

Quickstart - LWM2M to Azure IoT Hub

Quickly integrate LWM2M devices to Azure IoT Hub

LWM2M to Azure IoT Hub

This guide will walk you through integrating a LWM2M device to Azure IoT Hub using an available solution template.



Pre-requisites

1. Account on Tartabit IoT Bridge.
2. Access to a Microsoft Azure subscription.
3. LWM2M device that supports registration or bootstrap with CoAP over DTLS.

1. Create an IoT Hub in Azure

If you already have an IoT Hub, you can skip this step.

For a detailed walkthrough from Microsoft, check out the following link:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-create-through-portal> [↗](#)

Microsoft Azure

[Home](#) > [New](#) > [IoT Hub](#) >

IoT hub

Microsoft

Basics Networking Size and scale Tags Review + create

Create an IoT hub to help you connect, monitor, and manage billions of your IoT assets. [Learn more](#)

Project details

Choose the subscription you'll use to manage deployments and costs. Use resource groups like folders to help you organize and manage resources.

Subscription * ⓘ

Resource group * ⓘ [Create new](#)

Region * ⓘ

IoT hub name * ⓘ

A **B** **C**

[Review + create](#) [< Previous](#) [Next: Networking >](#) [Automation options](#)

- ▶ A: Select an existing resource group or create one.
- ▶ B: Provide a unique name for your IoT Hub.
- ▶ C: You can skip the other tabs and create a new Hub.

2. Retrieve your IoT Hub connection string

1. Navigate to your new Azure IoT Hub.

2. Click **Shared Access Policies**.

The screenshot shows the Microsoft Azure portal interface. At the top, there is a search bar and navigation options. The main content area displays the 'Shared access policies' settings for the 'tartabit-demo' IoT Hub. The 'Shared access policies' option in the left-hand navigation menu is highlighted with a red box. The main content area shows an 'Essentials' section with details such as Resource group (demo), Status (Active), Current location (East US 2), Subscription (Development), and Subscription ID (4c497868-a5fb-4fee-9f8d-fb1bddcef30c). A warning banner at the top right indicates that Azure IoT Hub and the Azure Device Provisioning Service are updating their TLS certificates. A promotional banner at the bottom right asks 'Need a way to provision millions of devices?'.

3. Select **iothubowner** (this is required because the IoT Bridge will automatically provision devices in the IoT Hub as they are needed).

The screenshot shows the 'Shared access policies' table in the Azure portal. The table has two columns: 'Policy' and 'Permissions'. The 'iothubowner' policy is highlighted with a red box, and its permissions are 'registry write, service connect, device connect'. Other policies listed include 'service', 'device', 'registryRead', and 'registryReadWrite'.

Policy	Permissions
iothubowner	registry write, service connect, device connect
service	service connect
device	device connect
registryRead	registry read
registryReadWrite	registry write

4. Copy the **Connection string - primary key** value and save it for later.

iothubowner
tartabit-demo



 Save  Discard  Regenerate keys ...

Access policy name

iothubowner

Permissions

- Registry read ⓘ
- Registry write ⓘ
- Service connect ⓘ
- Device connect ⓘ

Shared access keys

Primary key ⓘ



Secondary key ⓘ



Connection string—primary key



Connection string—secondary key



3. Prepare a LWM2M device

If you have a LWM2M device, you will need to save the following information for later:

- ▶ Endpoint
- ▶ DTLS Identity (as a string)
- ▶ DTLS Pre-shared key (Hex encoded)

If you don't have a LWM2M device, you can use the Anjay open source client from AVSystem. If you already have a LWM2M device, you can skip this step.

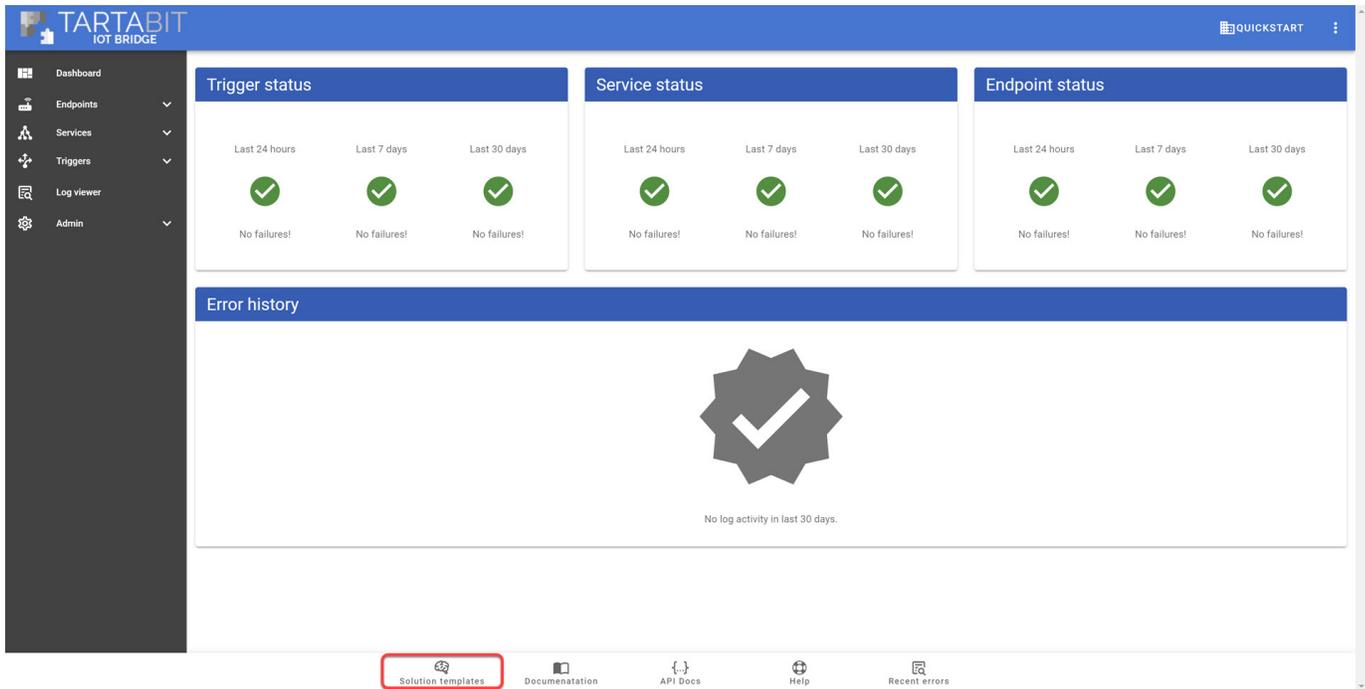
Go to <https://github.com/AVSystem/Anjay>  to find the instructions to build and run the Anjay client on your PC. We will cover the required command line arguments in a later step.

You can choose your own endpoint, identity and pre-shared key at this point, or use the randomly generated values as part of the import.

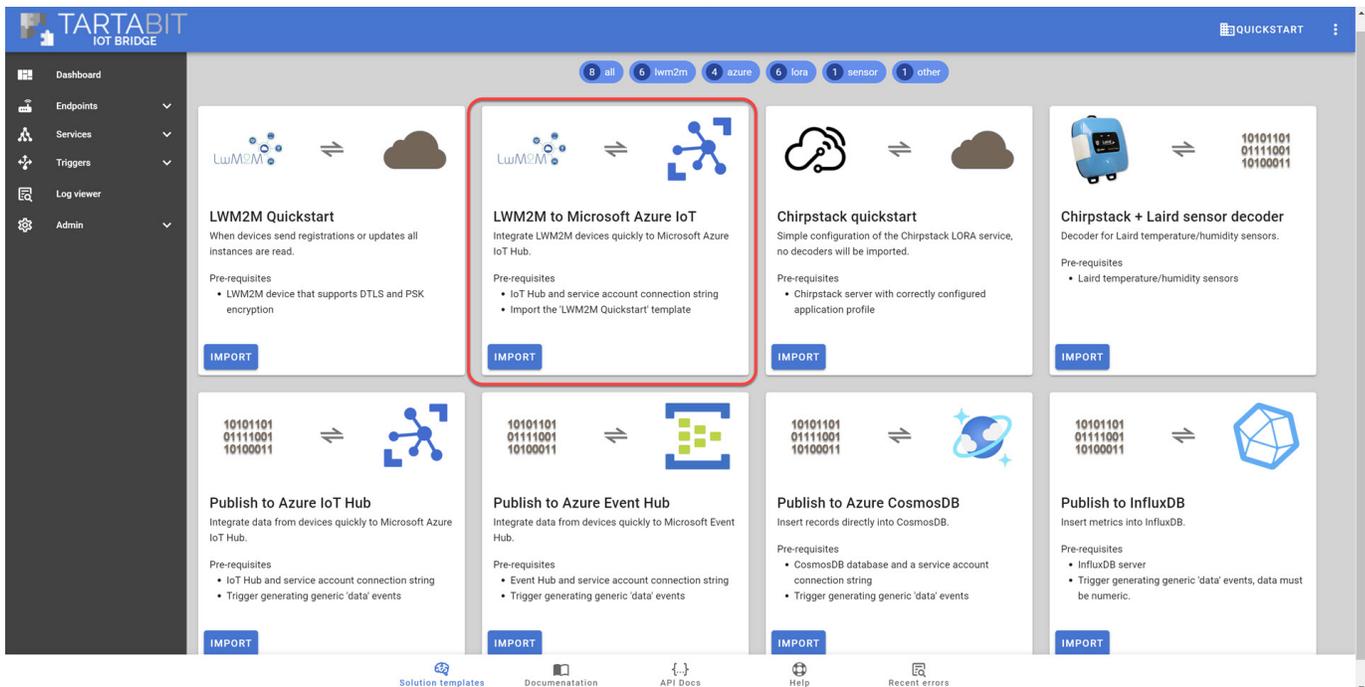
4. Import the solution template

Follow the steps below to import the solution template:

1. Click **Solution Templates** at the bottom of any page.



2. Find the **LWM2M to Microsoft Azure IoT Hub** template and click **Import**.



3. Follow the instructions below to import the template:

- ▶ A: If you know the endpoint, identity, and pre-shared key, enter it here. If you do not, save these values for configuring your client.
 - ▶ For the Anjay client, the identity is specified in HEX, and you will need to hex-encode the value generated in the import dialog before using it in the Anjay client.
- ▶ B: Enter the Azure IoT Hub connection string that you previously saved in Step #1.

- ▶ C: Select the option to **Start Triggers** this will automatically start all of the new triggers upon import.
- ▶ D: Click import when finished.

Import

LWM2M endpoint
fq05CFvMWA1Tsm1i

LWM2M identity
btUsiFONk3ayhyMA

LWM2M Pre-shared key
4C865640075F9651

LWM2M lifetime
60

Azure connection string
HostName=***.azure-devices.net;SharedAccessKeyName=iotechowner;SharedAccessKey=

Name	Type	Result
LWM2M Server	Service	<input type="radio"/>
Azure IoT Hub	Service	<input type="radio"/>
LWM2M Device	Endpoint	<input type="radio"/>
LWM2M read on update	Trigger	<input type="radio"/>
LWM2M read on register	Trigger	<input type="radio"/>
LWM2M write on twin update	Trigger	<input type="radio"/>
LWM2M send data to twin	Trigger	<input type="radio"/>

Start triggers

CANCEL IMPORT

5. Verify the import was successful

You should receive an import result like this:

Import

Name	Type	Result
LWM2M Server	Service	✓
Azure IoT Hub	Service	✓
LWM2M Device	Endpoint	✓
LWM2M read on update	Trigger	✓
LWM2M read on register	Trigger	✓
LWM2M write on twin update	Trigger	✓
LWM2M send data to twin	Trigger	✓

CLOSE

6. Connect your LWM2M device

Now that everything has been imported, you can connect your first LWM2M device.

If you have your own device and entered the credentials in step #4, then you can connect it now.

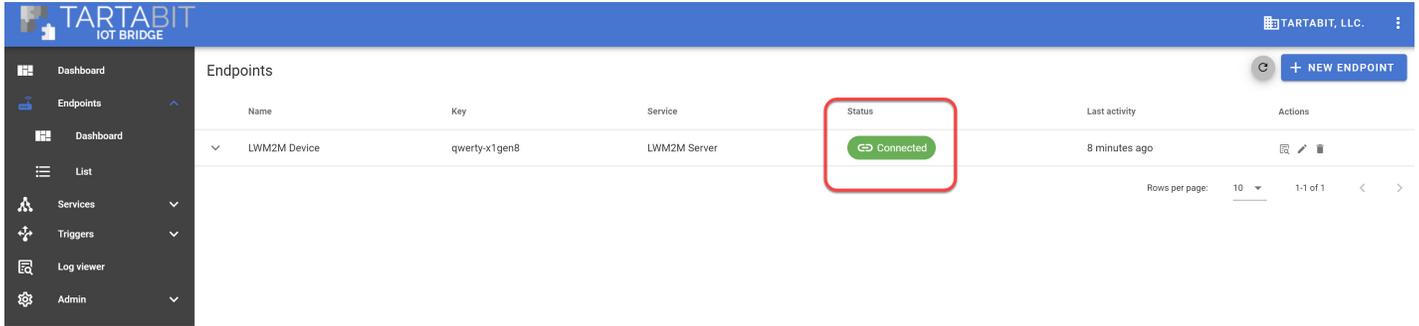
If you are using the Anjay client, run the client with the credentials generated during the import phase, or that you manually entered. Remember that the identity must be hex encoded.

```
./output/bin/demo --endpoint-name <endpoint> --server-uri
```

```
coaps://bs.tartabit.com:5684 --security-mode psk --identity <hex encoded identity> --key <pre-shared key>
```

You should see that your client reports connected.

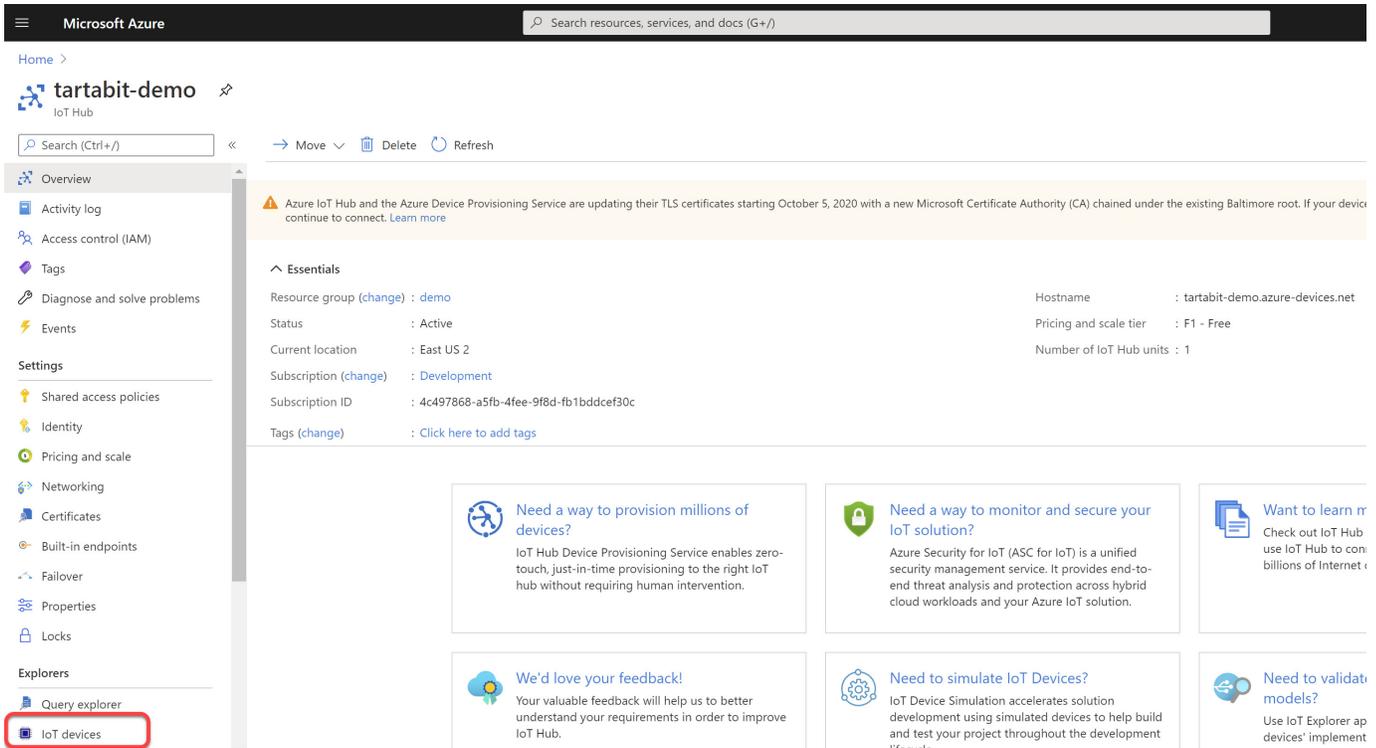
Verify that the client is connected by navigating to **Endpoints -> List** and checking the status.



7. Check your device in Azure IoT Hub

With the device connected, you should now see it connected in the IoT Hub, and data being updated in the device twin.

- 1. Navigate to your Azure IoT Hub
- 2. Click IoT Devices



3. You should see your newly created device, click on it.

The screenshot shows the Microsoft Azure IoT Hub interface for a resource named 'tartabit-demo'. The left sidebar contains navigation options such as Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Events, Settings, and Explorers. The main area displays a query editor with a clause: 'deviceid = 'qwerty-x1gen8''. Below the query editor is a table of IoT devices. The table has columns for DEVICE ID, STATUS, LAST STATUS UPDATE (UTC), AUTHENTICATION TYPE, and CLOUD TO DEVICE MESSAGE COUNT. One device, 'qwerty-x1gen8', is listed with a status of 'Enabled' and 0 messages. The device ID 'qwerty-x1gen8' is highlighted with a red box.

DEVICE ID	STATUS	LAST STATUS UPDATE (UTC)	AUTHENTICATION TYPE	CLOUD TO DEVICE MESSAGE COUNT
qwerty-x1gen8	Enabled	--	Sas	0

4. Click on Device Twin

Microsoft Azure Search resources, services, and docs (G+)

Home > tartabit-demo >

qwerty-x1gen8

tartabit-demo

Save Message to Device Direct Method Add Module Identity **Device Twin** Manage keys Refresh

Device ID: qwerty-x1gen8

Primary Key:

Secondary Key:

Primary Connection String:

Secondary Connection String:

Enable connection to IoT Hub: Enable Disable

Parent device: No parent device

Module Identities Configurations

MODULE ID	CONNECTION STATE	CONNECTION STATE LAST UPDATED (U...	LAST ACTIVITY TIME (UTC)
There are no module identities for this device.			

5. You will see the LWM2M object data in your twin.

Microsoft Azure

Search resources, services, and docs (G+)

Home > tartabit-demo > qwerty-x1gen8 >

Device twin

qwerty-x1gen8

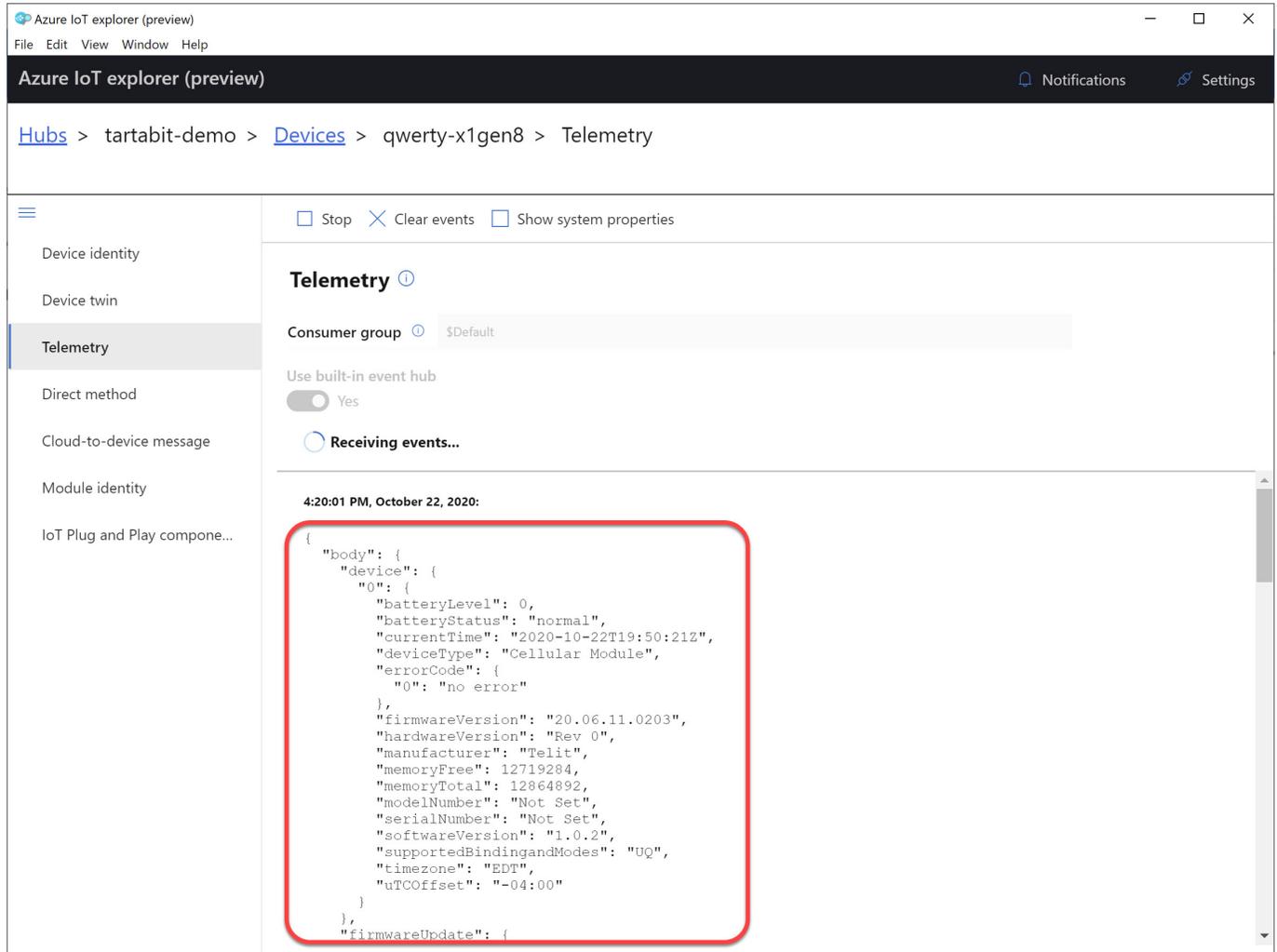
Save Refresh

The device twin for 'qwerty-x1gen8' is shown below. You can add tags and desired properties to your device twin here. To remove a tag or desired

```
},
"reported": {
  "device": {
    "0": {
      "batteryLevel": 0,
      "batteryStatus": "normal",
      "currentTime": "2020-10-08T15:12:19Z",
      "deviceType": "Cellular Module",
      "errorCode": {
        "0": "no error"
      },
      "firmwareVersion": "20.06.11.0203",
      "hardwareVersion": "Rev 0",
      "manufacturer": "Telit",
      "memoryFree": 12669152,
      "memoryTotal": 12864892,
      "modelName": "Not Set",
      "serialNumber": "Not Set",
      "softwareVersion": "1.0.2",
      "supportedBindingandModes": "UQ",
      "timezone": "EDT",
      "uTCOffset": "-04:00"
    }
  },
  "firmwareUpdate": {
    "0": {
      "firmwareUpdateDeliveryMethod": 2,
      "firmwareUpdateProtocolSupport": {
        "0": 0,
        "1": 1,
        "2": 2,

```

6. You can use the Azure IoT Explorer to view the telemetry as well.



The screenshot shows the Azure IoT Explorer (preview) interface. The breadcrumb navigation is [Hubs](#) > [tartabit-demo](#) > [Devices](#) > [qwerty-x1gen8](#) > [Telemetry](#). The left sidebar contains a menu with options: Device identity, Device twin, **Telemetry**, Direct method, Cloud-to-device message, Module identity, and IoT Plug and Play compone... The main content area has controls for Stop, Clear events, and Show system properties. Below these is the **Telemetry** section with a consumer group dropdown set to '\$Default'. A toggle for 'Use built-in event hub' is set to 'Yes'. A status indicator shows 'Receiving events...'. The telemetry data is displayed as a JSON object, with the main body of the object highlighted by a red box:

```
4:20:01 PM, October 22, 2020:
{
  "body": {
    "device": {
      "0": {
        "batteryLevel": 0,
        "batteryStatus": "normal",
        "currentTime": "2020-10-22T19:50:21Z",
        "deviceType": "Cellular Module",
        "errorCode": {
          "0": "no error"
        },
        "firmwareVersion": "20.06.11.0203",
        "hardwareVersion": "Rev 0",
        "manufacturer": "Telit",
        "memoryFree": 12719284,
        "memoryTotal": 12864892,
        "modelName": "Not Set",
        "serialNumber": "Not Set",
        "softwareVersion": "1.0.2",
        "supportedBindingandModes": "UQ",
        "timezone": "EDT",
        "uTCOffset": "-04:00"
      }
    },
    "firmwareUpdate": {
```

You are done!

© 2020 Tartabit, LLC.. All rights reserved. | Powered by [Wiki.js](#)