

## SENDING DATA FROM RASPBERRY-PI TO IBM WATSON

Date	7 NOVEMBER 2022
Team ID	PNT2022TMID54092
Project Name	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

To send sensor data (or any dummy data) from Raspberry –Pi to IBM Watson.

### REQUIREMENTS:

#### HARDWARE:

- RASPBERRY-PI (3B)(WITH ETHERNET CABLE OR WIFI CONNECTED)
- USB MOUSE
- USB KEYBOARD
- VGA TO HDMI CABLE
- A MONITOR
- RASPBERRY'S POWER SUPPLY
- DHT-11 Sensor ➤Connecting Wires

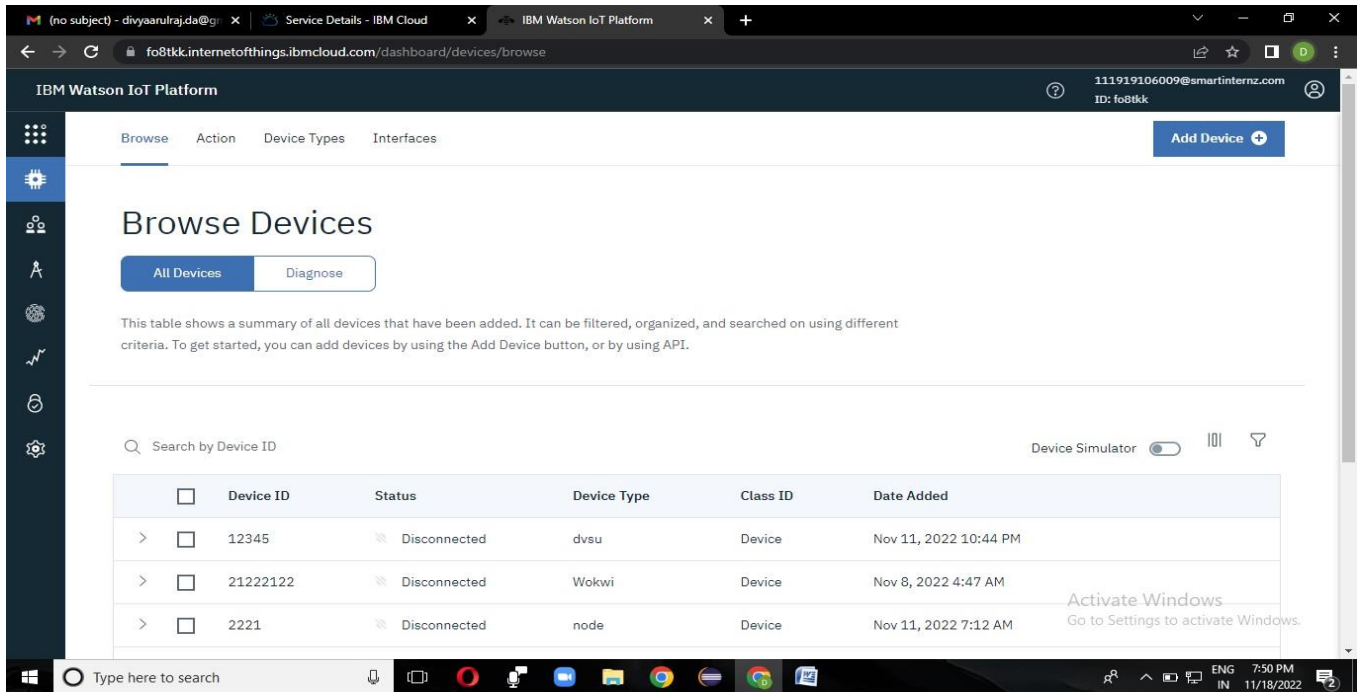
#### SOFTWARE:

- IBM BLUEMIX ACCOUNT

### STEPS TO BE FOLLOWED

#### Step-1: Create a device in IBM Watson:

The screenshot displays the IBM Cloud IoT Platform interface. At the top, the browser address bar shows the URL: [cloud.ibm.com/services/iotf-service/crm%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-de%3AAa%2Fcb61e7f7d2254334852aa31eba0ce5fd%3Ac606b18a-e9b...](https://cloud.ibm.com/services/iotf-service/crm%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-de%3AAa%2Fcb61e7f7d2254334852aa31eba0ce5fd%3Ac606b18a-e9b...). The page header includes the IBM Cloud logo, a search bar, and navigation links for Catalog, Manage, and the user profile (Divyadharshini J A's Ac...). The main content area shows the 'Internet of Things Platform-in' resource list, which is currently 'Active'. Below this, there is a 'Manage' sidebar with options for Plan and Connections. The central part of the page features a large graphic of a network node and a 'Let's get started with IBM Watson IoT Platform' section with a 'Launch' button. Below this, there is a 'Ready for the next level?' section titled 'IBM Watson IoT Platform Journey' with three options: Lite, Non-Production, and Production. The 'Lite' option is selected, and it includes a description: 'The Lite service plan provides a lightweight development environment to get you started'. The 'Non-Production' option is described as 'The Non-Production service plan is a full-featured, fully-integrated offering that enables...'. The 'Production' option is described as 'The Production service is a fully managed SaaS offering that enables you to manage and analyze...'. At the bottom of the page, there is a Windows taskbar with the search bar and several application icons.



## STEP-2: INSTALLING NECESSARY PACKAGES ON YOUR PI

```
File Edit Tabs Help
--2017-10-23 06:55:22-- http://ftp.nl.debian.org/debian/pool/main/o/openssl/lib
ssl1.0.0_1.0.1t-1-deb8u6_armhf.deb
Resolving ftp.nl.debian.org (ftp.nl.debian.org)... 130.89.149.21, 2091:67c:2564:
a120::21
Connecting to ftp.nl.debian.org (ftp.nl.debian.org)[130.89.149.21]:80... connect
ed.
HTTP request sent, awaiting response... 200 OK
Length: 867950 (848K) [application/x-debian-package]
Saving to: 'libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb'

libssl1.0.0_1.0.1t- 100%[=====] 847.61K  358KB/s   in 2.4s

2017-10-23 06:55:25 (358 KB/s) - 'libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb' saved [
867950/867950]

pi@raspberrypi:~$ sudo dpkg -i libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb
Selecting previously unselected package libssl1.0.0:armhf.
(Reading database ... 115606 files and directories currently installed.)
Preparing to unpack libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb ...
Unpacking libssl1.0.0:armhf (1.0.1t-1-deb8u6) ...
Setting up libssl1.0.0:armhf (1.0.1t-1-deb8u6) ...
pi@raspberrypi:~$ curl -LO https://github.com/ibm-messaging/iot-raspberrypi/rel
eases/download/1.0.2/1.0.2-armhf.deb
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   Dload  Upload  Total   Spent    Left    Speed
100 164    0 164    0    157   0  --:--:--  0:00:01 --:--:-- 157
100 609    0 609    0    457   0  --:--:--  0:00:01 --:--:-- 457
100 110k 100 110k    0 29117   0  0:00:03  0:00:03 --:--:-- 48190
pi@raspberrypi:~$ sudo dpkg -i iot_1.0-2_armhf.deb
(Reading database ... 115626 files and directories currently installed.)
Preparing to unpack iot_1.0-2_armhf.deb ...
Unpacking iot (1.0-1) over (1.0-1) ...
Setting up iot (1.0-1) ...
Processing triggers for systemd (232-25+deb9u1) ...
pi@raspberrypi:~$ service iot status
● iot.service - LSB: IoT service
   Loaded: loaded (/etc/init.d/iot; generated; vendor preset: enabled)
   Active: active (running) since Mon 2017-10-23 06:56:25 UTC; 17s ago
     Docs: man:systemd-sys-generator(8)
    CGroup: /system.slice/iot.service
            └─2562 /opt/iot/iot /dev/null

Oct 23 06:56:24 raspberrypi systemd[1]: Starting LSB: IoT service...
Oct 23 06:56:24 raspberrypi iot[2557]: Starting the iot program
Oct 23 06:56:25 raspberrypi iot[2562]: **** IoT Raspberry Pi Sample has started ****
Oct 23 06:56:25 raspberrypi iot[2562]: Config file not found. Going to Quickstart mode
Oct 23 06:56:25 raspberrypi iot[2562]: Running in Quickstart mode
Oct 23 06:56:25 raspberrypi systemd[1]: Started LSB: IoT service.
```

File Edit Tabs Help

```
pi@raspberrypi:~$ pip install ibmiotf
Collecting ibmiotf
  Downloading ibmiotf-0.3.0.tar.gz (58kB)
    100% |#####| 61kB 510kB/s
Collecting dicttoxml>=1.7.4 (from ibmiotf)
  Downloading dicttoxml-1.7.4.tar.gz
Collecting iso8601>=0.1.10 (from ibmiotf)
  Downloading iso8601-0.1.12-py2.py3-none-any.whl
Collecting paho-mqtt>=1.2 (from ibmiotf)
  Downloading paho-mqtt-1.3.1.tar.gz (80kB)
    100% |#####| 81kB 916kB/s
Collecting pytz>=2014.7 (from ibmiotf)
  Using cached pytz-2017.2-py2.py3-none-any.whl
Collecting requests>=2.5.0 (from ibmiotf)
  Downloading requests-2.18.4-py2.py3-none-any.whl (88kB)
    100% |#####| 92kB 1.6MB/s
Collecting requests-toolbelt>=0.7.0 (from ibmiotf)
  Downloading requests-toolbelt-0.8.0-py2.py3-none-any.whl (54kB)
    100% |#####| 61kB 1.6MB/s
Collecting xmldict>=0.10.2 (from ibmiotf)
  Downloading xmldict-0.11.0-py2.py3-none-any.whl
Collecting urllib3<1.23,>=1.21.1 (from requests>=2.5.0->ibmiotf)
  Downloading urllib3-1.22-py2.py3-none-any.whl (132kB)
    100% |#####| 133kB 1.4MB/s
Collecting idna<2.7,>=2.5 (from requests>=2.5.0->ibmiotf)
  Downloading idna-2.6-py2.py3-none-any.whl (58kB)
    100% |#####| 61kB 1.7MB/s
Collecting chardet<3.1.0,>=3.0.2 (from requests>=2.5.0->ibmiotf)
  Downloading chardet-3.0.4-py2.py3-none-any.whl (133kB)
    100% |#####| 143kB 1.6MB/s
Collecting certifi>=2017.4.17 (from requests>=2.5.0->ibmiotf)
  Using cached certifi-2017.7.27.1-py2.py3-none-any.whl
Building wheels for collected packages: ibmiotf, dicttoxml, paho-mqtt
Running setup.py bdist_wheel for ibmiotf ... done
Stored in directory: /home/pi/.cache/pip/wheels/7e/f9/45/bbc33ad957e82f7b71ba80e316d65a83d9d735a0d12e0c0418
Running setup.py bdist_wheel for dicttoxml ... done
Stored in directory: /home/pi/.cache/pip/wheels/45/62/59/96910b33ec6a7b2ae66a13765401b50def5468024078e12cce
Running setup.py bdist_wheel for paho-mqtt ... done
Stored in directory: /home/pi/.cache/pip/wheels/20/d8/0d/acdc8f2890111b7be7de71deebef0642fb83be0313dfff0493
Successfully built ibmiotf dicttoxml paho-mqtt
Installing collected packages: dicttoxml, iso8601, paho-mqtt, pytz, urllib3, idna, chardet, certifi, requests, requests-toolbelt, xmldict, ibmiotf
Successfully installed certifi-2017.7.27.1 chardet-3.0.4 dicttoxml-1.7.4 ibmiotf-0.3.0 idna-2.6 iso8601-0.1.12 paho-mqtt-1.3.1 pytz-2017.2 requests-2.18.4 requests-toolbelt-0.8.0 urllib3-1.22 xmldict-0.11.0
pi@raspberrypi:~$
```

File Edit Shell Debug Options Window Help

```
Python 2.7.13 (default, Jan 19 2017, 14:48:08)
[GCC 6.3.0 20170124] on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/pi/Downloads/dht11toibmiot.py =====
2017-10-23 07:10:37,768 ibmiotf.deviceClient INFO Connected successfully: d:gegtl4:mydevice:mydevice
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson
|
```

### Step-3: checking your data sent on IBM Bluemix:

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. The main content area displays the details for a device with ID 12345, which is currently 'Disconnected'. The device type is 'dvsu' and it was added on 'Nov 11, 2022 10:44 PM'. Below this, there is a section for 'Recent Events' showing a live stream of data. The events are listed in a table with columns: Event, Value, Format, and Last Received. The events are all labeled 'event\_1' and contain JSON data for 'hazardous gas', 'temperature', and 'humidity'. The last received time for all events is 'a few seconds ago'. At the bottom of the dashboard, there is a status bar indicating '2 Simulations running'.

Event	Value	Format	Last Received
event_1	{"hazardous gas":65,"temperature":8,"humidity":...}	json	a few seconds ago
event_1	{"hazardous gas":46,"temperature":54,"humidity":...}	json	a few seconds ago
event_1	{"hazardous gas":84,"temperature":94,"humidity":...}	json	a few seconds ago
event_1	{"hazardous gas":28,"temperature":88,"humidity":...}	json	a few seconds ago
event_1	{"hazardous gas":97,"temperature":17,"humidity":...}	json	a few seconds ago

### Step-4: Creating boards and cards for visualization of data:

The screenshot shows the IBM Watson IoT Platform dashboard with a board titled 'temp & hum'. The board contains three cards: a 'Line chart' card showing a fluctuating line graph for 'hazardous gas' data over time (from 20:14 to 20:18), a 'Value' card displaying '90.0 temperature', and a 'Gauge' card showing a circular gauge with the value '61.0'. The dashboard also includes a navigation bar with 'Add New Card', 'Paste Card', and 'Settings' buttons. At the bottom, there is a status bar indicating '2 Simulations running'.

