WoMaster

WoMaster

WR322-EC and LR140
Quick start & OTA integration

Brian

RS232 and RS485 connection on LR140

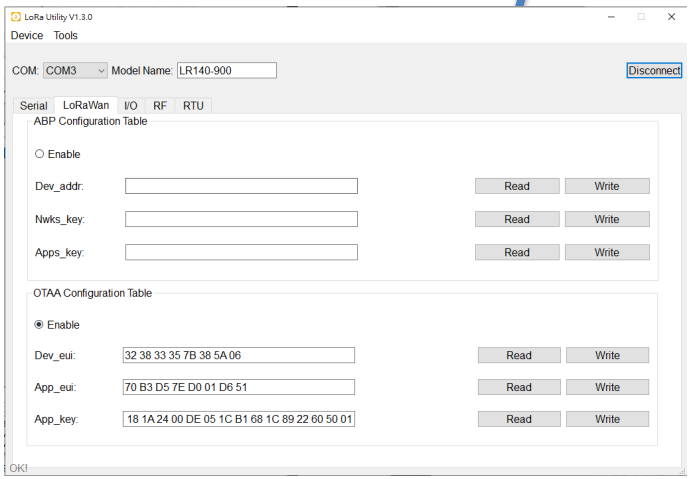
RS485A

RS485B

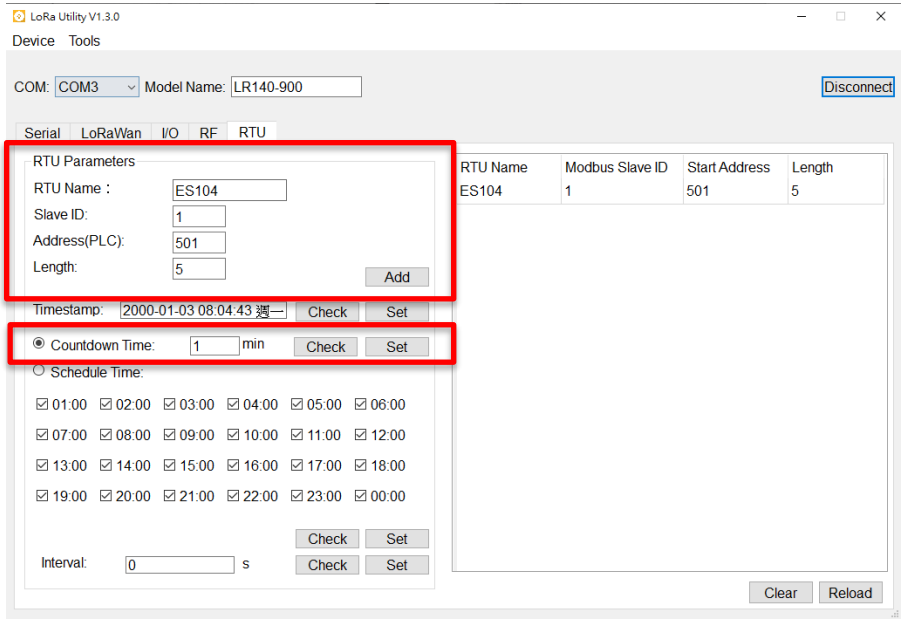
ES104

USB-To-RS232

LoRa Windows Utility



RTU settings on LR140



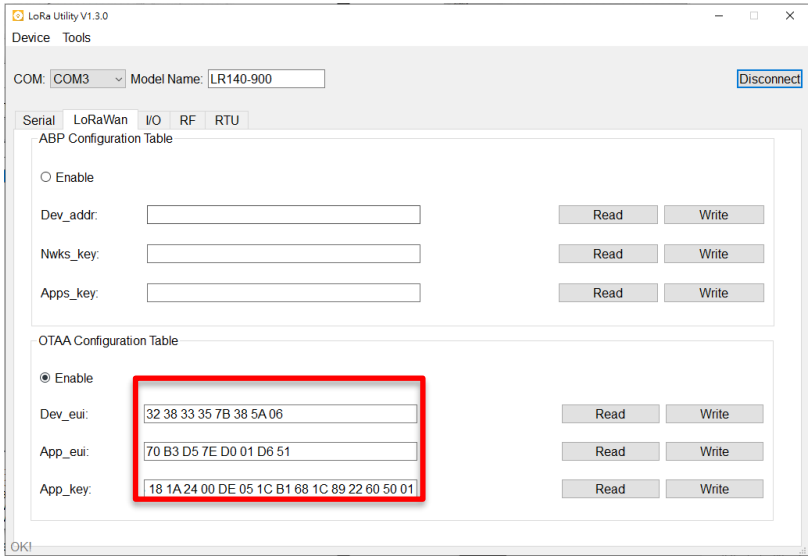
RTU -> RTU Parameters -> Enter Sensor Slave ID, Address , Length -> Add

Example: ES104, Slave ID :1, Address(PLC):501 , Length: 5

Select Countdown Time -> 1 minutes -> set (set LR140 every 1 minute to send RTU data to LoRaWAN)

ES104 Modbus Register Information				
Default slave ID	Parameters Function	Register Add. (HEX / DEC)	PLC Add. (Modbus Add.)	Note
	Device ID Storage Add.	07D0H / 2000	2001 (42001)	R/W , Range:1~254
	Serial Baud Rate Add.	07D1H / 2001	2002 (42002)	R/W , Default: 2 2(9600), 0(2400), 1(4800)
1	Humidity	01F4H / 500	501(40501)	R/O, Real Value = Read Value /10
1	Temperature	01F5H / 501	502(40502)	R/O, Real Value = Read Value /10
1	PM2.5	01F7H / 503	504(40504)	R/O, Real Value = Read Value
1	PM10	01F8H/ 504	505 (40505)	R/O, Real Value = Read Value

LoRaWAN settings on LR140



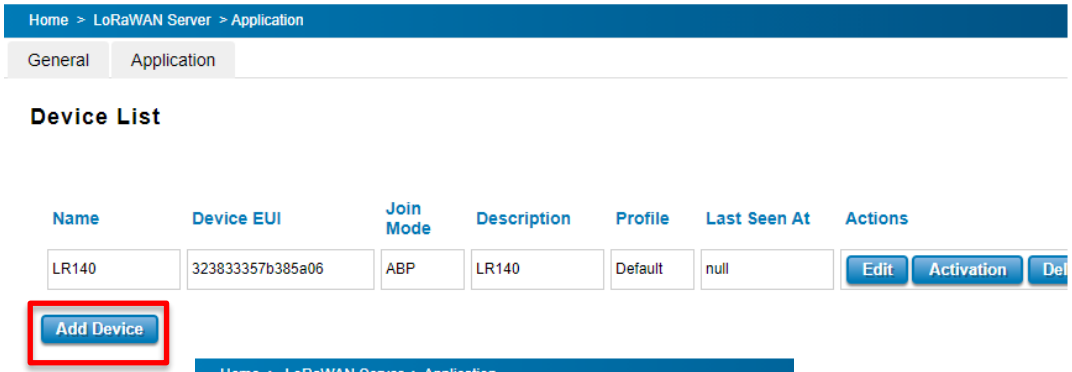
Get LoRaWAN node deveui and appkey for OTAA join
Select OTAA Enable

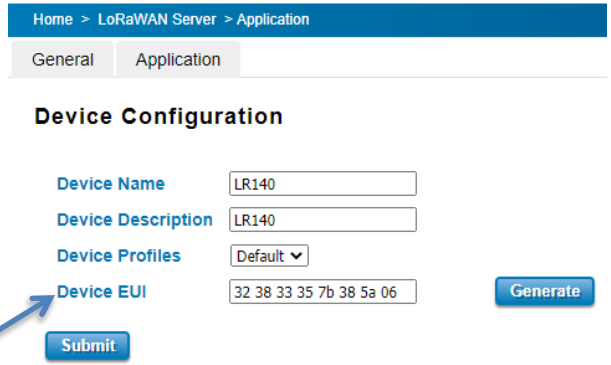
Join to LoRaWAN Server by OTAA (over th air)

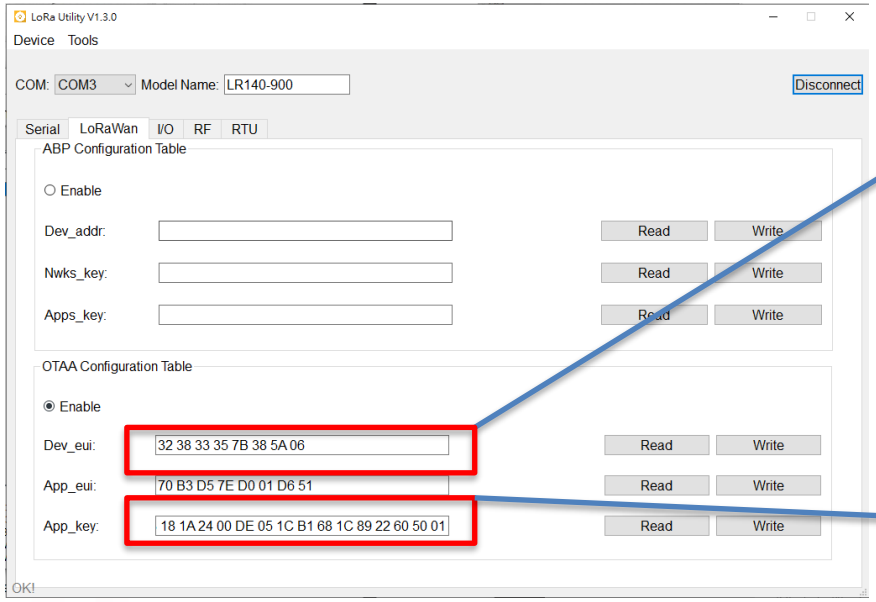
Disconnect USB and repowering, the LR140 will auto join LoRaWAN gateway and auto send LoRaWAN packet in a period

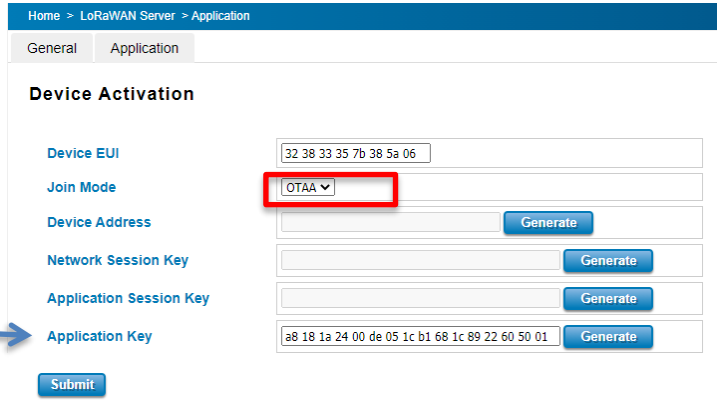
Add LR140 Device Info to WR322-EC

- http://host:8088/index.asp
- LoRaWAN Server-> Application-> Devices -> add Device
- Paste the DevEUI -> Submit
- LoRaWAN Server-> Application-> Devices -> Activation, Select Join mode to OTAA, paste Application key -> Submit



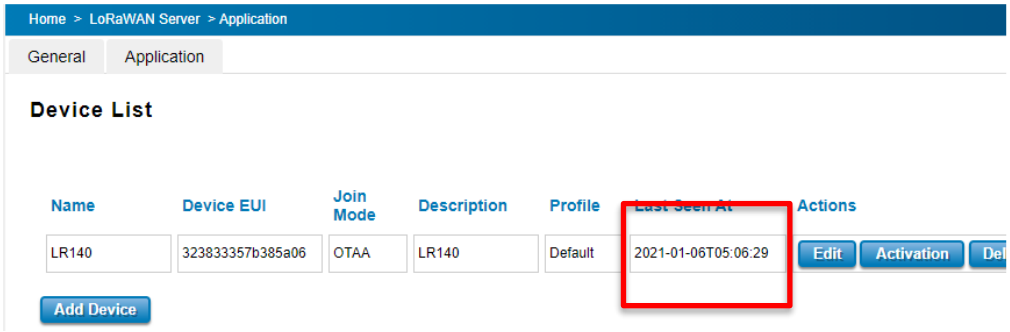


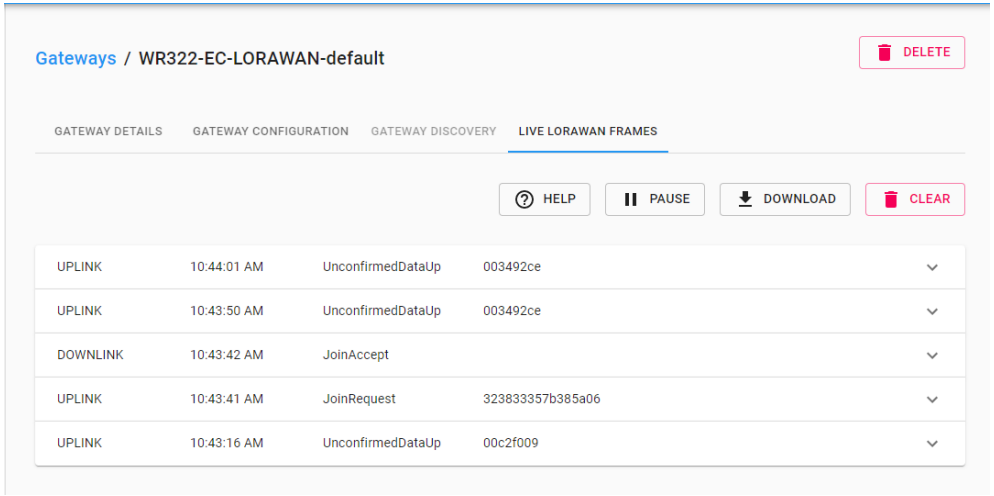




Display LR140 data on WR322-EC

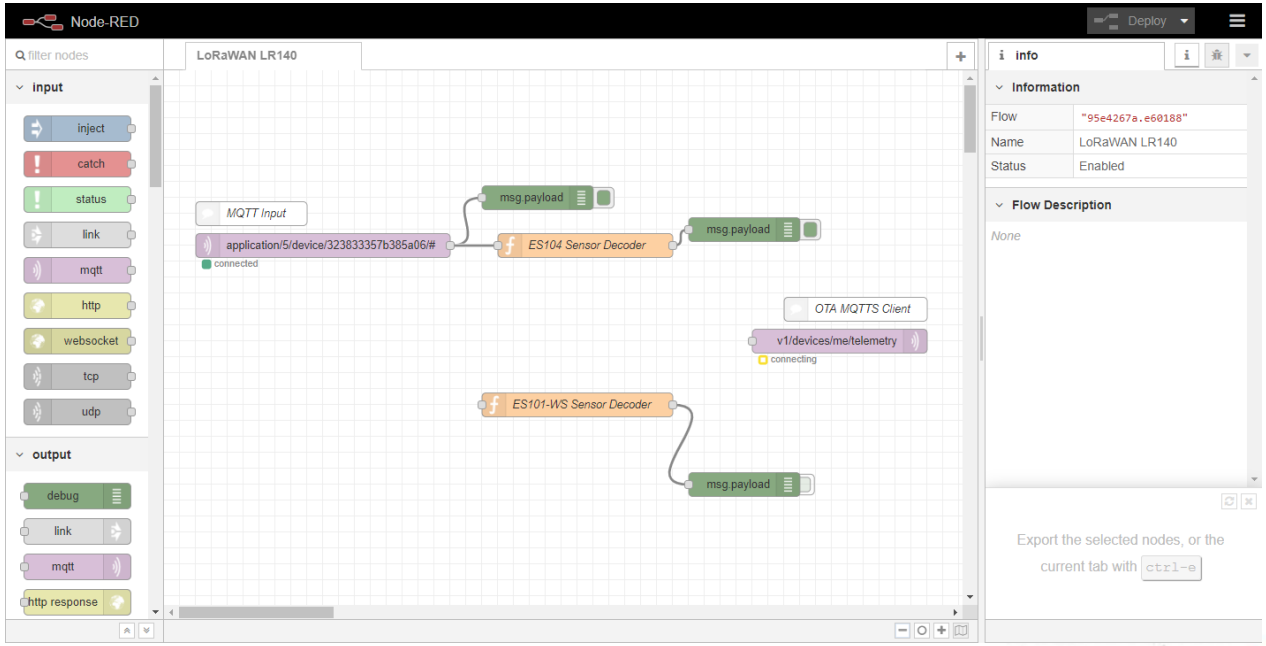
- Connect to http://host:8088/index.asp -> Application/Device List/Last seen at
- Connect to http://host:8080 (username/passwd: admin/admin)-> Gateway -> WR322-EC -> Live LoRaWAN Frames. You can see WR322-EC received data through LoraWan





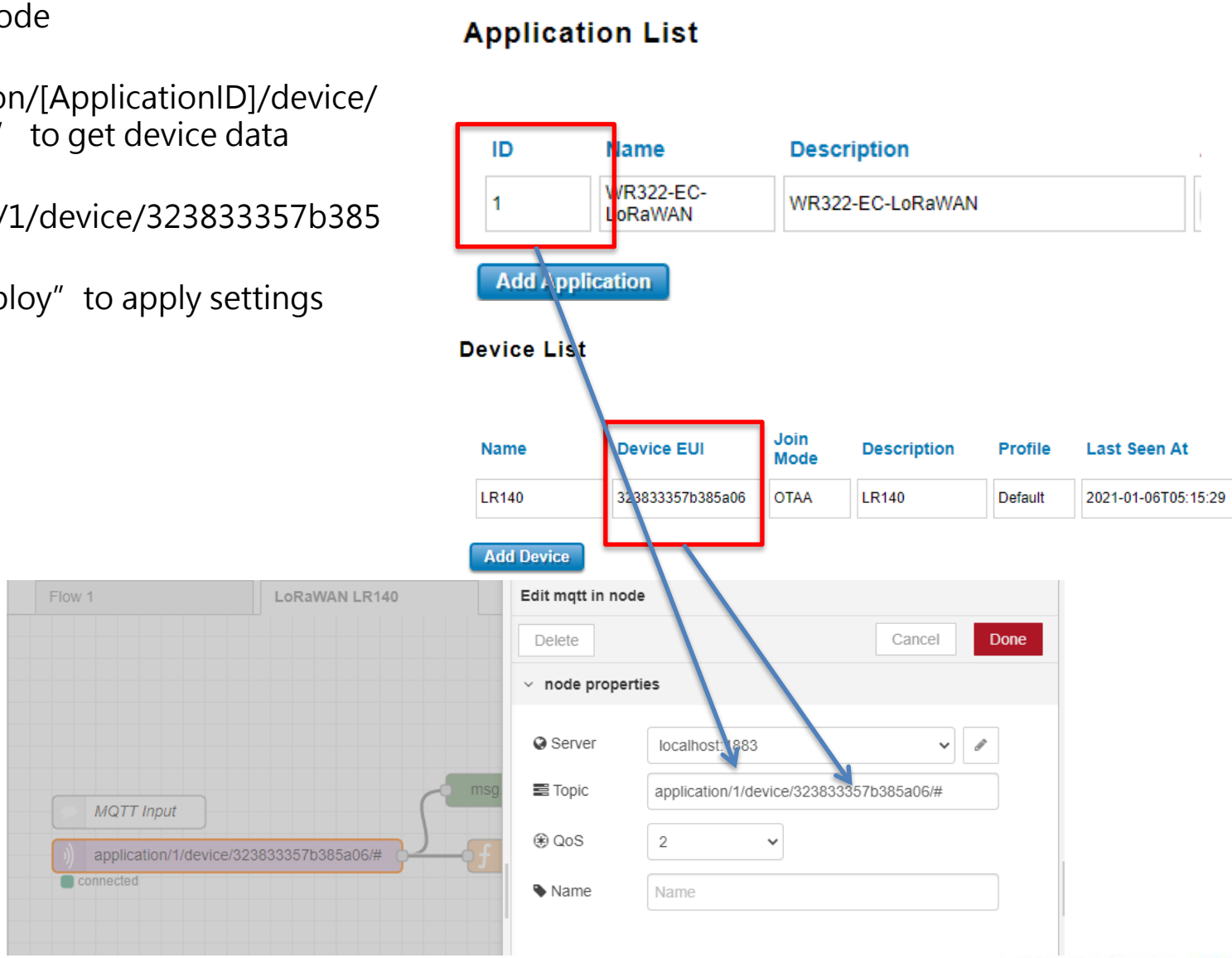
Open Node-RED on WR322-EC

- Open the node-red on your browser: http://<EC Series IP Address>:1880
- For example: http://192.168.1.1:1880
- And you will directly see the default flow for LoRaWAN LR140 and ES104/ES101-WS sensor



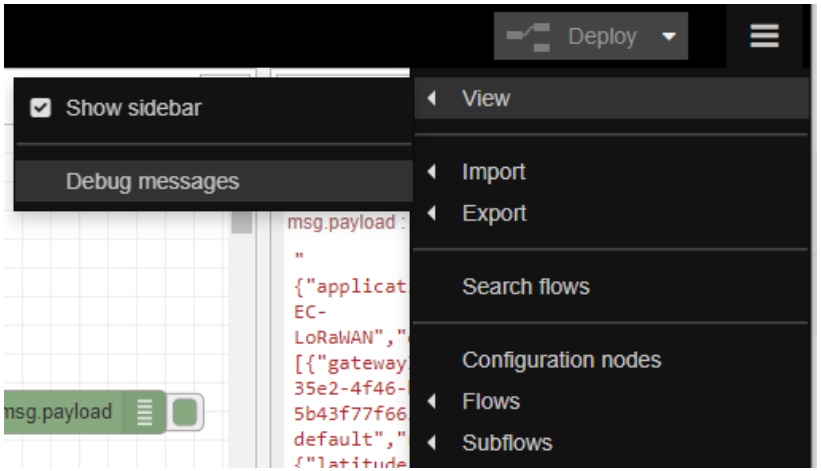
Edit MQTT input for LoRaWAN server

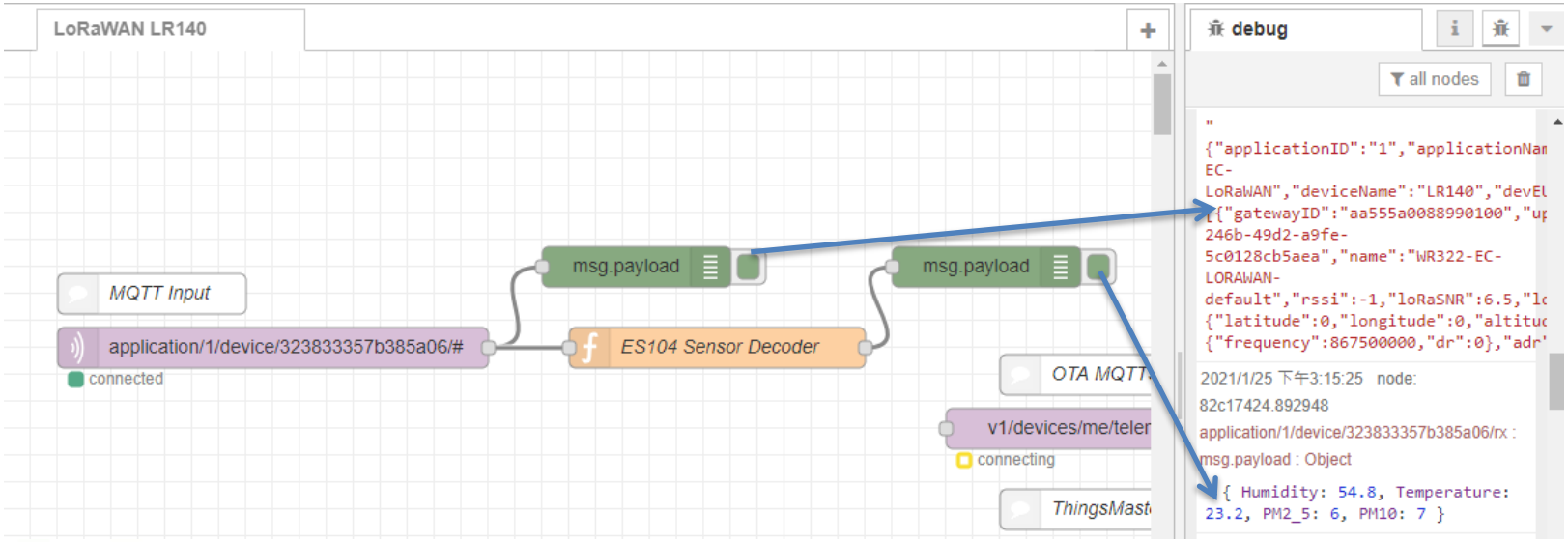
- Edit mqtt node
- Edit Topic
  “application/[ApplicationID]/device/[DevEUI]/#” to get device data
- Example: application/1/device/323833357b385a06/#
- Click “Deploy” to apply settings



Display Sensor ES104 data by Node-RED

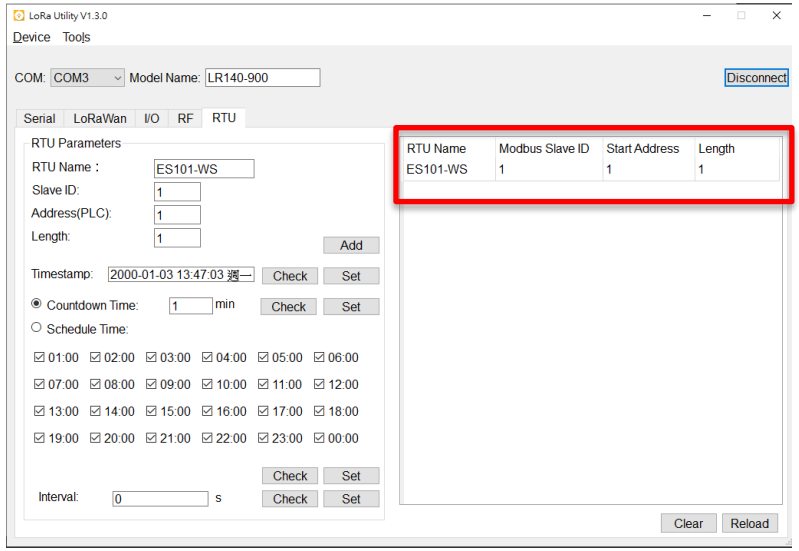
- View -> Debug Message
- It will display LR140 mqtt data and ES104 decoded data
- The ES104 decoded data will be {"Humidity":58.8,"Temperature":21,"PM2_5":4,"PM10":4}

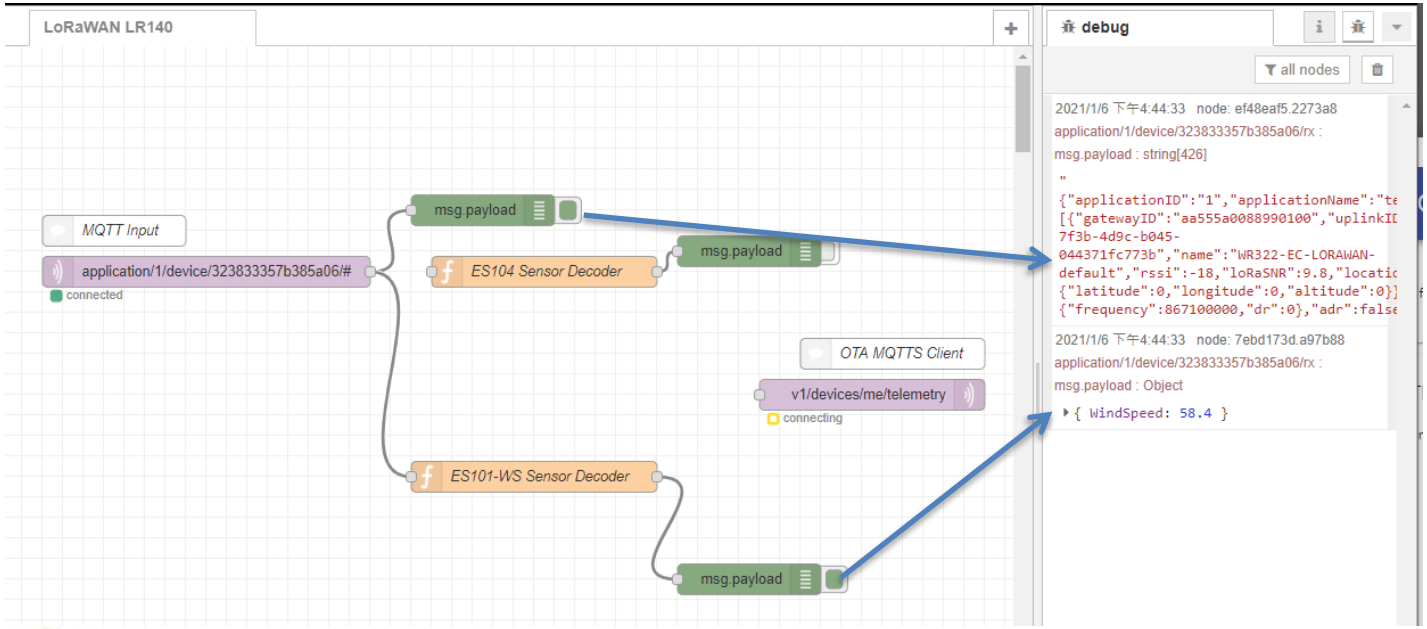




Display Sensor ES101-WS data by Node-RED

- Replace Sensor ES140 to ES101-WS
- Disconnect MQTT Input and ES104 node
- Connect MQTT Input and ES101-WS node
- The ES101-WS decoded data will be {"WindSpeed":58.4}





LR140 payload format and decoder

LR140 upstream payload Format：

0x0485 （2 byte）	0x0485 （2byte）	Time (7 byte)	Sensor Name (8 byte)

Modbus ID (1 byte)	Reg addr (2 byte)	Reg num (1 byte)	Reg val(Reg num*2 byte)	.....	Reg val(Reg num*2 byte)	0xaa (1 byte)

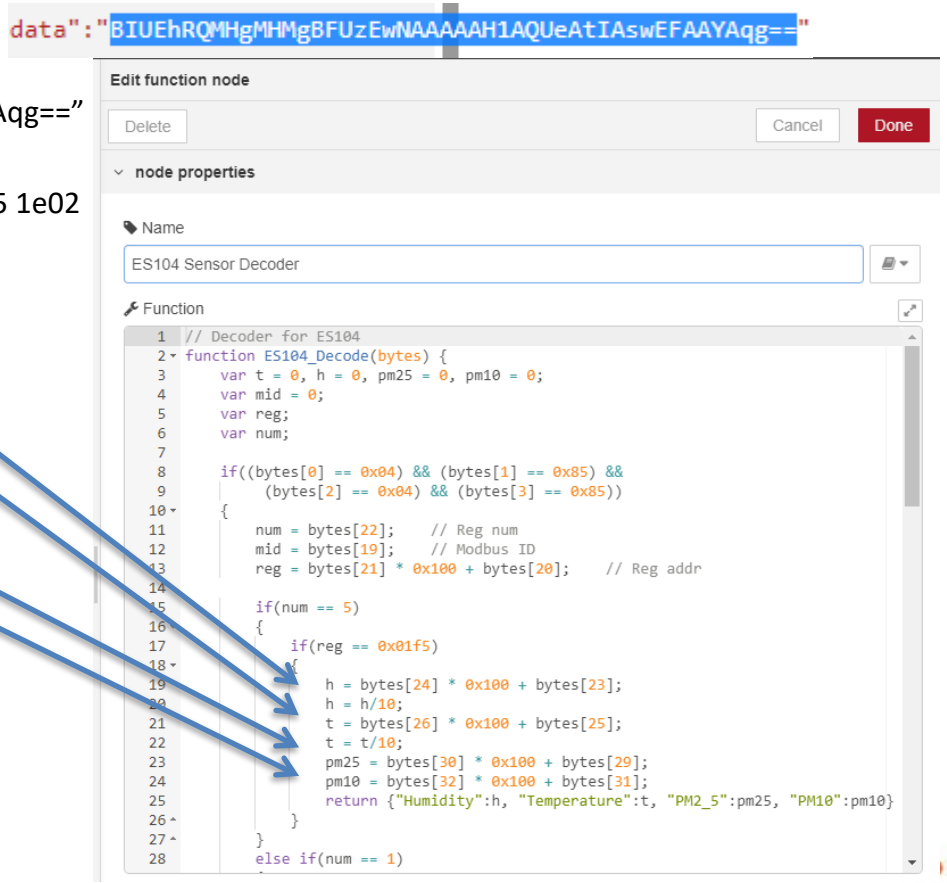
ES104 Modbus Register Information

Default slave ID	Parameters Function	Register Add. (HEX / DEC)	PLC Add. (Modbus Add.)	Note
	Device ID Storage Add.	07D0H / 2000	2001 (42001)	R/W , Range:1~254
	Serial Baud Rate Add.	07D1H / 2001	2002 (42002)	R/W , Default: 2 2(9600), 0(2400), 1(4800)
1	Humidity	01F4H / 500	501(40501)	R/O, Real Value = Read Value /10
1	Temperature	01F5H / 501	502(40502)	R/O, Real Value = Read Value /10
1	PM2.5	01F7H / 503	504(40504)	R/O, Real Value = Read Value
1	PM10	01F8H/ 504	505 (40505)	R/O, Real Value = Read Value

You will see LR140 data encode by base64
Example:
“BIUEhRQMHgMHMgBFUzEwNAAAAAH1AQUeAtIAswEFAAYAqg==”

Convert base64 to Hex:
04850485 140c1e 03 073200 4553313034000000 01 f501 05 1e02 d200 b301 0500 0600 aa
Sensor Name: “4553313034000000” -> “ES104”
Modbus ID: 01
Reg addr: “f501” -> 0x01f5 = 501
Reg num: 05
Reg 1 value: “1e02” -> 0x021e = 542 -> Humidity: 54.2
Reg 2 value: “d200” -> 0x00d2 = 210 -> Temperature: 21.0
Reg 4 value: “0500” -> 0x0005 = 5 -> PM2.5: 5
Reg 5 value: “0600” -> 0x0006 = 6 -> PM10: 6

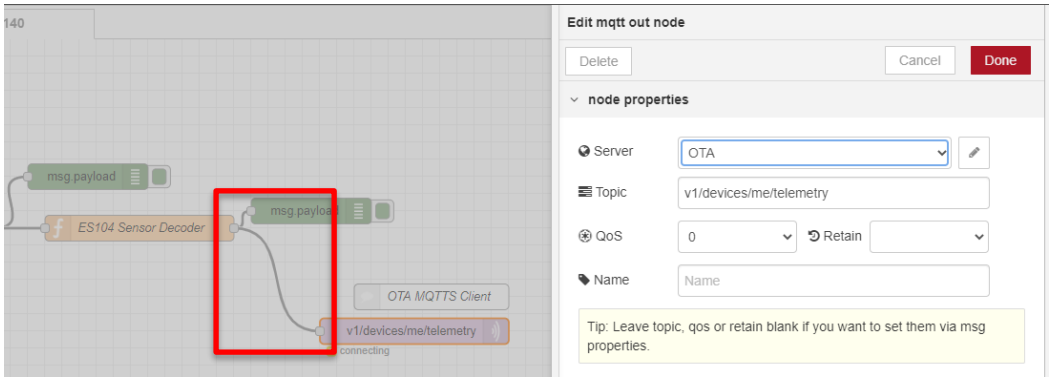
You can add decoder for other sensors by copy the “ES104 Sensor Decoder” and modify it.

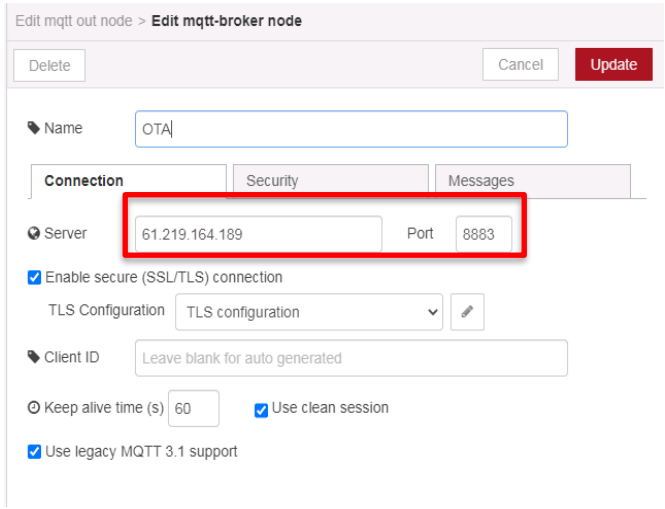


WoMaster

OTA integration (Optional)

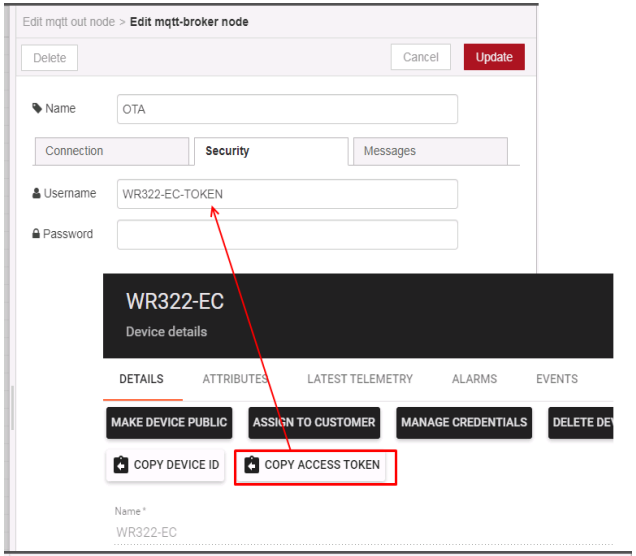
- Connect “ES104 Sensor Decoder” to OTA “mqtt out node”
- Edit the mqtt out node
- Edit the mqtt-broker node
- Fill in Server IP, Port, and enable the (SSL/TLS) connection

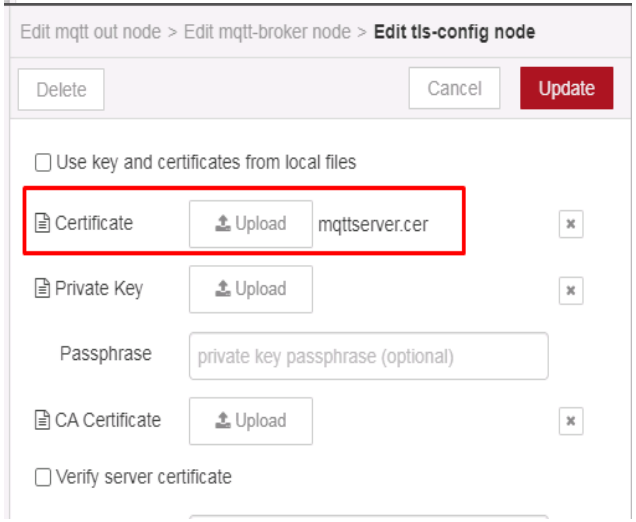


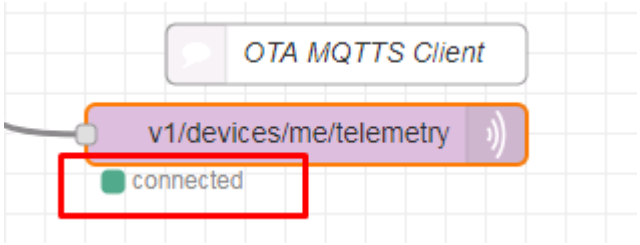


OTA integration (cont.)

- The Username in the Security tab should be set to the ACCESS TOKEN copy from OTA device details
- Upload the MQTTS Certificate and press Update
- The MQTT client will show “connected” if it connect to the OTA server

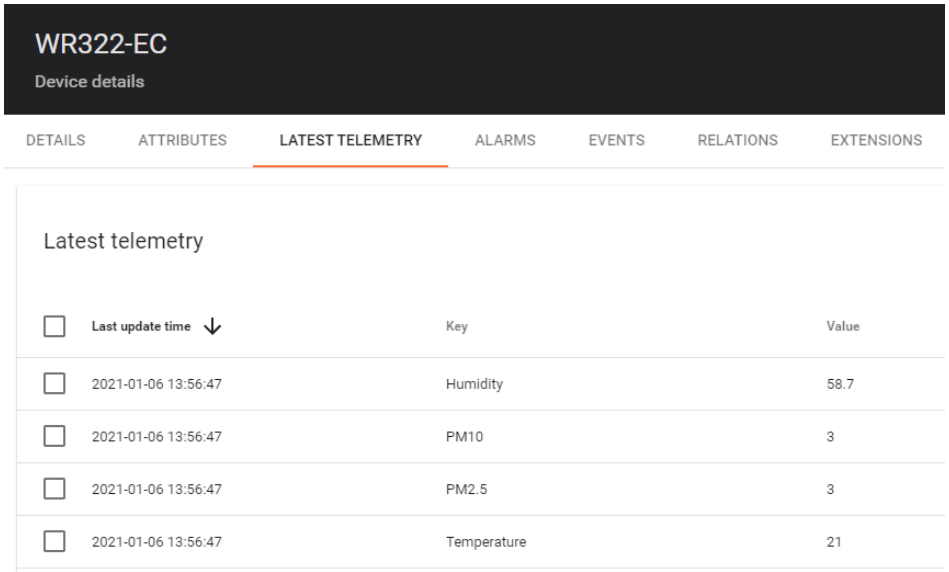






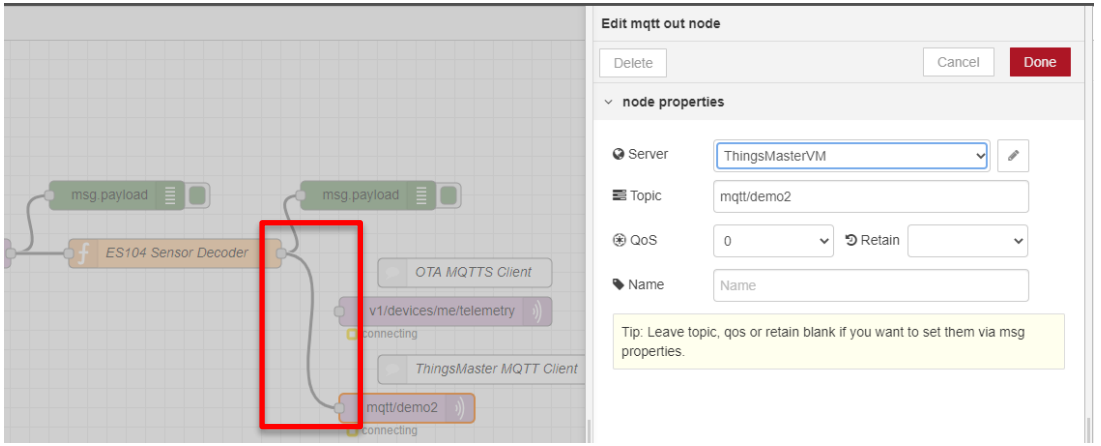
Display data from LR140 to OTA

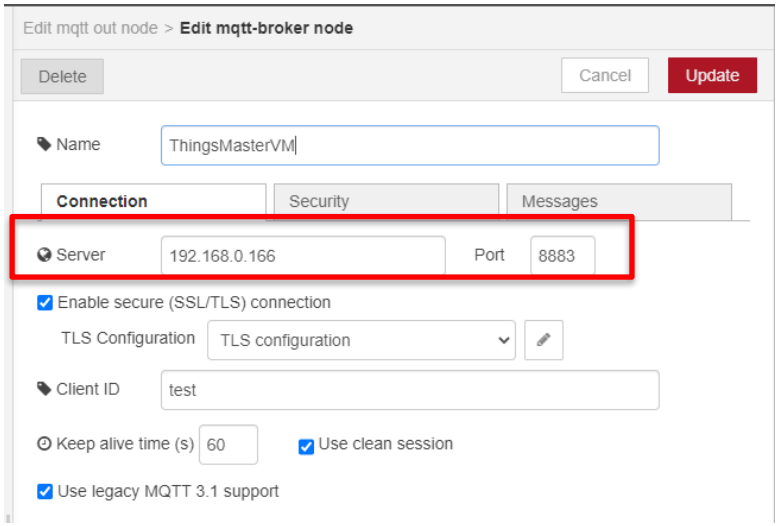
- OTA -> Device -> LASTEST TELEM
- OTA -> dashboard



ThingsMaster integration (Optional)

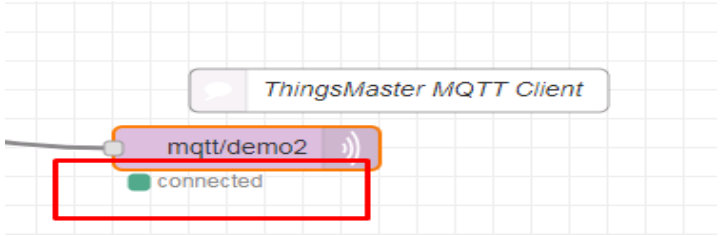
- Connect “ES104 Sensor Decoder” to ThingsMaster “mqtt out node”
- Edit the mqtt out node
- Edit the mqtt-broker node
- Fill in Server IP, Port, and enable the (SSL/TLS) connection

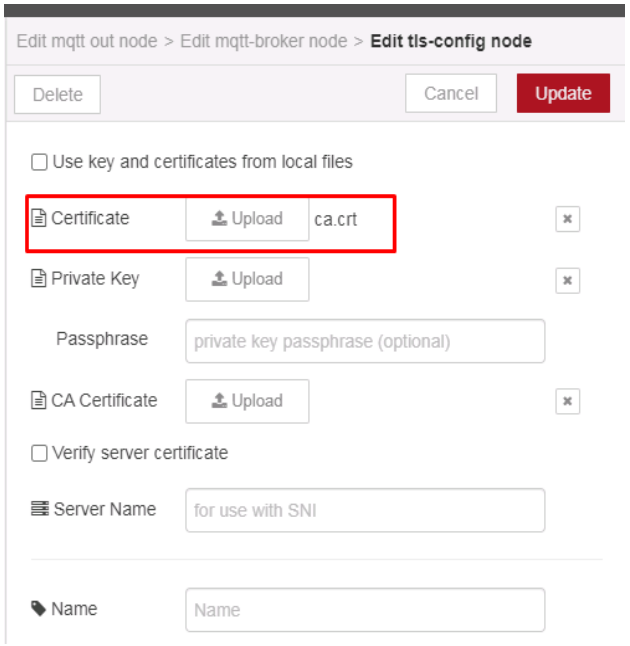




ThingsMaster integration (cont.)

- Upload the MQTTS Certificate and press Update
- The MQTT client will show “connected” if it connect to the ThingsMasterserver





Display data from LR140 to ThingsMaster