

Connect a Raspberry Pi to IBM Watson IoT Platform

Ingredients

Hardware

1. Raspberry Pi Model B/ Model B+/ 2
2. Minimum 8GB SD card with Raspbian operating system

Software

1. Bluemix Account
2. Watson IoT Platform Service
3. Node-RED Application

Step-by-step

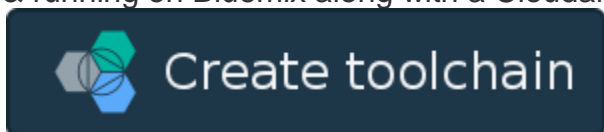
1. Get your Device and Bluemix setup ready

Get your Device ready:

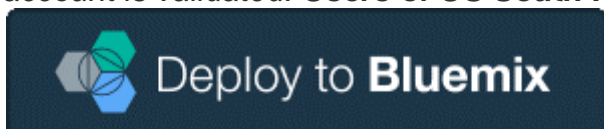
Connect your **Raspberry Pi** to the internet. Ensure you are able to reach (Ping) and connect to it.

Getting the Bluemix services ready:

This IoT Recipe makes use of the [Create Toolchain](#) button to help deploy the necessary services on to Bluemix. Click on the [Create Toolchain](#) button provided below, provide a custom name to your application and choose to click on Create button, to quickly deploy the Watson IoT Platform – Node-RED Bluemix starter app. Post successful deployment, you shall have both the above mentioned services, up & running on Bluemix along with a Cloudant NoSQL DB at your disposal.



Note: If you are a **User** using the **United Kingdom Region** in your Bluemix environment, then please make use of the steps mentioned in the IoT Recipe [Deploy Internet of Things Platform Starter service on Bluemix](#) to deploy the setup. Alternatively, you can also try using the [Deploy to Bluemix](#) button, to deploy the setup under your **United Kingdom Region**, provided your Jazzhub account is validated. **Users of US South Region can ignore this step.**



2. Connect (Quickstart)

Watson IoT Platform Quickstart connection

1. Download the installer from GitHub:

```
curl -LO https://github.com/ibm-messaging/iot-raspberrypi/releases/download/1.0.2.1/iot_1.0-2_armhf.deb
```

([Source code](#) is also available)

2. Install the package with:

```
sudo dpkg -i iot_1.0-2_armhf.deb
```

3. The iot process starts and publishes events to the IBM Watson IoT Platform Quickstart. The process runs as a system service, and starts whenever Raspberry Pi starts. To verify the process is running, use the command:

```
service iot status
```

4. You can stop the iot process with the following command:

```
sudo service iot stop
```

5. To uninstall the package, run:

```
sudo dpkg -P iot
```

*Note: For troubleshooting, go to `/var/log/syslog` to find more information about the **iot** process.*

3. Visualize

Real-time visualization of device data

1. Find the MAC address of your Raspberry Pi by running the following service command:

```
service iot getdeviceid
```

Enter your [MAC address here](#) to visualize your data.

4. Register your Device In Watson IoT Platform

To explore and make use of full capabilities of Watson IoT Platform, we need to setup and register your device(s) in it. This section shows how you can setup the same.

Open the Bluemix Dashboard and click on the Application service that you have currently deployed (using the [Create Toolchain](#) button), if you have already moved away from it. You should see the Watson IoT Platform listed under the Connections. Click on the WIoT Platform service and choose the click on Launch button to launch the WIoT Platform Dashboard.

Carry out the [steps present in this recipe](#) to register your device(s) in IBM Watson Internet of Things Platform.

Organization ID	
Device Type	iotsample-devicetype
Device ID	Device01
Authentication Method	token
Authentication Token	

At this step, we have successfully created the Watson IoT service and registered your device(s) in it.

5. Connect (Registered)

Watson IoT Platform Registered connection

1. During the device registration process you will get file configuration information containing the following details, copy these when you get them.

Organization ID

Device Type ID

Device ID
Authentication Method
Authentication Token

2. Stop the iot process

```
sudo service iot stop
```

3. Type in the following command

```
sudo nano /etc/iotsample-raspberrypi/device.cfg
```

4. Create a new **device.cfg** file at **/etc/iotsample-raspberrypi** fill in the configuration file with the details that you copied when registering the device.

```
#Device configuration file
org = yourOrganizationCode
type = iotsample-raspberrypi
id = b827eba84426
auth-method = token
auth-token = yourAuthToken
#End of Configuration file
```

Note: All the properties in the configuration file are mandatory. Currently, only token based authentication is supported.

5. Save your file by pressing **CTRL-X** in the Nano editor and follow the steps on the screen.
6. Start the iot process for the device to start in registered mode

```
sudo service iot start
```

6. Visualize the Raspberry Pi events in Watson IoT Platform dashboard

With the new boards and cards capability in the Watson Internet of Things platform, you can build your own Custom dashboard **without writing any code**. You can use the boards as the landing page of interest and then make use of the cards within them to:

- Create visualization charts for the real time data from your devices
- Create Gauges for visualizing physical quantities like Vehicle Speed, Temperature, pressure
- Create Donuts charts, bar charts to display the current value of the data points
- See the Data and storage consumption of your devices
- List of registered devices and etc..

Refer to [Configuring Boards & Cards in the new Watson IoT Dashboard](#) recipe to view the Raspberry Pi events in Watson IoT Platform using the boards and cards.

7. Access the Node-RED application to receive events from the device

With events now being sent to Watson IoT, the next step is to access the application in Bluemix to receive the events.

Access the Node-RED application in Bluemix

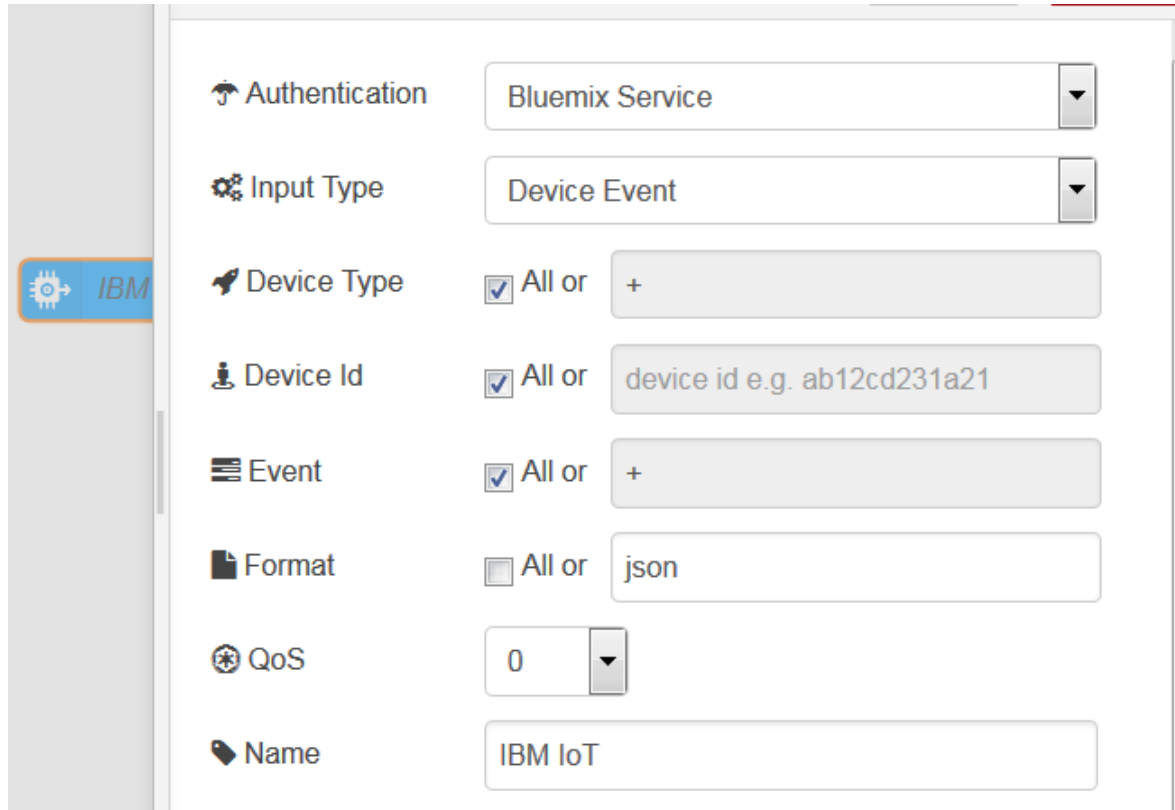
0. You have deployed the Node-RED service on to the Bluemix by clicking on the [Create Toolchain](#) button in Pre-requisites section. Access your Node-RED application by clicking on **View App** link and then choosing the option *Go to your Node-RED flow editor*, to access the editor.
1. Parallely, you can also access the Node-RED editor directly by accessing the following URL

<http://<your-application-name>.mybluemix.net/red>

Note: The Node-RED Editor opens up with a default flow. Click on the + sign to add a new tab and continue with the steps mentioned here after.

Create a simple flow to start receiving the device events

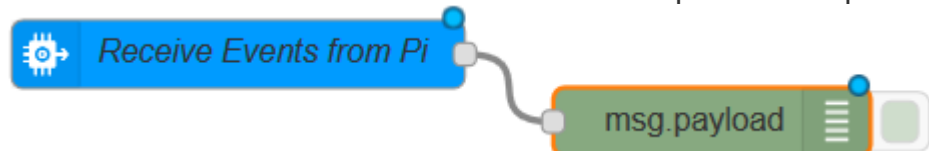
- Open the Node-RED flow editor,
- Add an **ibmiot input** node and configure it to use **Bluemix Service** authentication. Set the **Device Type** field to match the type you used to register your device with. Leave the **Device Id** and **Event** fields set to all.



The screenshot shows the configuration interface for the 'IBM IoT' node in Node-RED. The configuration is as follows:

- Authentication:** Bluemix Service
- Input Type:** Device Event
- Device Type:** All or +
- Device Id:** All or device id e.g. ab12cd231a21
- Event:** All or +
- Format:** All or json
- QoS:** 0
- Name:** IBM IoT

- Attach the **debug** node to the IBM IoT node as shown below to output the datapoints



in the debug panel.

- Click **Deploy** to deploy the flow in Node-RED and observe that the readings are printed in the console.

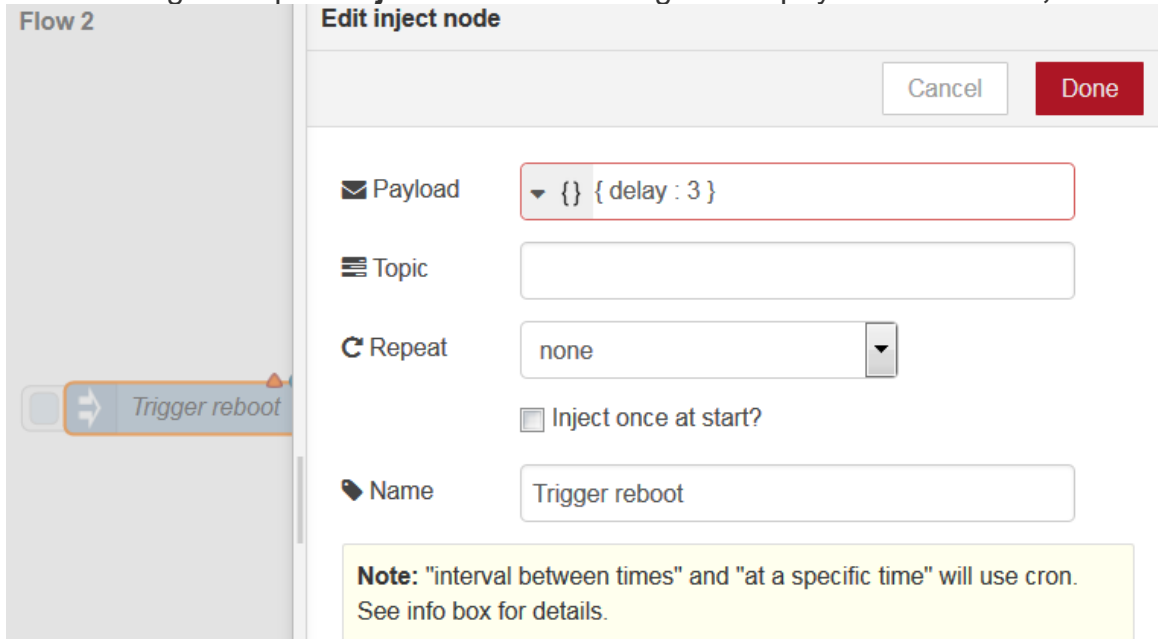
In this step, we have successfully configured the IBM IoT App In Node to receive the Raspberry Pi events from the Watson IoT Platform.

8. Send commands to your Raspberry Pi device

When the Raspberry Pi Sample is running in registered mode, it supports the receipt of Commands sent by an Application.

This sample supports the Reboot Command. When the user sends this command, the Raspberry Pi will reboot based on the seconds provided in the payload.

- Drag-n-Drop an **inject node** and configure the payload as follows,



Flow 2

Edit inject node

Cancel Done

✉ Payload

📄 Topic

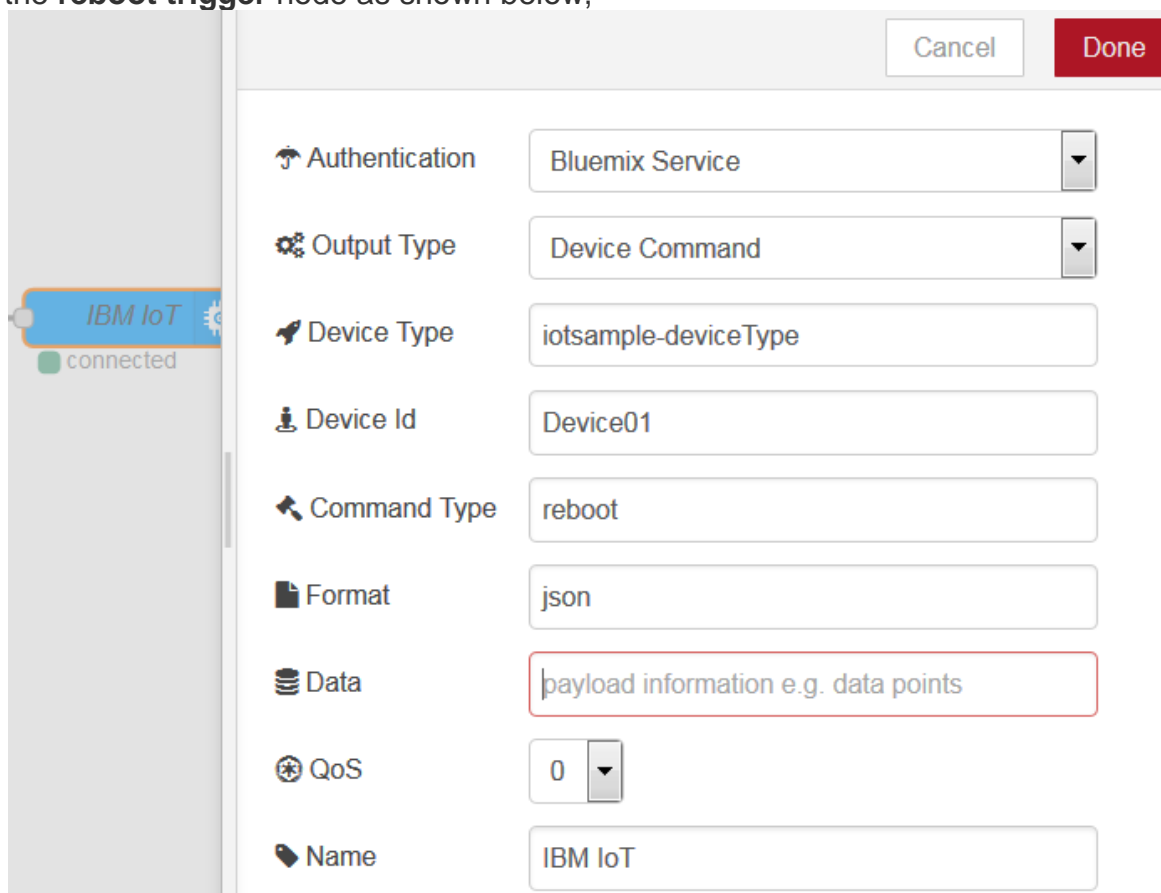
🔄 Repeat

☐ Inject once at start?

🔖 Name

Note: "interval between times" and "at a specific time" will use cron. See info box for details.

- As shown the above payload instructs the sample to wait for 3 seconds and reboot the device.
- Drag and drop **ibmiot** out node from the output pane and configure it to use **Bluemix Service** authentication. Set the Device Type, Device Id fields appropriately. Set the **Output type as Device Command** and connect it to the **reboot trigger** node as shown below,



Cancel Done

☂ Authentication

⚙️ Output Type

🔗 Device Type

👤 Device Id

🔑 Command Type

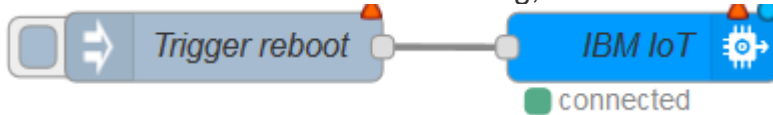
📄 Format

📄 Data

🌐 QoS

🔖 Name

- The flow should look like the following,



- Click **deploy** to deploy the flow.
 - Click on the Trigger reboot node to send the reboot command to Raspberry Pi device.
In this step, we saw how to send a command to the device to control it using a Node-RED application.
9. Where to go from here?

As a next step, go through the following recipes that will help you to create an end to end IoT solution,

- [Real-time data analytics in Watson IoT Platform](#) – Showcases how to create rules and actions to monitor assets in real-time with **just few mouse clicks. There is no coding involved.**
- Go through [How to connect Raspberry Pi as managed device to Watson IoT Platform](#) to take advantage of device management capabilities present in Watson IoT Platform
- Go through [Connect your device as Gateway to Watson IoT Platform](#) recipe to connect your gateway and devices behind the Gateway to IBM Watson IoT Platform
- Go through [Configure Cloudant NoSQL DB as Historian Data Storage for IBM Watson IoT](#) to store your device events in Cloudant NoSQL DB **without writing any code.**

Share this: