



IBM BlueMix Workshop

Creating Internet of Things apps using BlueMix

The information contained herein is proprietary to IBM. The recipient of this document, by its retention and use, agrees to protect the information contained herein. Readers are advised that this project may be subject to the terms of a non-disclosure agreement. DO NOT DISCLOSE ANY OF THIS INFORMATION WITHOUT OBTAINING PERMISSION FROM THE MANAGEMENT RESPONSIBLE FOR THIS DOCUMENT.

Overview

In this lab, we will showcase

1. A device sending data to the Internet of Things Cloud (IoTCloud)
2. Create an application in BlueMix to use that data
3. Analyze and act upon that data

We use a device that sends temperature data to the IoTCloud, and with the help of IoT application (node-RED) in BlueMix, we read the temperature and tweet temperature alerts if the temperature goes above 50 deg C or below 10 deg C.

We can either use the Raspberry Pi model B or the Online device simulator for this lab. (If you are using Raspberry Pi, then follow the steps for setting it up in the section below. Otherwise, that section can be ignored)

Please note that the devices using the [MQTT](#) protocol to distribute data across the internet.

Lab prerequisites

This lab assumes that you have a few common tools already installed. These pre-reqs and where you can find them are listed below.

1. BlueMix account

You need valid account in the BlueMix beta. You can sign up here <https://ace.ng.bluemix.net/>

2. Raspberry Pi Model B (Optional)

You require a **Raspberry Pi Model B with network connectivity**. Normally, Raspberry Pi comes with Raspbian OS pre-installed.

If you **do not own** a Raspberry Pi, you can use the device simulator sensor <http://iotsensor.ng.bluemix.net/>

3. Twitter account

You need a valid account with Twitter. Your twitter account will be used to tweet alerts.

Setup

If you are using Raspberry Pi, then follow the steps below. This sets up the Raspberry Pi to send data to the IoTCloud. In this exercise, Raspberry Pi sends CPU temperature, CPU load and a simulated sine wave.

1. Setting up Raspberry Pi device

1. Log in to Raspberry Pi using Putty or any ssh client. (Default login Username: pi Password: raspberry).
2. Download the tarball using the following command:

```
wget https://github.com/ibm-messaging/iot-raspberrypi/releases/download/1.0.0-nodered/iot-1.0.0-nodered.tar.gz
```

3. Untar it using **tar -xvf iot-1.0.0-nodered.tar.gz**.
4. **cd iot-raspi-nodered**
5. Run **sudo ./setup.sh**. This completes the following tasks: Installs Node-RED, contributed nodes, npm requisites, and configures a system service to run a supplied flow. This will take a **few minutes**. (This command requires **sudo** as it registers the Node-RED as a init service in Raspberry Pi)
6. Find the MAC address of your device by running **./iotGetDeviceID.sh**. Note down this value as we will use it later in BlueMix.
7. Start Node-RED with **sudo service node-red start**

You can verify if your device has started emitting events by checking in <http://quickstart.internetofthings.ibmcloud.com/>. Type in your MAC address and you should be able to see the graph of the events.

If you are using the device simulator, follow the steps below.

2. IoT Device Simulator

1. Open <http://iotsensor.ng.bluemix.net/>
2. There are three data points
 - a. Temperature
 - b. Humidity
 - c. Object temperature

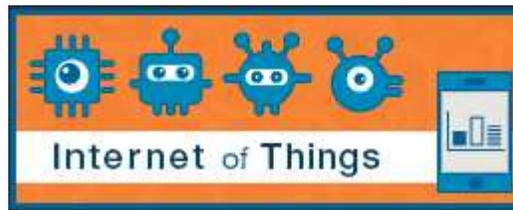
3. You should be able to change the values by clicking on the arrows in the bottom.
4. The simulator also has a MAC address on the top right corner. Note down this value as this is required later. Click on the MAC address to see the graphical visualization of the events sent by the device simulator. (Do not reload the page after noting the MAC address. This address will be input in Node-RED. If you do reload the page, note and input the new MAC address in Node-RED.)

BlueMix Development

1. Log in to BlueMix. (you need to provide your BlueMix id and password)
<https://ace.ng.bluemix.net/>
2. Go to Catalog and click on Internet of Things Boilerplate.

Notice that the Boilerplate deploys a combination of BlueMix services including:

- **Node-Red** – a visual tool for wiring and programming the Internet of Things
- **Time series database** – for collecting and archiving Internet of Things data
- **Internet of Things API** – for connecting to sensors and devices using protocols like MQTT



3. Then click on “Create Application”. Input the name and host for the application of your choice. This will create the application on that given host. (The screen shot below uses the string “MyIoTTest”)

×

Finish adding **Internet of Things Starter**

Name:

Host: .ng.bluemix.net

4. Click on the application URL that was created. In this case, the URL is:
<http://MyIoTTest.ng.bluemix.net/>
5. Click on “Go to your node-RED flow editor.”

Node-RED

Node-RED is a tool for wiring together hardware devices, APIs and online services. In this lab, we will use Node-RED to get data from the devices using the IoT node and act upon that data.

1. Click menu (top-right corner), then click on Import from Clipboard or press Ctrl+I.

2. Paste the following code (single line)

```
[{"id":"588f19a5.a1db88","type":"iot","name":"IoT
device","deviceId":"","messageType":"*","x":103,"y":195,"z":"11ea8b41.55b0cd","wires":["c5767134.51
056"]}],{"id":"c5767134.51056","type":"function","name":"Extract temperature","func":"var data =
{};\n\nif (!context.global.hasOwnProperty(\"device\")) {\n\tcontext.global.device = \"\";\n}\n\nvar events
= msg.payload.d;\nvar temperature = 0;\n\n// Raspberry Pi events have \"myPi\" name, \n// so extract
cputemp \nif(events.myName == \"myPi\") {\n\tcontext.global.device = \"Raspberry Pi\";\n\ttemperature
= events.cputemp;\n} else {\n// IoT simulator sensor events\n// so extract temp\n\tcontext.global.device
= \"IoT sensor\";\n\ttemperature = events.temp;\n}\n\nndata.cputemp = temperature;\n\nmsg.payload =
temperature;\n\nreturn
msg;","outputs":1,"x":293,"y":196,"z":"11ea8b41.55b0cd","wires":["dec22b67.b2ea38"]}],{"id":"dec22b6
7.b2ea38","type":"switch","name":"Switch high and low
temp","property":"payload","rules":[{"t":"gt","v":"50"},{"t":"lt","v":"10"}],"checkall":"true","outputs":2,"x
":552,"y":196,"z":"11ea8b41.55b0cd","wires":["975d7125.34bdc8"],["727dfdf1.29cfcc"]}],{"id":"975d712
5.34bdc8","type":"function","name":"High Temp","func":"var temperature = msg.payload;\n\nvar
twitMessage = \"HIGH temperature Alert @ \"+context.global.device+\"!! The temperature is
\"+temperature+\" deg C at \"+new Date();\n\nreturn {topic: \"\", payload:
twitMessage};","outputs":1,"x":773,"y":111,"z":"11ea8b41.55b0cd","wires":["824d6856.5bc698"]}],{"id":
"727dfdf1.29cfcc","type":"function","name":"Low Temp","func":"var temperature = msg.payload;\n\nvar
twitMessage = \"LOW temperature Alert @ \"+context.global.device+\"!! The temperature is
\"+temperature+\" deg C at \"+new Date();\n\nreturn {topic: \"\", payload:
twitMessage};","outputs":1,"x":772,"y":278,"z":"11ea8b41.55b0cd","wires":["824d6856.5bc698"]}],{"id":
"824d6856.5bc698","type":"twitter out","twitter":"","name":"Tweet temperature
alerts","x":978,"y":189,"z":"11ea8b41.55b0cd","wires":[]}]
```

The above is a basic node-RED flow which sends an alert to your twitter account if the temperature is either high or low.

- The *IoT device node*, will listen for events sent from the Raspberry Pi/Iot sensor(based on the MAC address provided).
- In the *Extract temperature* function node, the temperature is extracted from the event.
- The *Switch* node bifurcates the events if the temperature is either above 50 deg C or below 10 deg C.
- The final *Twitter* node, it tweets the alerts to the configured twitter account.
- Double click on each node to see its functionality.

3. In the flow editor, you will see



4. Double click on the IoT node (labelled as *IoT device*) to enter the device external ID(MAC address)
 1. If you are using *Raspberry Pi* – Enter the MAC address of the Raspberry Pi which you had noted above. The Raspberry Pi sends CPU temperature.
 2. If you are using *IoT sensor* – Enter the MAC address of the IoT sensor which is present in the top right corner. The IoT sensor sends the simulated temperature.
5. Double click on the twitter node (labelled as *Tweet temperature alerts*), and add your twitter credentials. It will open a new window where you will need to authorize this application to access your account.
6. Finally, click on deploy.

Now the flow is deployed and this flow will tweet if the temperature of Raspberry Pi/IoT sensor goes above 50 deg C or below 10 deg C. If you are using the IoT sensor, change the value of “Temperature” to see the variation in result. (Please note that once the temperature crosses the threshold, it tweets every 2 seconds)

Finally, check your twitter account to see the tweets of the alerts.

Troubleshooting

1. You can wire the debug node in the node-RED to see the messages between the nodes.
2. Twitter allows a fixed number of tweets per user per day, so if it reaches that limit, this application will not be able to tweet anymore.