FIWARE - IoT over MQTT Tutorial

# Code repository

[FIWARE/tutorials.IoT-over-MQTT: :orange\_book: FIWARE 203: Provisioning Ultralight with an alternative transport: IoT over MQTT (github.com)](https://github.com/FIWARE/tutorials.IoT-over-MQTT)

Other tutorials

[FIWARE/tutorials.NGSI-LD at 06cfecb4beaa270eeb2c1165276773073c338d81 (github.com)](https://github.com/FIWARE/tutorials.NGSI-LD/tree/06cfecb4beaa270eeb2c1165276773073c338d81)

# Deployment

## Start Up

[FIWARE/tutorials.IoT-over-MQTT: :orange\_book: FIWARE 203: Provisioning Ultralight with an alternative transport: IoT over MQTT (github.com)](https://github.com/FIWARE/tutorials.IoT-over-MQTT#start-up)

Note: Please ensure that you are using Docker version 20.10 or higher and Docker Compose 1.29 or higher and upgrade if necessary.

To check your current **Docker** and **Docker Compose** versions using the following commands

docker-compose -v  
docker version

git clone https://github.com/FIWARE/tutorials.IoT-over-MQTT.git

cd tutorials.IoT-over-MQTT

git checkout NGSI-v2

./services create

./services start

## Checking Mosquitto Health

Use the available docker container

[FIWARE/tutorials.IoT-over-MQTT: :orange\_book: FIWARE 203: Provisioning Ultralight with an alternative transport: IoT over MQTT (github.com)](https://github.com/FIWARE/tutorials.IoT-over-MQTT#start-up)

docker run -it --rm --name mqtt-subscriber \

--network fiware\_default efrecon/mqtt-client sub -h mosquitto -t "/#"

Other option is to install mosquitto client

[Download | Eclipse Mosquitto](https://mosquitto.org/download/)

mosquitto\_sub.exe -t '#' -v

## Mosquitto Security

See chapter “Certificate based SSL/TLS support” in the configuration file

[tutorials.IoT-over-MQTT/mosquitto/mosquitto.conf at master · FIWARE/tutorials.IoT-over-MQTT (github.com)](https://github.com/FIWARE/tutorials.IoT-over-MQTT/blob/master/mosquitto/mosquitto.conf)

Useful links

[Mosquitto Docker Configuration - Ultimate Guide | Cedalo](https://cedalo.com/blog/mosquitto-docker-configuration-ultimate-guide/)

# Standard installation

sudo mosquitto\_passwd -c /etc/mosquitto/passwd username

Enter new password

sudo nano /etc/mosquitto/conf.d/default.conf

Add the following lines in the file

allow\_anonymous false

password\_file /etc/mosquitto/passwd

sudo systemctl restart mosquitto

# Experiments

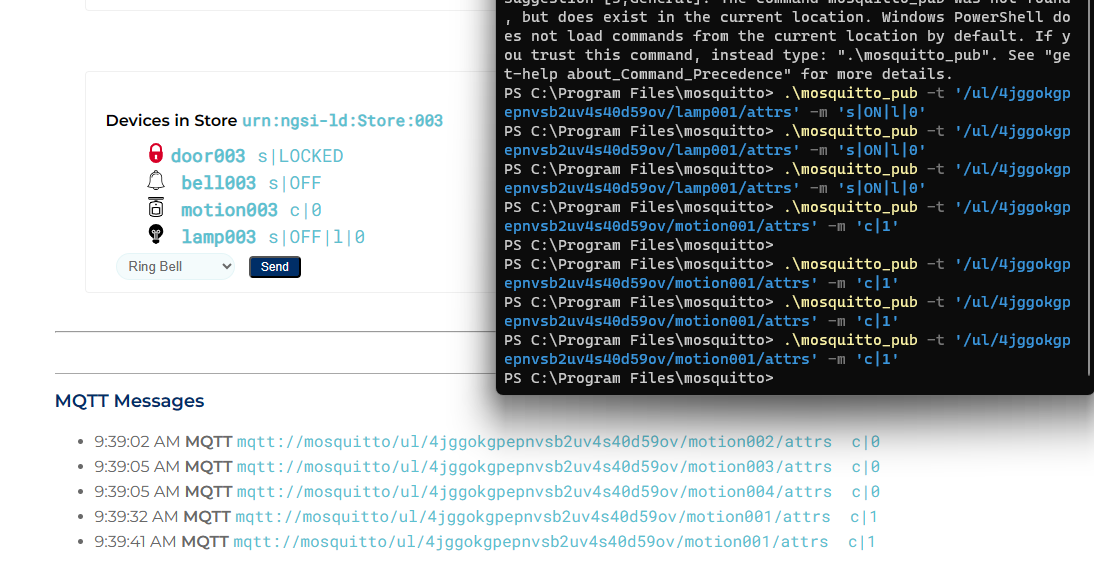
For the purpose of this tutorial, a series of dummy IoT devices have been created, which will be attached to the context broker. Details of the architecture and protocol used can be found in the [IoT Sensors tutorial](https://github.com/FIWARE/tutorials.IoT-Sensors/tree/NGSI-v2)

The state of each device can be seen on the UltraLight device monitor web page found at:

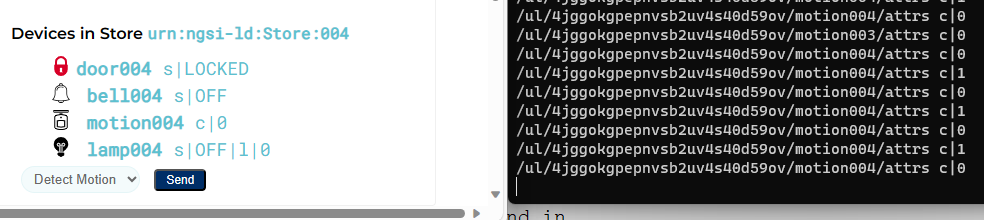
<http://localhost:3000/device/monitor>

konqueror <http://localhost:3000/device/monitor>

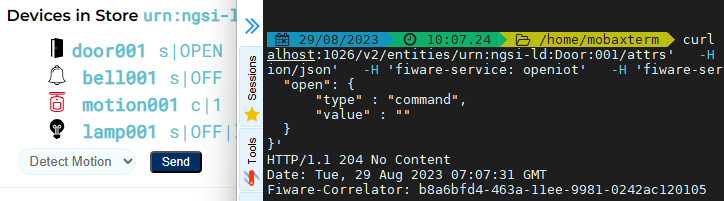
mosquitto\_pub.exe -h 62.44.125.136 -t "/ul/4jggokgpepnvsb2uv4s40d59ov/motion001/attrs" -m "c|1"



MQTT messages are send and shown on the GUI (MQTT Messages)



CURL requests also work



Provisioning a Service Group for MQTT

Note Measures and commands are sent over different MQTT topics:

* *Measures* are sent on the /<protocol>/<api-key>/<device-id>/attrs topic
* *Commands* are sent on the /<api-key>/<device-id>/cmd topic

mosquitto\_pub -t '/ul/4jggokgpepnvsb2uv4s40d59ov/motion001/cmd' -m 's|1'

For some devices provisioning is required first, for example smart lamp

curl -iX POST \

'http://localhost:4041/iot/devices' \

-H 'Content-Type: application/json' \

-H 'fiware-service: openiot' \

-H 'fiware-servicepath: /' \

-d '{

"devices": [

{

"device\_id": "lamp001",

"entity\_name": "urn:ngsi-ld:Lamp:001",

"entity\_type": "Lamp",

"protocol": "PDI-IoTA-UltraLight",

"transport": "MQTT",

"apikey": "4jggokgpepnvsb2uv4s40d59ov",

"commands": [

{"name": "on","type": "command"},

{"name": "off","type": "command"}

],

"attributes": [

{"object\_id": "s", "name": "state", "type":"Text"},

{"object\_id": "l", "name": "luminosity", "type":"Integer"}

],

"static\_attributes": [

{"name":"refStore", "type": "Relationship","value": "urn:ngsi-ld:Store:001"}

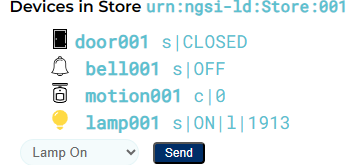
]

}

]

}

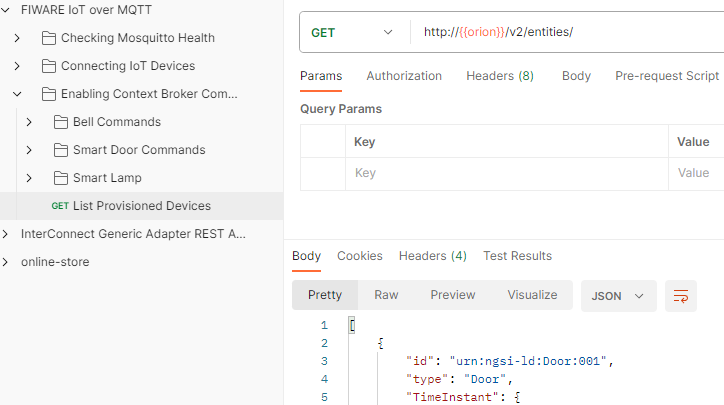
'

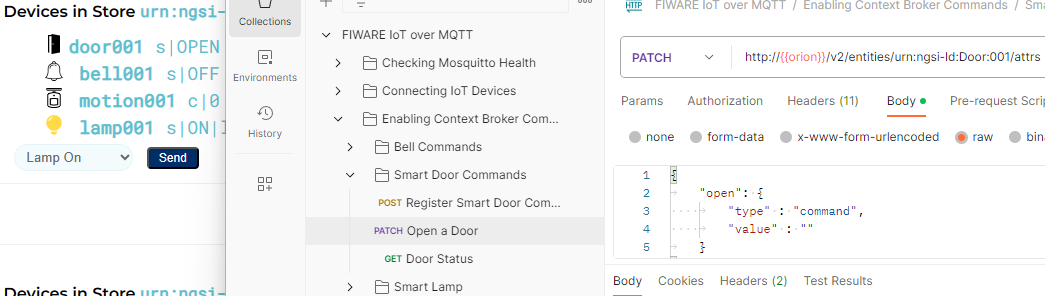


# Other useful

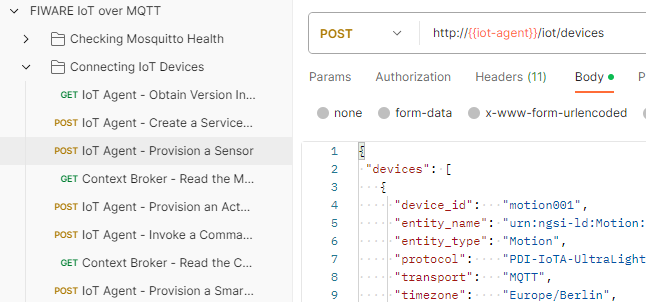
Postman collection

[tutorials.IoT-over-MQTT/FIWARE IoT over MQTT.postman\_collection.json at master · FIWARE/tutorials.IoT-over-MQTT (github.com)](https://github.com/FIWARE/tutorials.IoT-over-MQTT/blob/master/FIWARE%20IoT%20over%20MQTT.postman_collection.json)





Provisioning a device



# Questions

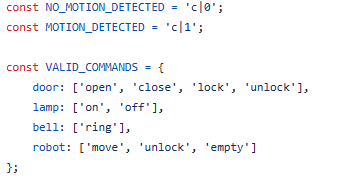
Where is the code of the dummy devices?

Probably in the tutorial container



https://github.com/fiware/tutorials.NGSI-v2

[tutorials.NGSI-v2/context-provider/models/devices.js at master · FIWARE/tutorials.NGSI-v2 (github.com)](https://github.com/FIWARE/tutorials.NGSI-v2/blob/master/context-provider/models/devices.js)



[fiware/tutorials.ngsi-ld - Docker Image | Docker Hub](https://hub.docker.com/r/fiware/tutorials.ngsi-ld)

Useful links

[What is Linked Data? - YouTube](https://www.youtube.com/watch?v=4x_xzT5eF5Q&ab_channel=ManuSporny)

[What is JSON-LD? - YouTube](https://www.youtube.com/watch?v=vioCbTo3C-4&ab_channel=ManuSporny)

[Wednesday Webinar: Introduction to NGSI-LD - YouTube](https://www.youtube.com/watch?v=rZ13IyLpAtA&ab_channel=FIWARE)