

Watson IoT Platform – T4 - NodeRed – Subscriber Client

Käyttöönotto-ohje 4

1. Johdanto

Aiemmissä käyttöönotto-ohjeissa loimme yhteyden todellisen laitteen ja IBM Watson -IoT -alustan välille. Loimme myös yhteyden simuloidun client -laitteen sekä todellisen client -laitteen ja IBM Watson:iin sijoitetun NodeRED -palvelimen välille.

Aiemmissä käyttöönotto-ohjeissa client toimi mqtt-protokollan mukaisena julkaisijana – Publisher. Tässä käyttöönotto-ohjeessä määrittelemme IBM Watson IoT-alustaan laitteen, joka voi toimia mqtt-protokollan mukaisena tilaajana – Subscriber.

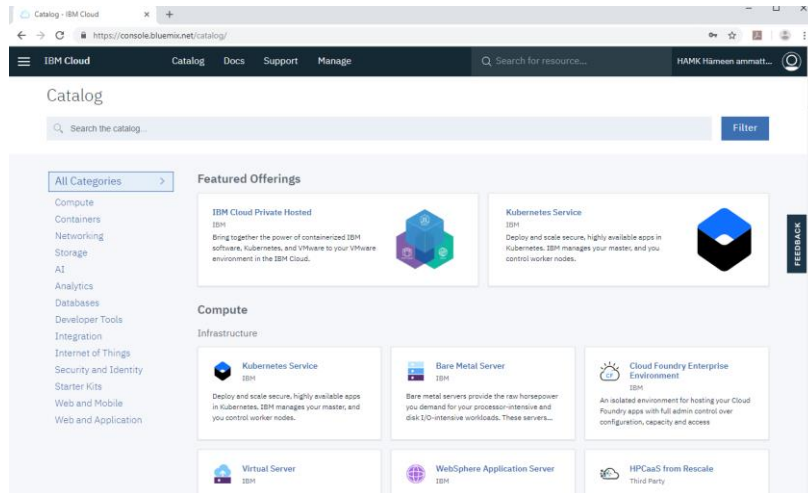
Sekä publisher että subscriber -laitteet tunnistautuvat Watson IoT -alustasta saamansa client ID -tunnuksen ja Authentication – salasanan perusteella. Lisäksi sekä publisher topic että subscriber topic on määriteltävä juuri täsmälleen tietyn muotoiseksi.

2. Luodaan client sekä tiedon tuottajaa varten että tiedon lukijaa varten

Voit selaimessa hakea haullla "IBM Cloud". Valitse tarjolla olevista haun kohteista <https://console.bluemix.net/catalog/>. Oikea osoite tietenkin tehtävän suoritushetkellä voi jo poiketa tästä 2019 alun tilanteesta.

Client -laitteiden määrittelyä varten sinulla on oltava IBM Cloud -käyttäjätunnus. IoT-alustan toimintojen luomiseen tarvittavan NodeRED -ympäristön perustamiseen Watson IoT:ssa käyttäjätunnuksella on oltava academic student tai staff -status tai jokin muu free trial -statusta laajemmat oikeudet omaava käyttäjätunnus. Ohjeet näiden luomiseen ovat aiemmissa käyttöönotto-ohjeissa.

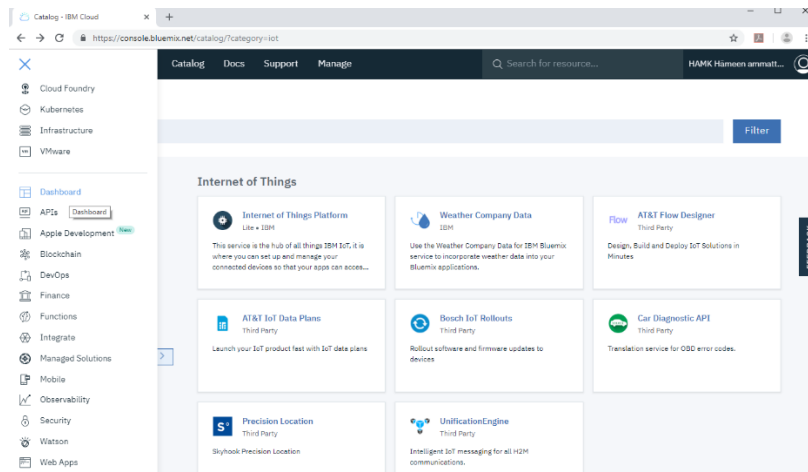
Kirjaudu tunnuksellasi IBM Cloud:iin!



kuva 2.1 Kirjautu IBM Cloud -palveluun.

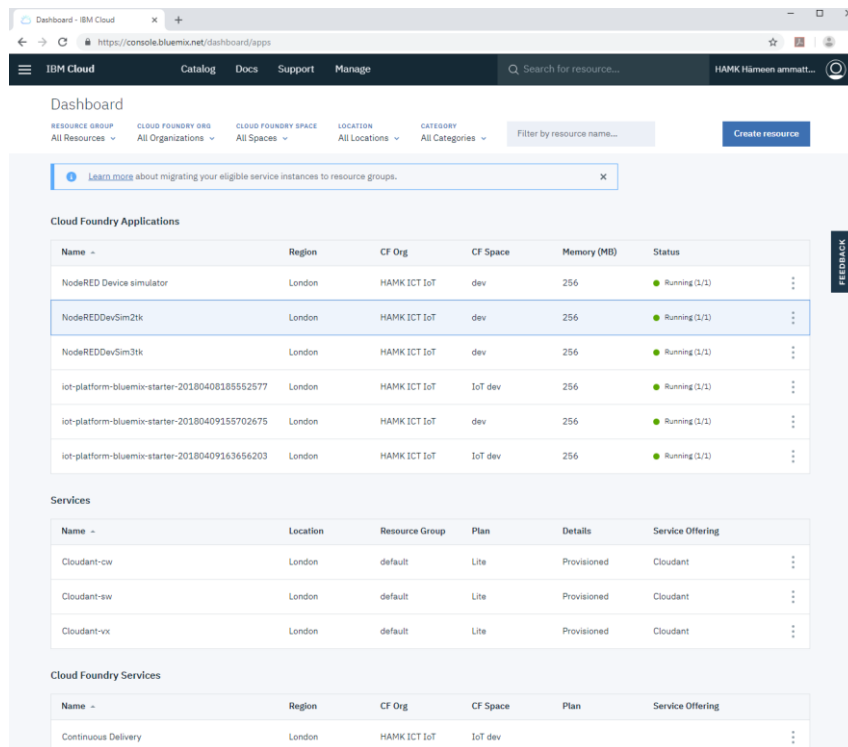
Jos et ole jo aiemmin perustanut Internet of Things Platform -palvelua käyttäjätunnuksellesi, perusta se nyt.

Jos sinulla jo on aiempien tehtävien Internet of Things Platform -palvelu, hakeudu siihen valitsemalla vasemmalta alasvetovalikosta Dashboard .



Kuva 2.2 Valitse alasvetovalikosta Dashboard.

Dashboard näyttää sinulle kaikki aiemmissa tehtävissä luodut palvelut.



The screenshot shows the IBM Cloud Dashboard with the following sections:

- Cloud Foundry Applications:** A table listing applications with columns: Name, Region, CF Org, CF Space, Memory (MB), and Status.

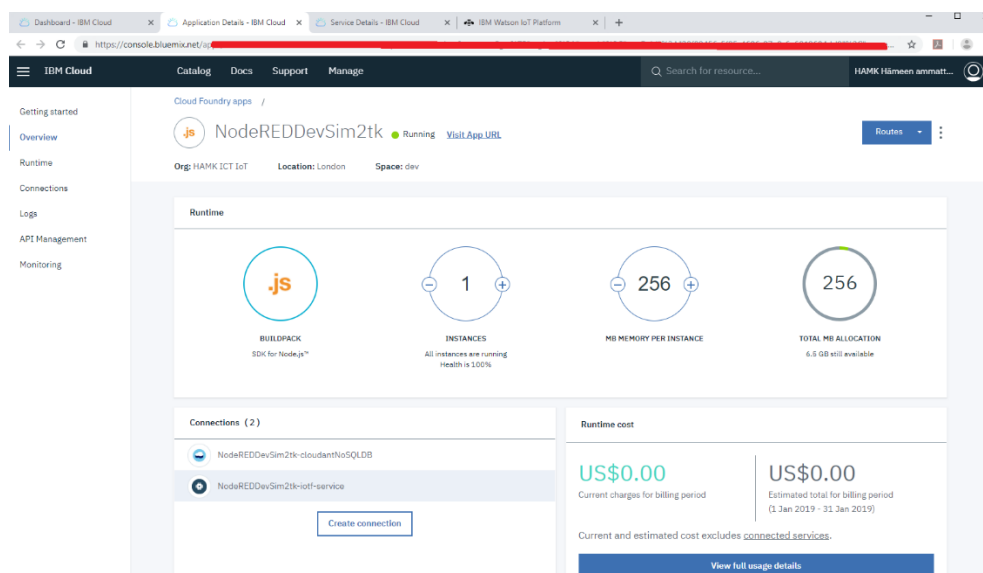
Name	Region	CF Org	CF Space	Memory (MB)	Status
NodeRED Device simulator	London	HAMK ICT IoT	dev	256	Running (2/1)
NodeREDDevSim2tk	London	HAMK ICT IoT	dev	256	Running (2/1)
NodeREDDevSim3tk	London	HAMK ICT IoT	dev	256	Running (2/1)
iot-platform-bluemix-starter-2018040818552577	London	HAMK ICT IoT	IoT dev	256	Running (2/1)
iot-platform-bluemix-starter-20180409155702675	London	HAMK ICT IoT	dev	256	Running (2/1)
iot-platform-bluemix-starter-20180409163656203	London	HAMK ICT IoT	IoT dev	256	Running (2/1)
- Services:** A table listing services with columns: Name, Location, Resource Group, Plan, Details, and Service Offering.

Name	Location	Resource Group	Plan	Details	Service Offering
Cloudant-cv	London	default	Lite	Provisioned	Cloudant
Cloudant-sw	London	default	Lite	Provisioned	Cloudant
Cloudant-vx	London	default	Lite	Provisioned	Cloudant
- Cloud Foundry Services:** A table listing services with columns: Name, Region, CF Org, CF Space, Plan, and Service Offering.

Name	Region	CF Org	CF Space	Plan	Service Offering
Continuous Delivery	London	HAMK ICT IoT	IoT dev		

Kuva 2.3 Aiemmissä tehtävissä luodut palvelut

Client -laitteen voi luoda avaamalla ylimmän alaotsikon Cloud Foundry Applications - sovellusalueista sen, jota käytettiin edellisissä tehtävissä Device – ja NodeRED -tehtävissä. Sovellusalueelta sitten avataan ...-iotf-service -tila.



The screenshot shows the IBM Cloud Application Details page for NodeREDDevSim2tk. The page includes the following sections:

- Overview:** Shows the application is running with a status of "Running".
- Runtime:** Displays key metrics:
 - Buildpack:** SDK for Node.js™
 - Instances:** 1 instance running, Health is 100%.
 - MB Memory per Instance:** 256 MB.
 - Total MB Allocation:** 256 MB, 6.5 GB still available.
- Connections (2):**
 - NodeREDDevSim2tk-cloudantNoSQLDB
 - NodeREDDevSim2tk-iotf-service
- Runtime cost:**
 - Current charges for billing period: US\$0.00
 - Estimated total for billing period (1 Jan 2019 - 31 Jan 2019): US\$0.00

Kuva 2.4 IoT Service

Samaan tilaan pääsee myös valitsemalla Dashboard-alaotsikon Cloud Foundry Services alta vastaavaiotf-service -tila.

Kun ...iotf-service avautuu valitaan painike Launch. Saamme näkyviin Browse devices -luettelon.

Tätä käyttöönotto tehtävää varten kannattaa luoda kaksi uutta laitemäärittelyä. Klikkaa Add Device ja vastaile kysymyksiin. Voit nimetä uudet laitteet vastaavasti kuin tässä esimerkissä. Anna palvelun luoda authentication token -salasanat.

Tämän esimerkin kaksi uutta laitemäärittelyä:

Organization ID: 8yyyyyp
Device Type: MonitoringClient
Device ID: MonitorC01
Authentication Method: use-token-auth
Authentication Token: BgF?xxxxxxxxQfLvfe

Organization ID: 8yyyyyp
Device Type: MonitoringClient
Device ID: MonitorC02
Authentication Method: use-token-auth
Authentication Token: olh8xxxxxxxxN!oJ*W

Määrittelyssä yyyy peittää osan organisation ID-tunnuksesta ja xxxxxx peittää osan Authentication token:sta. Tallenna omat laitemäärittelyt erilliseen tekstitiedostoon myöhempiä käyttöä varten!

3. Testaaminen tehdään mqtt -protokollan mukaisen client:n toteuttavalla sovelluksella

Tällä kertaa emme käytä erillistä antureita lukevaa laitetta. Teemme käyttöönotto testauksen käyttäen Windows-käyttöjärjestelmässä tavallisessa tietokoneessa ajettavaa MQTTbox -sovellusta. Muitakin vastaavia sovelluksia voi käyttää. Vastaavia sovelluksia on olemassa myös Android- ja iOS -järjestelmien älypuhelimille.

HAMK:n tietoverkossa oleville koneille MQTTbox voidaan asentaa SoftwareCenter-palvelusta. Omalle tietokoneelle sen voi asentaa sivustolta <http://workswithweb.com/mqttbox.html>. Asentamiseen ei tarvita tietokoneessa asennusoikeuksia.

Mikä mqtt client sinulla onkaan käytössä on sille syötettävä seuraavanlaiset asetukset:

MQTT Client Name: anything goes
MQTT Client Id: d:8yyyyyp:MonitoringClient:MonitorC01
Protocol: mqtt/tcp
Host: 8zzzp.messaging.internetofthings.ibmcloud.com
Username: use-token-auth
Password: BgF?xxxxxxxxQfLvfe
Append timestamp to MQTT client id: No

Topic to publish: iot-2/evt/Mon01/fmt/json
Payload Type: String/JSON/XML/Character
Payload: { "Opening": "This is Mon01", "fromNetwork": "Home",
"fromLogicNetwork": "Home", "AttemptNumber": 1 }

MQTT Client Name: anything goes
MQTT Client Id: d:8yyyynp:MonitoringClient:MonitorC02
Protocol: mqtt/tcp
Host: 8zzzzp.messaging.internetofthings.ibmcloud.com
Username: use-token-auth
Password: olh8xxxxxxxxN!oJ*W
Append timestamp to MQTT client id: No

Topic to subscribe: iot-2/cmd/monitoringResponse/fmt/json

Asetuksissa xxxx, yyyy, zzzz peittävät osan tunnuksista ja salasanasta.

Sovelluksessa voit clientille antaa minkä hyvänsä nimen. Client ID on oltava täsmälleen yllä kuvatus mukainen. Tähän syötät organisaation tunnuksen ja luomasi laitteen tunnuksen.

Host -osoitteessa näkyy luomasi organisaation iot-palvelun tunnus.

Muista poistaa aikaleiman lisäys client ID -tunnukseen. Yleensä lisätään aikaleima. Näin ei tule vahingossa yritettyä luoda kahta client-yhteyttä samalla tunnuksella.

Topic to Publish -kenttään on syötettävä mallin mukainen teksti. Mallissa näkyvän Mon01 -sanon sijaan voit antaa tapahtumalle muunkin nimen.

Topic to Subscribe -kenttään on syötettävä mallin mukainen teksti. Sanaa MonitoringResponse käytät myöhemmin luodessasi vastaus-toiminnon. Toki tässä voi käyttää muutakin sanaa.

Kuvissa alla näemme määrittelyt MQTTbox -sovelluksessa.

The screenshot shows the 'MQTT CLIENT SETTINGS' window in the MQTTBox application. The window has a menu bar with 'Menu', 'MQTT CLIENT SETTINGS', and 'Client Settings Help'. The settings are organized into a grid with labels and input fields. The fields are as follows:

MQTT Client Name	MQTT Client Id	Append timestamp to MQTT client id?	Broker is MQTT v3.1.1 compliant?
MonitoringClient01	d:8np:MonitoringClient:Mo	<input type="checkbox"/> No	<input checked="" type="checkbox"/> Yes
Protocol	Host	Clean Session?	Auto connect on app launch?
mqtt / tcp	8np.messaging.internetofthings.il	<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Username	Password	Reschedule Pings?	Queue outgoing QoS zero messages?
use-token-auth	*****	<input checked="" type="checkbox"/> Yes	<input checked="" type="checkbox"/> Yes
Reconnect Period (milliseconds)	Connect Timeout (milliseconds)	KeepAlive (seconds)	
5000	30000	10	
Will - Topic	Will - QoS	Will - Retain	Will - Payload
Will - Topic	0 - Almost Once	<input type="checkbox"/> No	

At the bottom, there are 'Save' and 'Delete' buttons.

Kuva 3.1 Ensimmäisen mqtt client:n asetukset

MQTTBox

MQTT CLIENT SETTINGS

Client Settings Help

MQTT Client Name: MonitoringClient02

MQTT Client Id: d:8...mp:MonitoringClient:Mo

Append timestamp to MQTT client id? ☐ No

Broker is MQTT v3.1.1 compliant? ☒ Yes

Protocol: mqtt / tcp

Host: 8...mp.messaging.internetofthings.il

Clean Session? ☒ Yes

Auto connect on app launch? ☒ Yes

Username: use-token-auth

Password: *****

Reschedule Pings? ☒ Yes

Queue outgoing QoS zero messages? ☒ Yes

Reconnect Period (milliseconds): 1000

Connect Timeout (milliseconds): 30000

KeepAlive (seconds): 10

Will - Topic: Will - Topic

Will - QoS: 0 - Almost Once

Will - Retain: ☐ No

Will - Payload:

Save Delete

Kuva 3.2 Toisen mqtt client:n asetukset

Ensimmäiseltä client:ltä lähetetään mqtt -protokollan mukainen sanoma Watson IoT -alustaan. Sanoman sisällöksi payload-kenttään voidaan kirjoittaa lähes mitä hyvänsä json-muodossa.

MQTTBox

MQTTBox Edit Help

Menu

Connected

Add publisher

Add subscriber

Settings

MonitoringClient01 - mqtt://8...mp.messaging.internetofthings.ibmcloud.com

Topic to publish: iot-2/evt/Mon01/fmt/json

QoS: 0 - Almost Once

Retain: ☐

Payload Type: Strings / JSON / XML / Characters

e.g: {"hello":"world"}

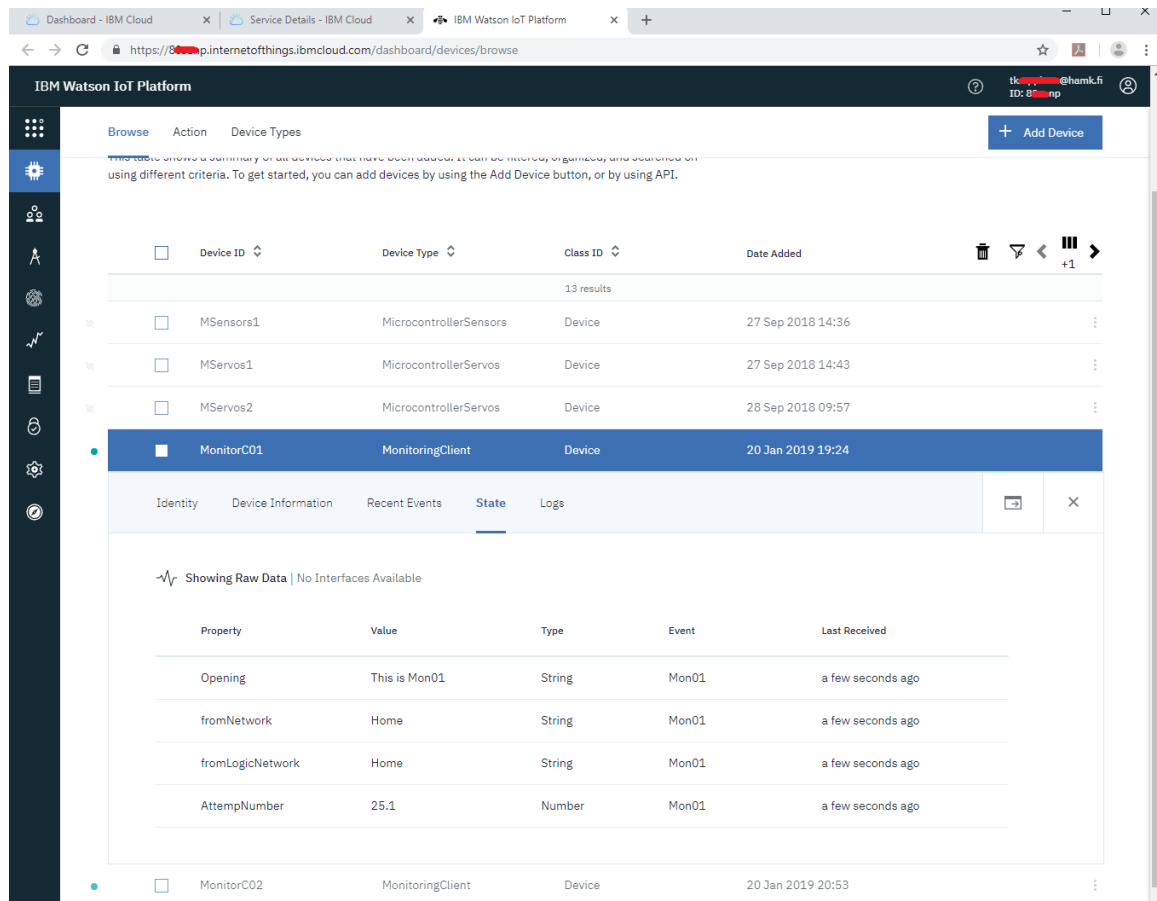
Payload: {"Opening":"This is Mon01", "fromNetwork":"Home", "fromLogicNetwork":"Home", "AttemptNumber":25.1 }

Publish

Kuva 3.3 Kirjoitetaan sanoman sisältö json-muodossa.

IBM Watson IoT Devices – näkymässä klikkaa vastaava laite auki ja laitteesta välilehti state.

Lähetä MQTTBox-sovelluksesta painikkeesta Publish.



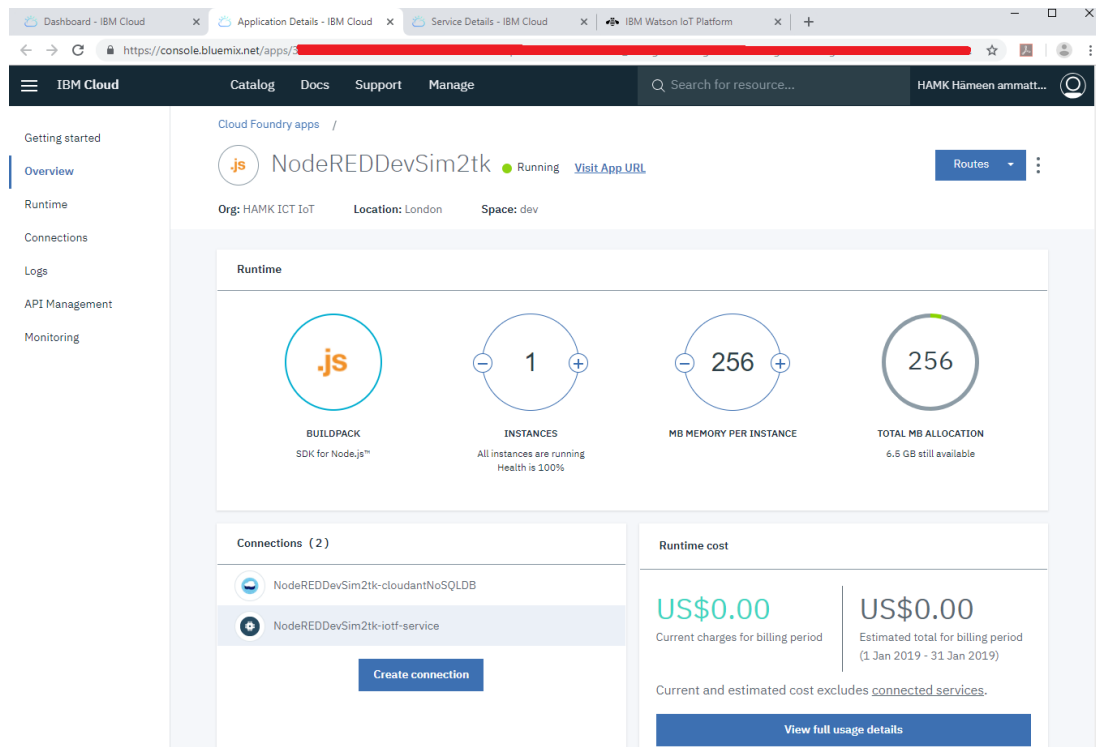
Kuva 3.4 Dashboard Devices -näkymässä nähdään json sanoma purettuna.

Tehtävä 1. Kirjoita mqtt client:n payload-kenttää json -muotoiseen sanomaan numeroarvo muodoissa: kokonaisluku 25, desimaaliluku 25.1, luku heittomerkeillä rajattuna "25.1" . Kokeile myös erikoismerkkejä. Mitä Watson IoT pystyy purkamaan json-muotoisesta sanomasta?

4. Sanoman lähetys Watson IoT:stä mqtt client:lle, NodeRED-määrittelyt.

Luomme sanoman tulkinnan ja vastaustoiminnon käyttäen NodeRED-kehitysympäristöä.

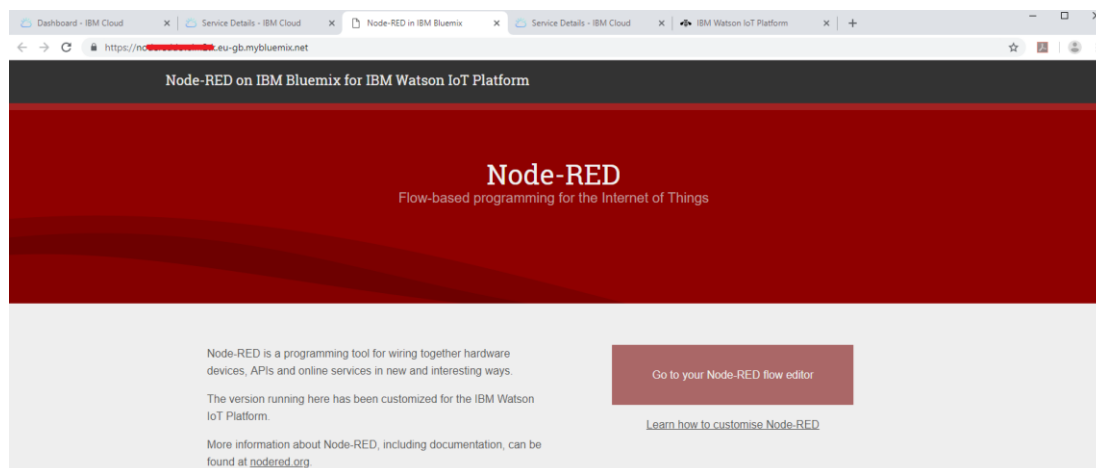
NodeRED -kehitysympäristö luotiin jo edeltävissä käyttöönotto-tehtävissä. Avaa jo luomasi NodeRED. Dashboard-näkymässä valitse saman Cloud Foundry Application -alueen kuin mihin loit edellä laitemäärittelyt.



Kuva 4.1 Cloud Foundry Application sisältää CloudantNoSQLDB -teitokannan ja iotf-service -IoT-laitemäärittelyt.

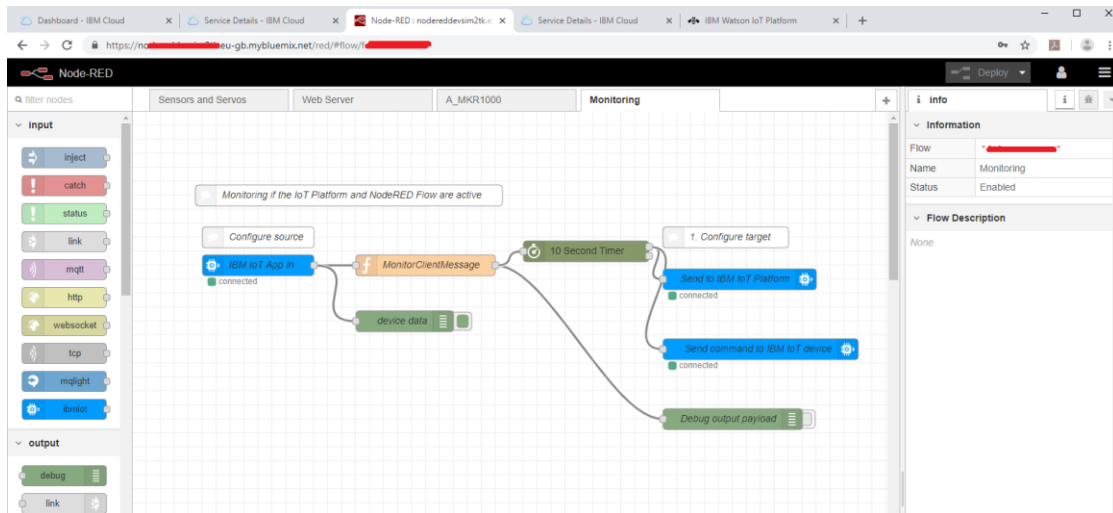
CloudantNoSQLDB sisältää aiemmissa käyttöönottoehtävissä luomasi NodeRED-ympäristön. Jos sinulla ei tätä ole, palaa käyttöönottoehtävään ” IBM Watson IoT – T2 – Platform – NodeRED flow” . Voit luoda NodeRED-määrittelyn tuon ohjeen mukaisesti.

Avaa connections-ikkunassa ...-cloudantNoSQLDB ja seuraavassa näymässä sovelluksesi ja avautuvaan sivuun annat oman NodeRED-editorisi käyttäjätunnuksen ja salasanan.



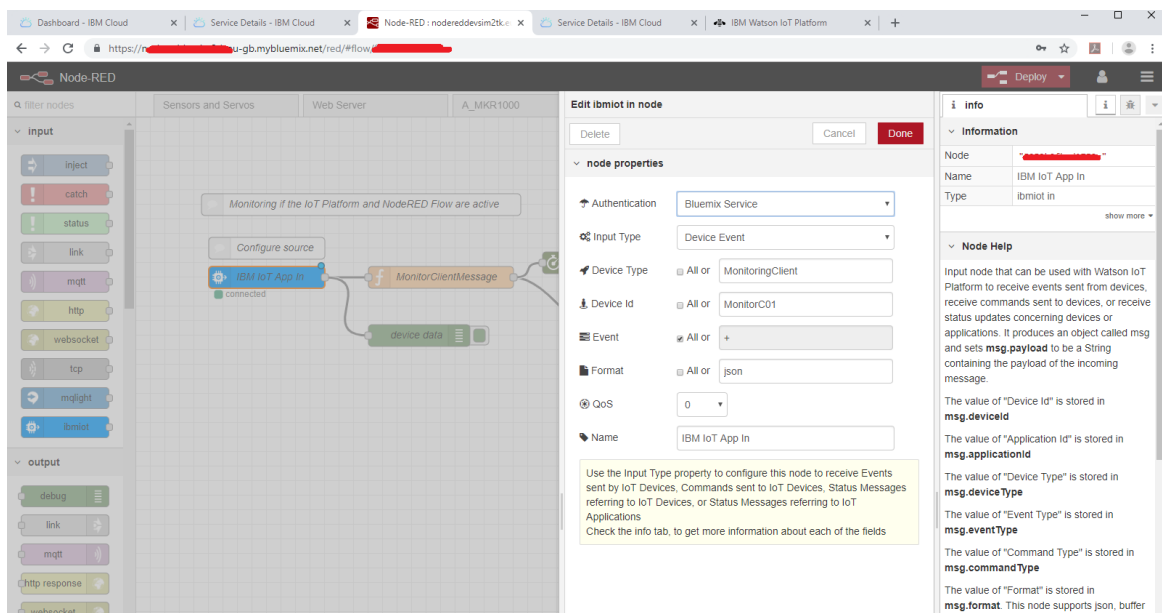
Kuva 4.2 Syötä oma käyttäjätunnus ja salasana NodeRED-editoriin.

Luo alla kuvassa näkyvän kaltainen flow. Tarvittavat node:t tunnistat väristä ja pienestä symbolista. Selvyden vuoksi voit nimetä node:t samoin kuin kuvassa.



Kuva 4.3 NodeRED flow, jolla tuotetaan laitteelle mqtt-protokollan mukainen vastausviesti.

Ensimmäisenä vasemmalla on node, jolla liitytään IoT-laitteeseen tai muutoin luotuun mqtt-protokollan mukaiseen client:iin.



Kuva 4.4 IBM IoT App In avattuna.

Kun klikkaat node:n auki, näet Info-välilehdellä Help-tekstin. Node:n kenttiin on syötettävä luomaasi laitetta vastaavat tiedot:

Authentication: Bluemix Service

Input Type: Device Event

Device Type: MonitorinClient Edellä määrittelyssä kirjaamasi Device Type- laitetyyppi

Device Id: MonitorC01 Määrittelyssä kirjaamasi Device Id

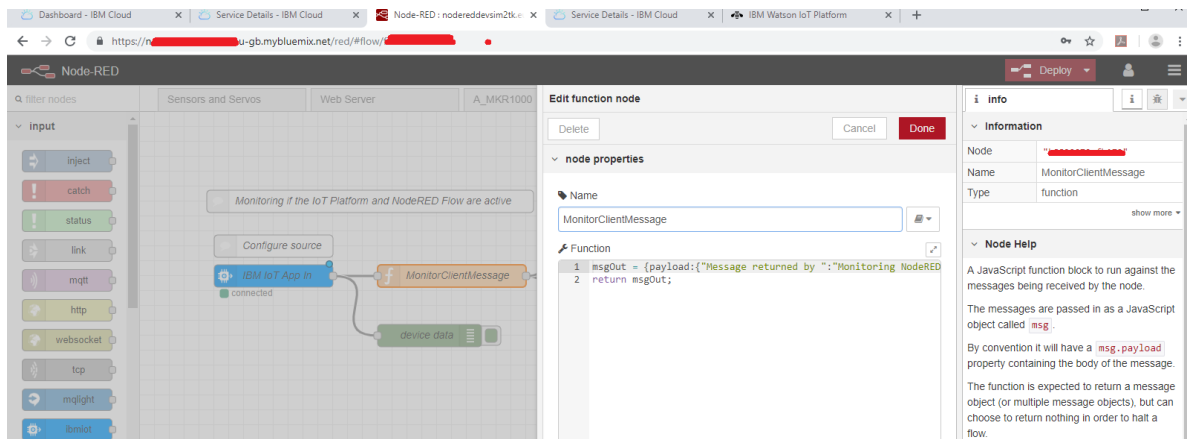
Event: valitse All

Format: json

QoS: 0

Name: voit nimetä tarvittaessa uudelleen

Vastaavasti avaa funktio-node.



Kuva 4.5 Function node avattuna.

Funktion sisällöksi voit kirjoittaa:

```
msgOut = {payload: {"Message returned by ": "Monitoring NodeRED flow", "received from device of type ": msg.deviceType, " of ID ": msg.deviceId, " payload ": msg.payload}};
return msgOut;
```

msgOut -sanomaksi muodostamme tässä json-muotoisen sisällön.

Olio msg välittää arvoja saapuvasta viestistä; msg.deviceType, msg.deviceId, msg.payload.

Avataan edelleen oikeasta ylhäältä node Send to IBM IoT Platform. Tälle kirjoitetaan määrittelyksi:

Authentication: Bluemix Service
Output Type: Device Event
Device Type: MonitoringClientmäärittelyssä kirjaamasi tyyppi
Device Id: MonitorC02 määrittelyssä kirjaamasi Id
Event Type: update
Format: json
Data: temp:10 tämä korvautuu funktion tuottamalla sanomalla
QoS: 0
Name: Send to IBM IoT Platform ... tai voit vaihtaa nimeä

Avataan edelleen oikeasta ylhäältä node Send command to IBM IoT Device. Tälle kirjoitetaan määrittelyksi:

Authentication: Bluemix Service
Output Type: Device Command
Device Type: MonitoringClient määrittelyssä kirjaamasi tyyppi
Device Id: MonitorC02 määrittelyssä kirjaamasi Id
Command Type: monitoringResponse ... tähän voit kirjata haluamasi tekstin
Format: json
Data: {"data": "data"} tämä korvautuu funktion tuottamalla sanomalla QoS

Name: Send command to IBM IoT device tai voit vaihtaa nimeä

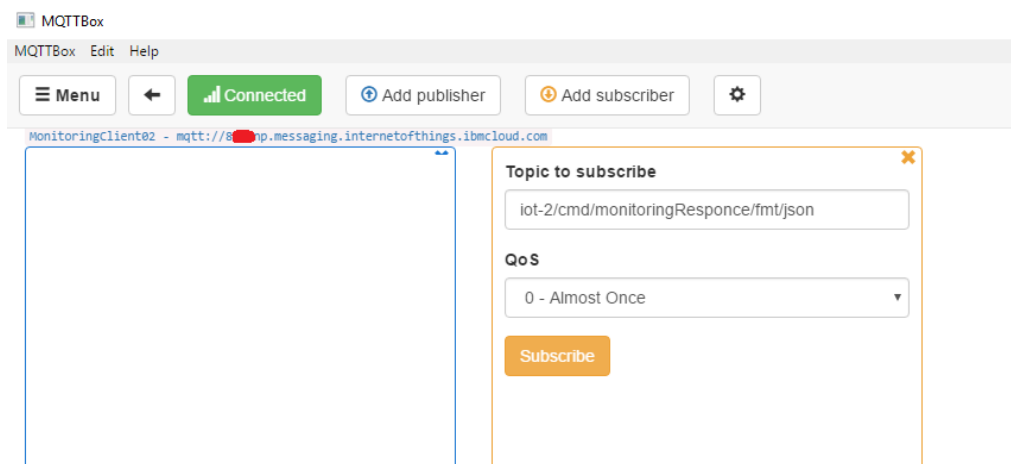
Ajastimen 10 Second Timer lisäsin, jotta on helpompi havaita, mikä alkuperäinen viesti ja mikä on vastauksena lähtevä käsky.

Määrittelyt flow editor:iin on nyt tehty.

5. Sanoman lähetys Watson IoT:stä mqtt client:lle, MQTTBox-määrittelyt.

Testaamme käyttäen MQTTBox -sovellusta.

Määritä MQTTBox:iin edellä määrittämäsi toiseen client:iin subscribe-toiminto.

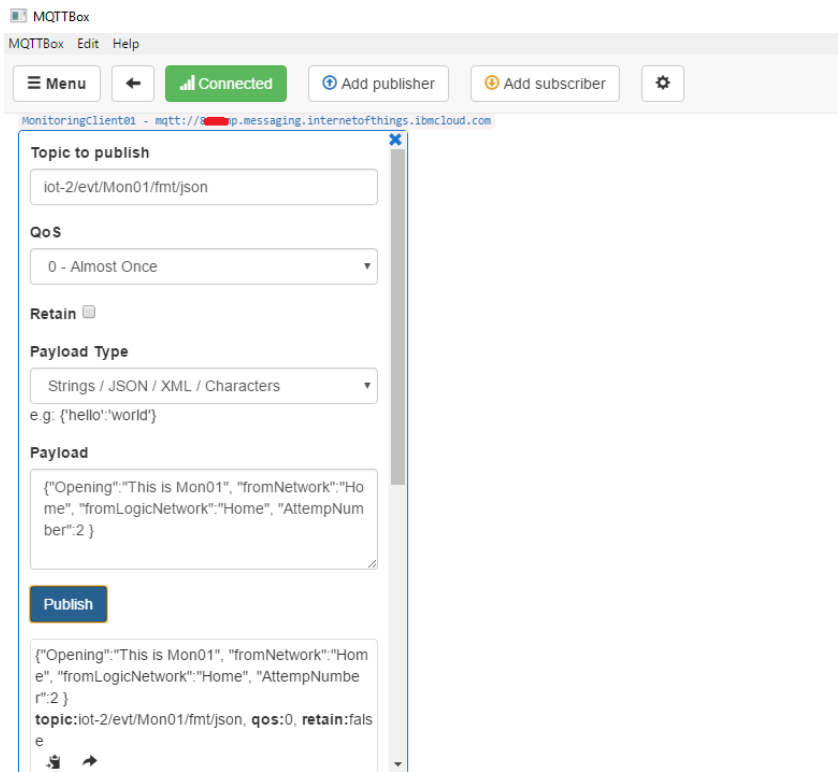


Kuva 5.1 Määritellään subscribe toiminto

Subscribe-toimintoon kirjaa asetuksiksi:

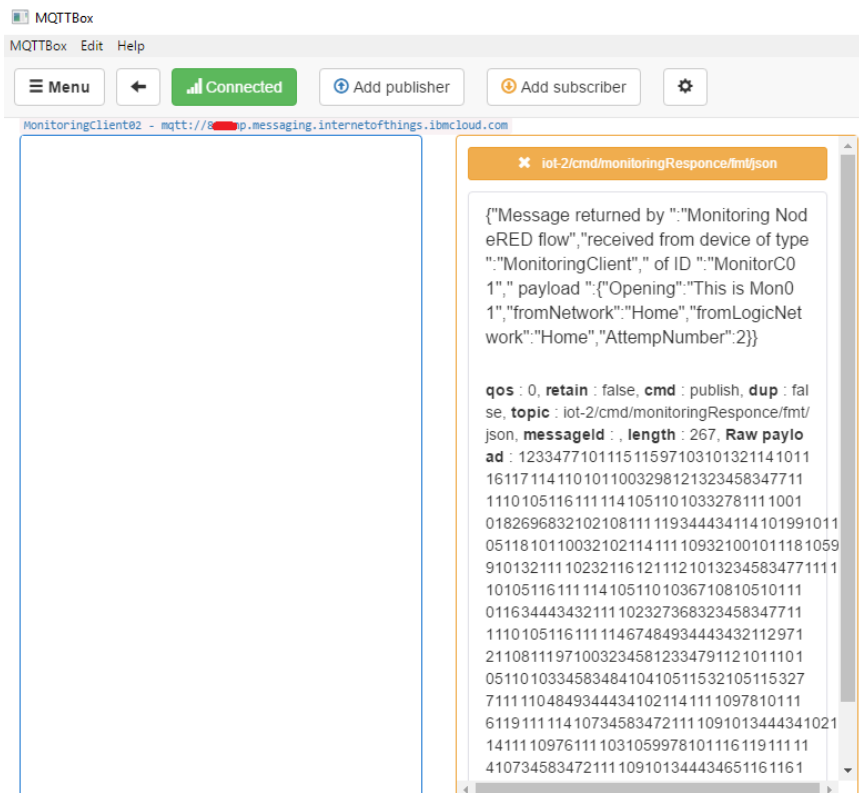
Topic to subscribe: iot-2/cmd/monitoringResponse/fmt/json
QoS: 0

Avaat lähettävän client:in ja lähetät publish-toiminnolla viestin.



Kuva 5.2 Lähetät viestin.

NodeRED flow:ssa määrittelit viiveen, jotta sinulle jää aikaa avata toinen client ennen viestin saapumista.



Kuva 5.3 Vastaanotat laitekomennon

6. Lopuksi

Huomaa vastaanotetussa sanomassa sanoman merkkien määrä. Tehtävämme komentosanomassa on 267 tavua.

Jos otat vastaan komentoja jollekin mikro-ohjainkortille kirjoitetun mqtt-kirjaston avulla, voi sanoman pituus olla rajoitettu huomattavan pieneksi.

Arduino pubsubclient: 90 bytes / <https://pubsubclient.knolleary.net/>

Arduino mqttclient Joël Gähviler: 128 bytes / <https://github.com/256dpi/arduino-mqtt>

Näissä edellä mainituissa mqtt clienteissa voi kasvattaa sanoman suurinta pituutta muuttamalla parametrin arvoa. Tutustu dokumentaatioon! Linkit dokumentaatioon ovat yllä.

Tehtävä 2. Muokkaa NodeRED flow:ssa vastauksena lähetettävän komentosanoman sisältöä niin, että sanoman pituudeksi tulee alle 90 tavua. Tämä vaatii muutaman kokeilukerran... merkkien määrää ei ole aivan helppo laskea tarkalleen oikeaksi!

Tehtävä 3. Muokkaa subscriber client:n asetukset ja topic-määrittely muotoon, jossa se ottaa vastaan tälle laitetyypille, tälle laitteelle lähetettyjä minkä hyvänsä Command Type: xxxx - määrittelyn komentoja.