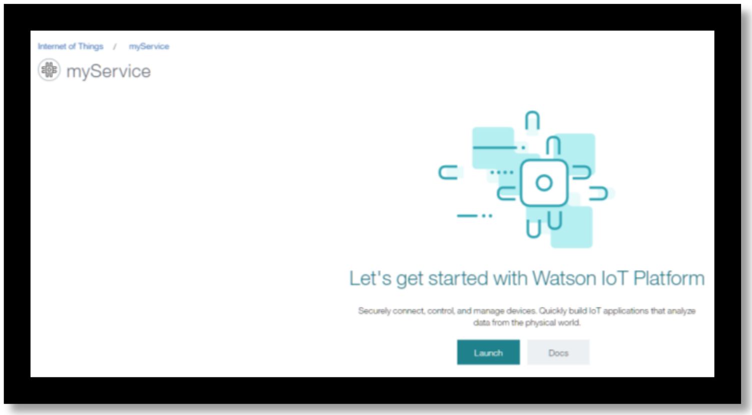
3.3. Now select Internet of Things Platform.



This service is the hub of all things IBM IoT, it is where you can set up and manage your connected devices

3.4. For this guide, use *myService* as the ***Service name***, though any name may be used. The service name will be used later on in this guide. The **Lite** pricing plan should be pre-selected and is free for trial users. Click ***Create***.

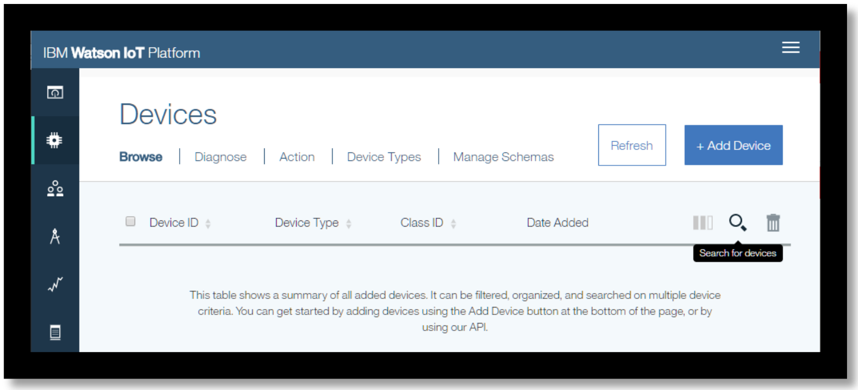
3.5. Select ***Launch***.



Let's get started with Watson IoT Platform

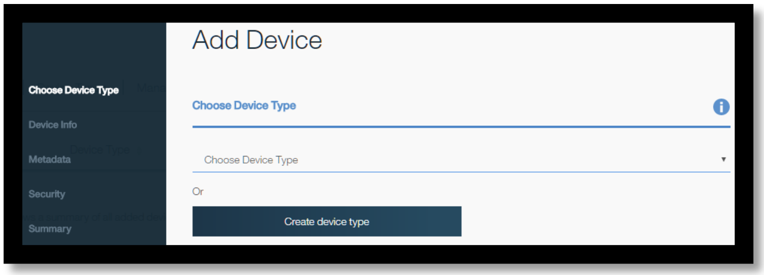
Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.

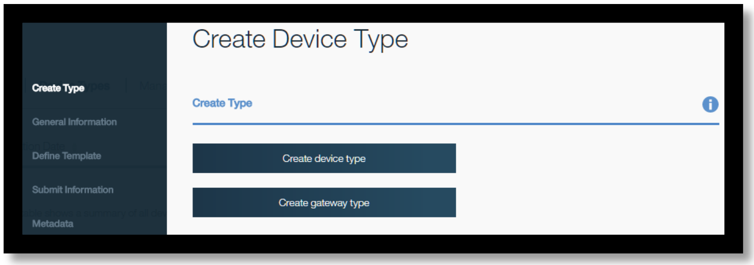
3.6. From the right hand menu, select ***Devices*** and click ***+ Add Device***.



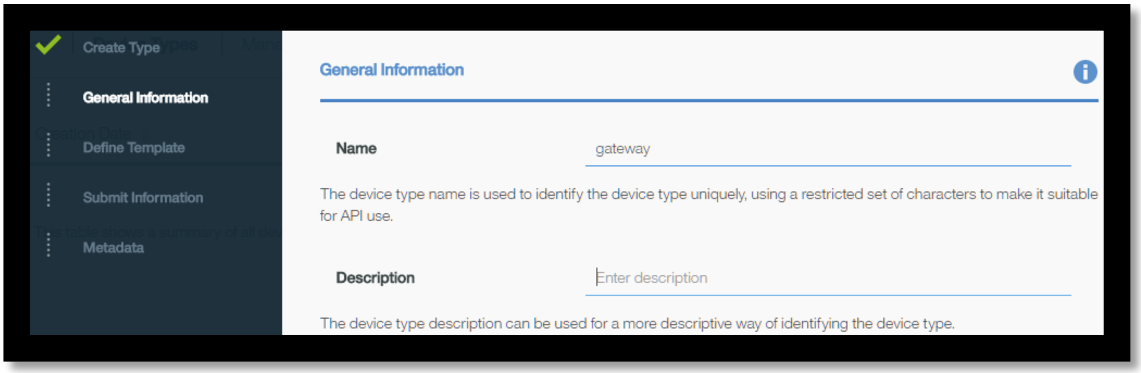
This table shows a summary of all added devices. It can be filtered, organized, and searched on multiple device criteria. You can get started by adding devices using the Add Device button at the bottom of the page, or by using our API.

3.7. Select ***Create device type*** and ***create gateway type***.





3.8. Enter *gateway* as the ***Name*** and add an optional description. Select ***Next*** in the bottom right hand side.



General Information

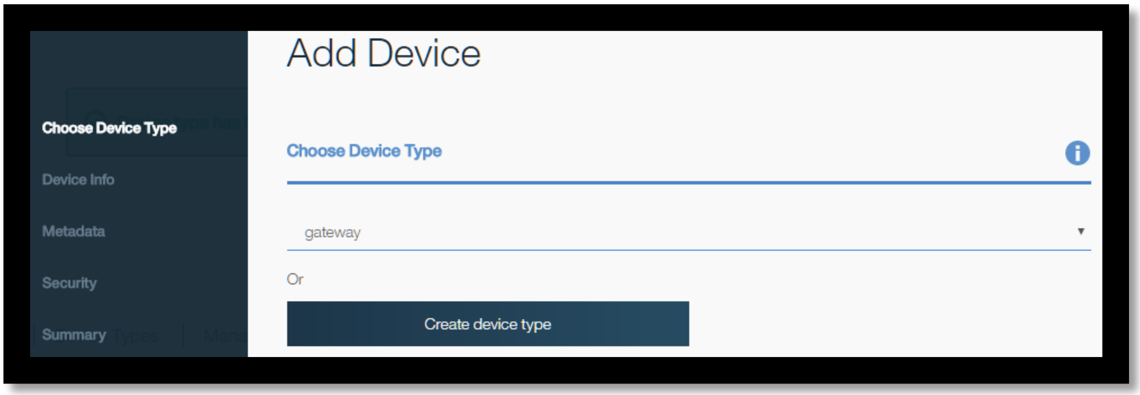
Name gateway

The device type name is used to identify the device type uniquely, using a restricted set of characters to make it suitable for API use.

Description Enter description

The device type description can be used for a more descriptive way of identifying the device type.

3.9. Select ***Next***—ignoring the subsequent options—until reaching the ***Add Device*** screen seen in step 3.7. From the ***Choose Device Type*** drop down menu, select *gateway*. Click ***Next***.



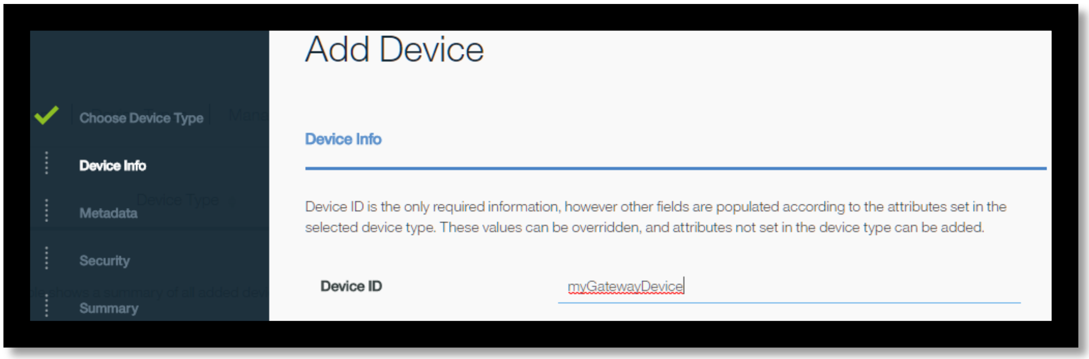
Add Device

Choose Device Type

gateway

Or

3.10. Enter a ***Device ID***. Take note as the Device ID used as this will be needed in a later step. This guide uses *myGatewayDevice*. Click ***Next*** until reaching the ***Security*** step.



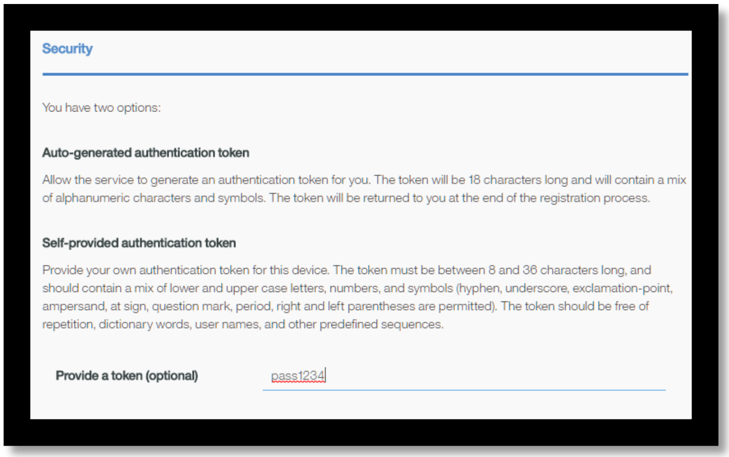
Add Device

Device Info

Device ID is the only required information, however other fields are populated according to the attributes set in the selected device type. These values can be overridden, and attributes not set in the device type can be added.

Device ID myGatewayDevice

3.11. Enter a token. Take not of the token used as this will be needed in a later step. Click ***Next***.



Security

You have two options:

Auto-generated authentication token

Allow the service to generate an authentication token for you. The token will be 18 characters long and will contain a mix of alphanumeric characters and symbols. The token will be returned to you at the end of the registration process.

Self-provided authentication token

Provide your own authentication token for this device. The token must be between 8 and 36 characters long, and should contain a mix of lower and upper case letters, numbers, and symbols (hyphen, underscore, exclamation-point, ampersand, at sign, question mark, period, right and left parentheses are permitted). The token should be free of repetition, dictionary words, user names, and other predefined sequences.

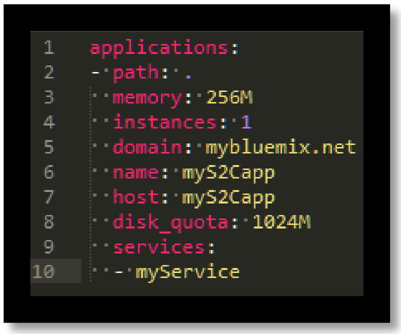
Provide a token (optional) pass1234

3.12. Use the Summary view to verify the ***Device ID*** and ***Authentication Token***. Use the ***Back*** button in the bottom right hand side to return and edit any items. When done, click ***Add***. Your device credentials will be displayed.

4. Upload the provided Cloud Foundry App.

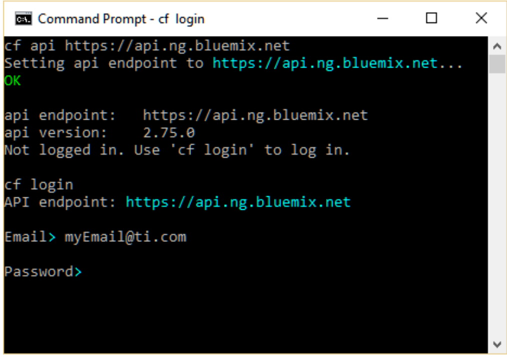
4.1. Unzip the provided <filename>.zip.

4.2. Navigate to <folder>/<subfolder> and use a text editor to modify the ***manifest.yml*** file. Select a ***name*** and unique hostname (***host***). The ***host*** will be used to generate a unique link to your cloud application. You will be able to tell if your selected hostname is already used in step 4.4. Add your service name to ***services*** using the service name from step 3.4.



```
applications:
- path: .
  memory: 256M
  instances: 1
  domain: mybluemix.net
  name: myS2Capp
  host: myS2Capp
  disk_quota: 1024M
  services:
  - myService
```

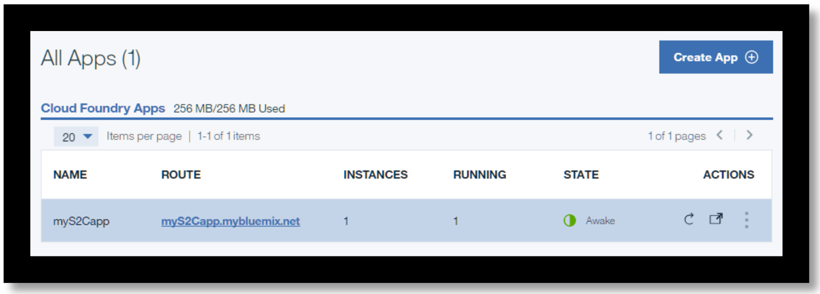
4.3. Open a system terminal and navigate to the <subfolder> directory. Enter the command, ***cf api https://api.ng.bluemix.net***. Then use the ***cf login*** command and login with your account credentials created in step 2.



```
cf api https://api.ng.bluemix.net
Setting api endpoint to https://api.ng.bluemix.net...
OK

api endpoint:   https://api.ng.bluemix.net
api version:    2.75.0
Not logged in. Use 'cf login' to log in.

cf login
API endpoint: https://api.ng.bluemix.net

Email> myEmail@ti.com

Password>
```

4.4. After successfully logging in, use ***cf push*** to upload the application. Note that if a unique hostname is not used, an error message will be generated. In that case, return to the ***manifest.yml*** and modify the ***host*** and try again.

4.5. Open the [IBM Bluemix dashboard](). The application will now be visible under ***Cloud Foundry Apps***. Note that the route column displays a unique hostname for the application. Select the application by clicking the row—do not click the URL in the route column. This will open the application overview.

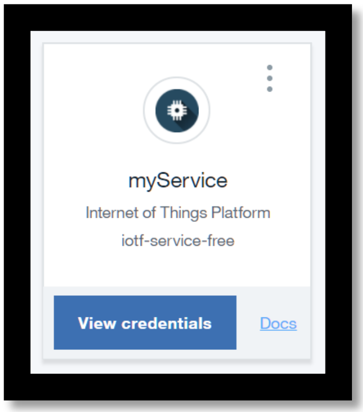


All Apps (1)
Create App
Cloud Foundry Apps 256 MB/256 MB Used
20 Items per page | 1-1 of 1 items
1 of 1 pages

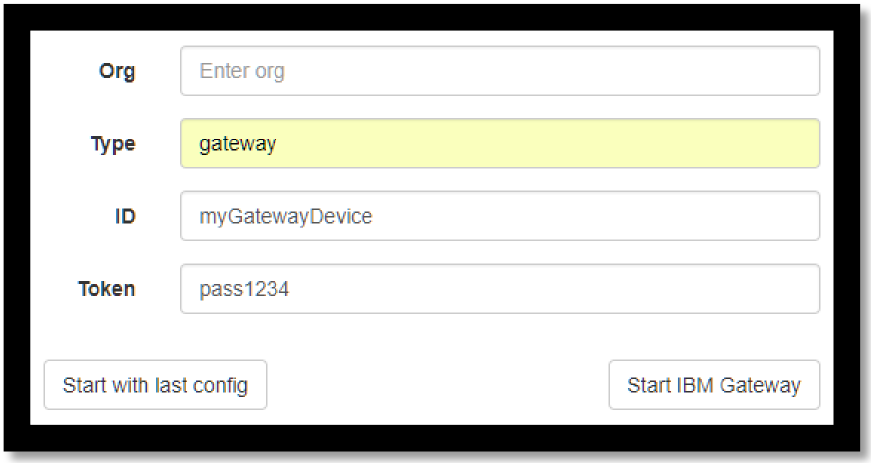
NAME	ROUTE	INSTANCES	RUNNING	STATE	ACTIONS
myS2Capp	myS2Capp.mybluemix.net	1	1	Awake	

4.6. Click ***Connections*** in the left hand menu. The service created in step 3 should be visible. If not, use the ***Connect existing*** button to add the service.

5. Click **View credentials**. This will display all the information necessary to launch your BBB gateway.

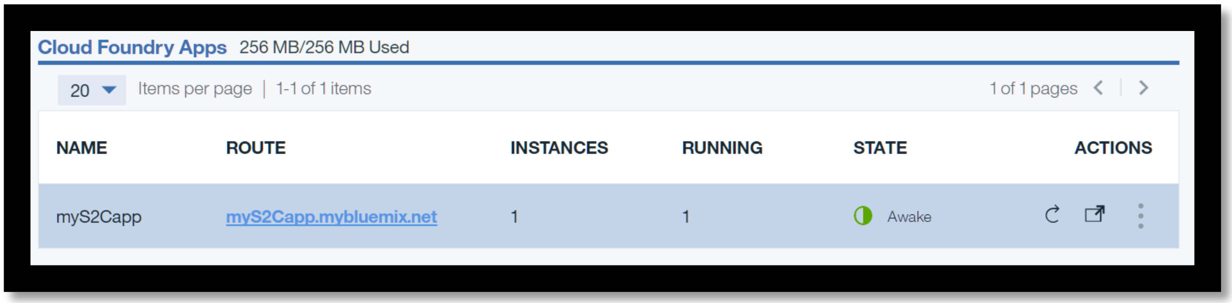


6. In another web browser window, access the BBB Sensor2Cloud portal and select ***IBM*** as the cloud service. Use the ***Org*** listed in the service credentials from step 5. For ***Type*** use *gateway* and ***ID*** use the device ID from step 3.10. Lastly, use the token created in step 3.11. Click ***Start IBM Gateway***.



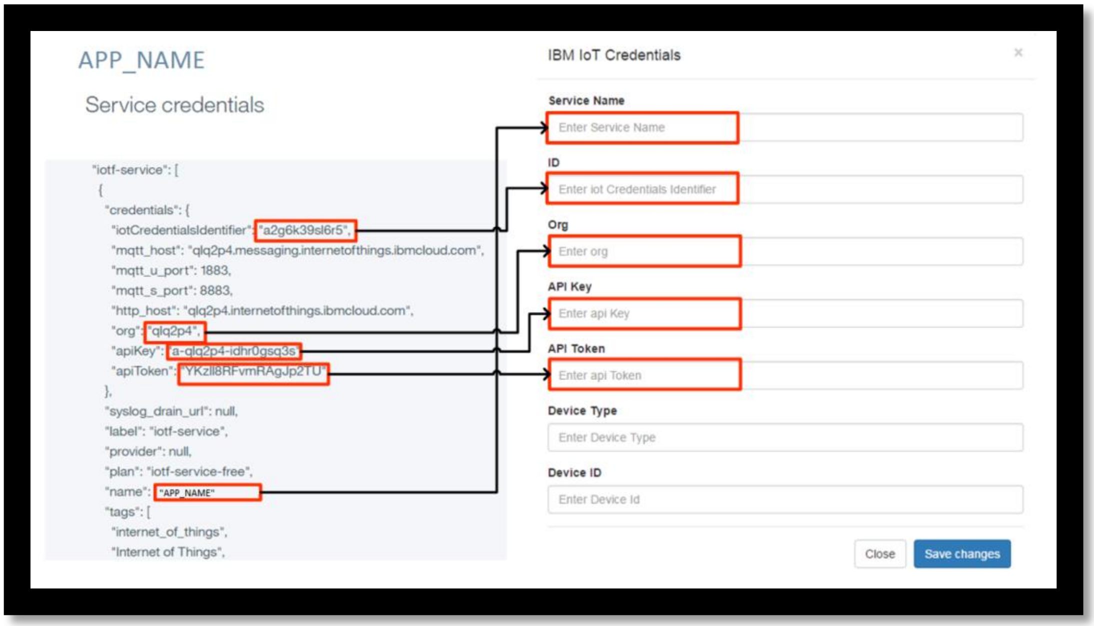
Org	Enter org
Type	gateway
ID	myGatewayDevice
Token	pass1234

7. From the IBM Bluemix dashboard, click the application URL in the route column. This will launch the Sub 1G Sensor To Cloud application webpage.


Cloud Foundry Apps 256 MB/256 MB Used
20
Items per page | 1-1 of 1 items
1 of 1 pages

NAME	ROUTE	INSTANCES	RUNNING	STATE	ACTIONS
myS2Capp	myS2Capp.mybluemix.net	1	1	Awake	

8. Select the gear icon in the upper right hand corner. Using the credentials from step 5, enter the following information. For ***Device Type***, use *gateway*. Use the same ***Device ID*** used in step 6. Click ***Save changes***.



APP_NAME

Service credentials

```
"iotf-service": [
 {
  "credentials": {
   "iotCredentialsIdentifier": "a2g6k39sl6r5",
   "mqtt_host": "qlq2p4.messaging.internetofthings.ibmcloud.com",
   "mqtt_u_port": 1883,
   "mqtt_s_port": 8883,
   "http_host": "qlq2p4.internetofthings.ibmcloud.com",
   "org": "qlq2p4",
   "apiKey": "a-qlq2p4-idhr0gsq3s",
   "apiToken": "YKzll8RFvmRAgJp2TU"
  },
  "syslog_drain_url": null,
  "label": "iotf-service",
  "provider": null,
  "plan": "iotf-service-free",
  "name": "APP_NAME"
  "tags": [
   "internet_of_things",
   "Internet of Things",
```

IBM IoT Credentials

Service Name: Enter Service Name

ID: Enter iot Credentials Identifier

Org: Enter org

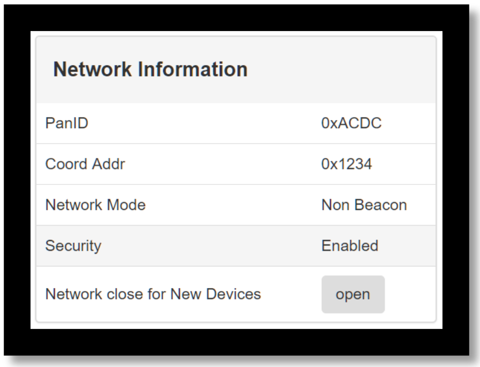
API Key: Enter api Key

API Token: Enter api Token

Device Type: Enter Device Type

Device ID: Enter Device Id

9. Select ***open*** to allow the sensor(s) to join the network.



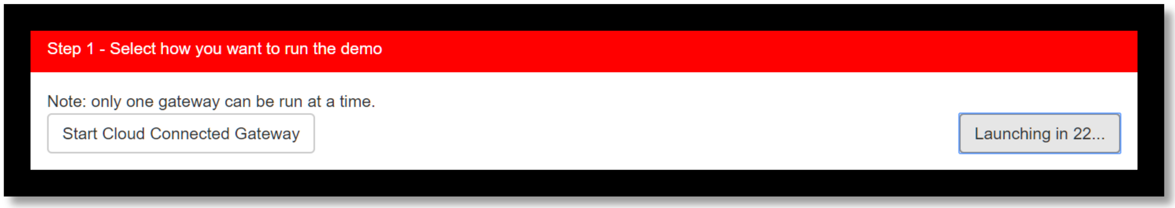
Network Information

PanID	0xACDC
Coord Addr	0x1234
Network Mode	Non Beacon
Security	Enabled
Network close for New Devices	open

10. Power on the sensor(s), if not done so already.
    - The sensor(s) will automatically join the network.

Running Local Gateway

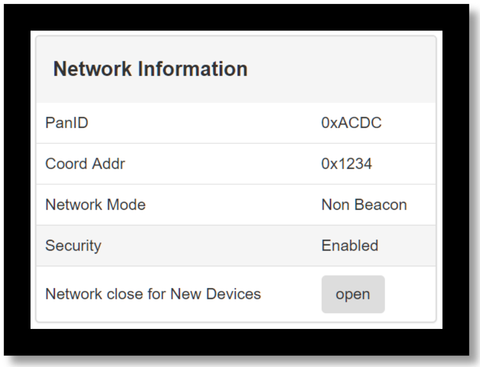
Selecting ***Local Gateway*** will start a countdown timer from 30 seconds. When the timer expires, the page will automatically redirect to the **TIMAC CollectorApp**.



Step 1 - Select how you want to run the demo

Note: only one gateway can be run at a time.

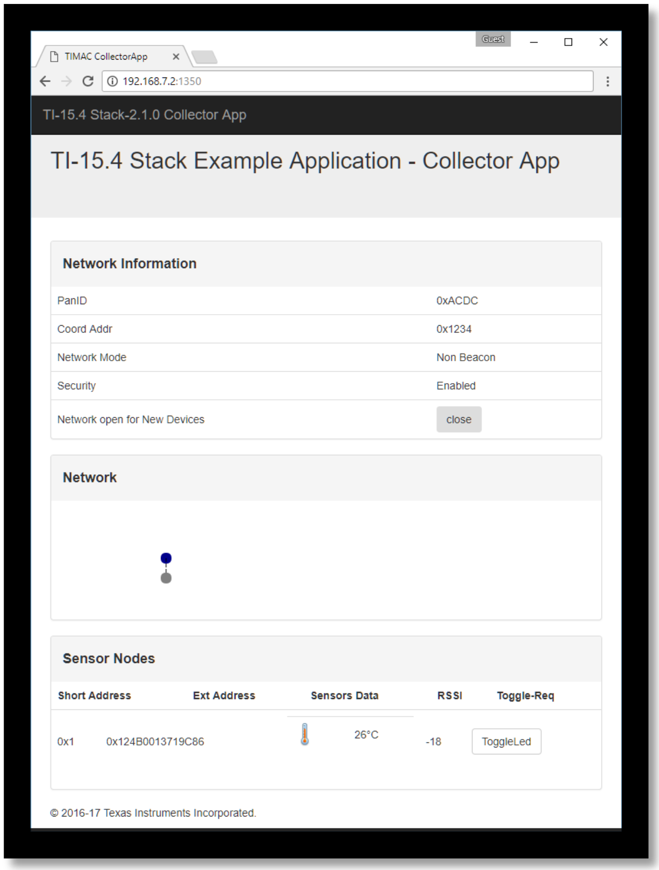
1. Select ***open*** to allow the sensor(s) to join the network.



Network Information

PanID	0xACDC
Coord Addr	0x1234
Network Mode	Non Beacon
Security	Enabled
Network close for New Devices	open

2. Power on the sensor(s), if not done so already.
   - The sensor(s) will automatically join the network.



TI-15.4 Stack-2.1.0 Collector App

TI-15.4 Stack Example Application - Collector App

Network Information

PanID	0xACDC
Coord Addr	0x1234
Network Mode	Non Beacon
Security	Enabled
Network open for New Devices	close

Network

Sensor Nodes

Short Address	Ext Address	Sensors Data	RSSI	Toggle-Req
0x1	0x124B0013719C86	🌡 26°C	-18	ToggleLed

© 2016-17 Texas Instruments Incorporated.

Troubleshooting

Sensor will not join the network.

Toggle the option to “open” the network for sensors through the web interface. This will ensure the network is open.

Sensors use non-volatile storage to remember previous networks. Press and hold ***BTN-2/DIO014*** and toggle the ***Reset*** button. This will clear the non-volatile storage and will cause the sensor to search for new networks.