Raspberry Pi Server based on Docker

Node-red, MQTT, Node-red dashboard

Table of contents

01 Pengenalan

Pengenalan Rpi

02 Instalasi RPI

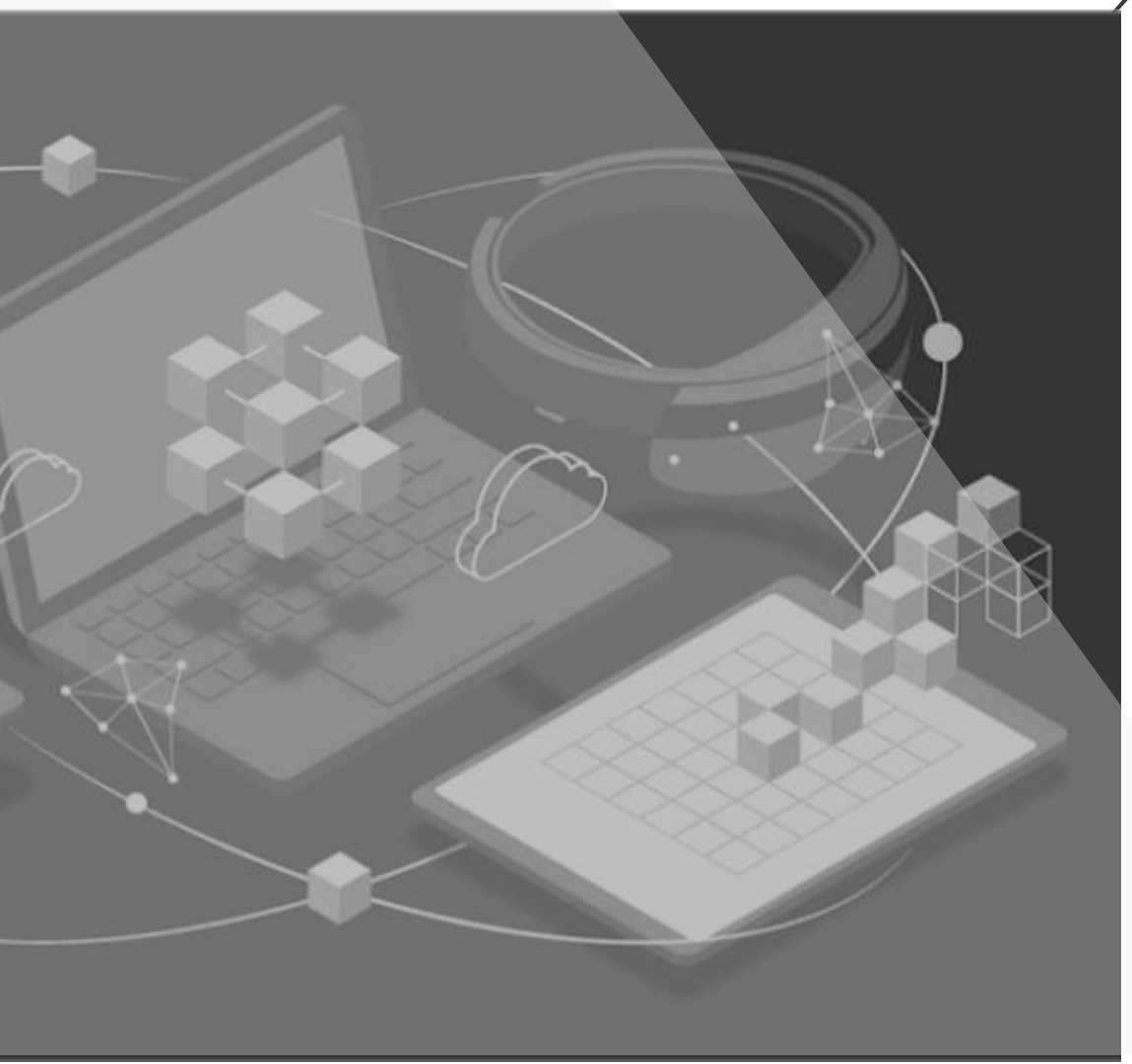
Instalasi docker menggunakan IoT Stack

03 MQTT Server

Koneksi MQTT ke Rpi

04 Dashboard

Menampilkan data pada dashboard



01

Pengenalan

Goal

- Memahami sistem Docker pada RaspberryPi
- Menggunakan headless Rpi
- Instalasi docker dengan IoTStack
- Koneksi ESP32 – Rpi dengan MQTT
- Menampilkan data sensor melalui mqtt dashboard

Raspberry pi

Raspberry Pi Boards

Raspberry Pi Zero

Raspberry Pi 4 Model B

Raspberry Pi 3 Model A+

Raspberry Pi 3 Model B+

Raspberry Pi 3 Model B

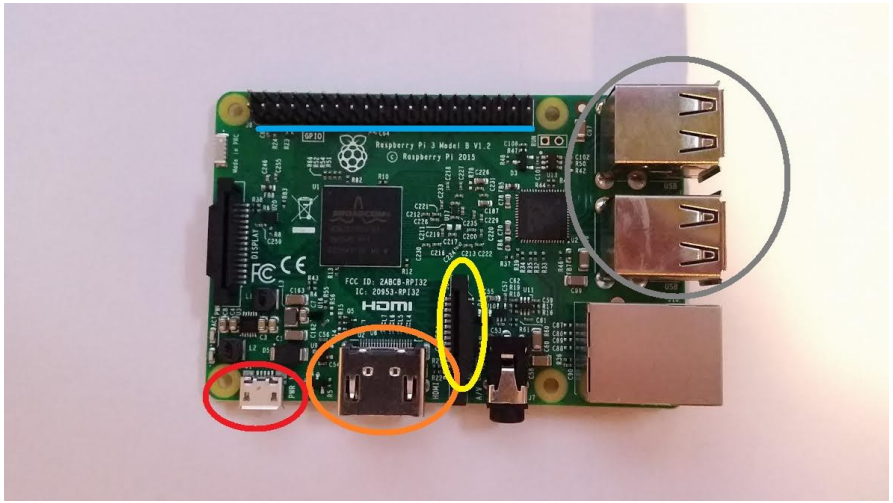
Raspberry Pi 2 Model B

Raspberry Pi 1 Model B+

Raspberry Pi 1 Model A+

Raspberry Pi Zero W

Raspberry pi 3

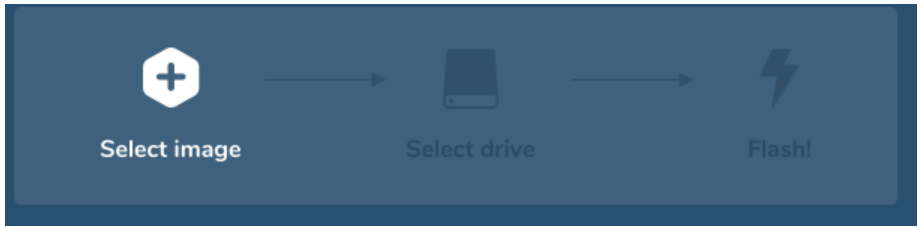


4x USB
Power micro usb
HDMI display
Camera interface
Ethernet
GPIO

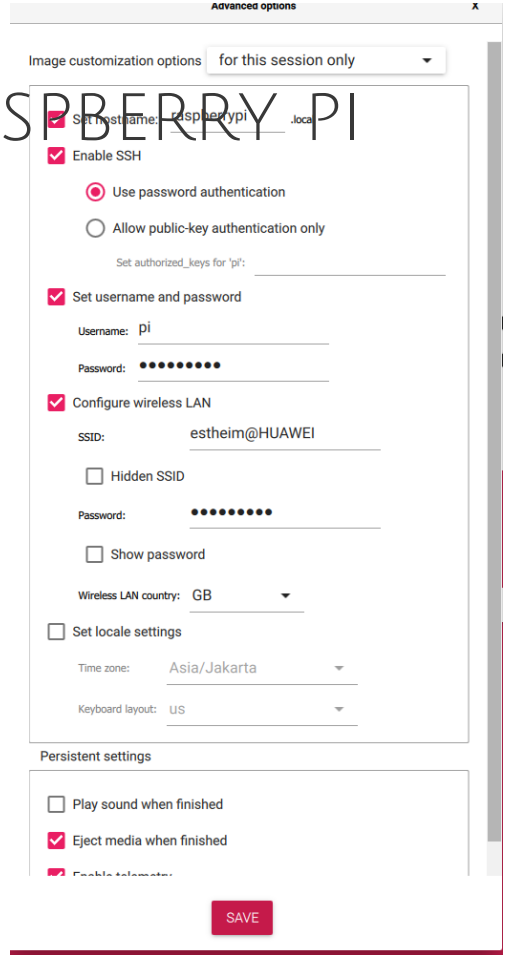
Mico sd

Install sdcard

- Download image
- Connect sdcard
- Find Raspberry pi *.img
- Flash



Install menggunakan Raspberry Pi Imager



Setting headless Rpi

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
        ssid="YOUR_SSID"
        psk="YOUR_PASSWORD"
}
```

- Set Open and edit wpa_supplicant.conf
- Scan ip
- Ssh using putty

```
C:\Users\estheim>nmap -PN 192.168.0.1/24

Starting Nmap 7.60 ( https://nmap.org ) at 2019-08-04 22:36 SE Asia Standard Time
Nmap scan report for 192.168.0.101
Host is up (0.013s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
5900/tcp open  vnc
MAC Address: B8:27:EB:34:DB:6F (Raspberry Pi Foundation)
```

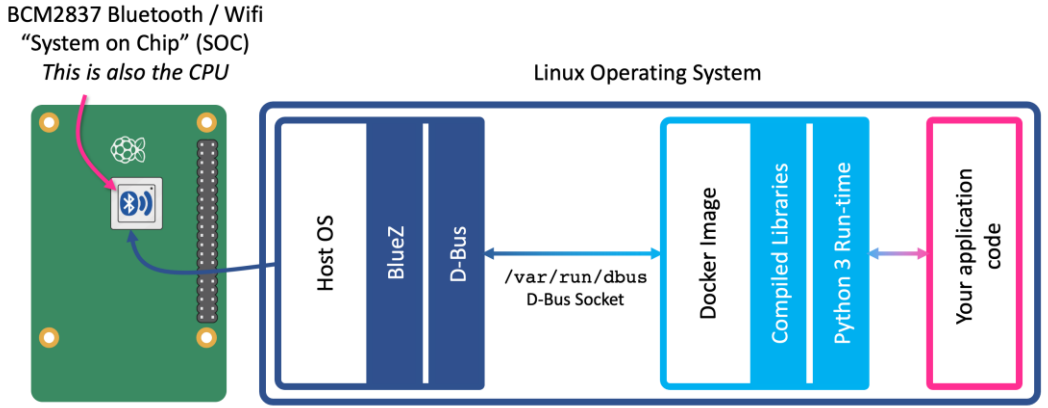
```
estheim@Machina:~$ ssh pi@192.168.0.101
pi@192.168.0.101's password:
Linux raspberrypi 4.14.79-v7+ #1159 SMP Sun Nov 4 17:50:20 GMT 2018 armv71

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Aug  4 22:40:41 2019 from 192.168.0.100
pi@raspberrypi:~ $ lsb_release -a
No LSB modules are available.
Distributor ID: Raspbian
Description:    Raspbian GNU/Linux 9.4 (stretch)
Release:        9.4
Codename:       stretch
pi@raspberrypi:~ $
```

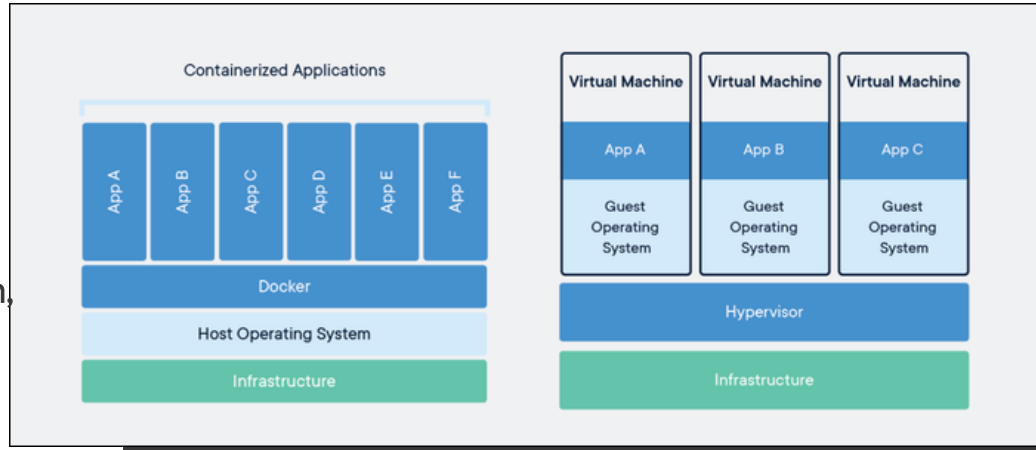
Docker Rpi

IoT Stack untuk docker pada raspberry pi

DOCKER

Docker adalah platform perangkat lunak yang memungkinkan Anda membuat, menguji, dan menerapkan aplikasi dengan cepat. Docker mengemas perangkat lunak ke dalam unit standar yang disebut **kontainer** yang memiliki semua yang diperlukan perangkat lunak agar dapat berfungsi termasuk **pustaka, alat sistem, kode, dan waktu proses**.



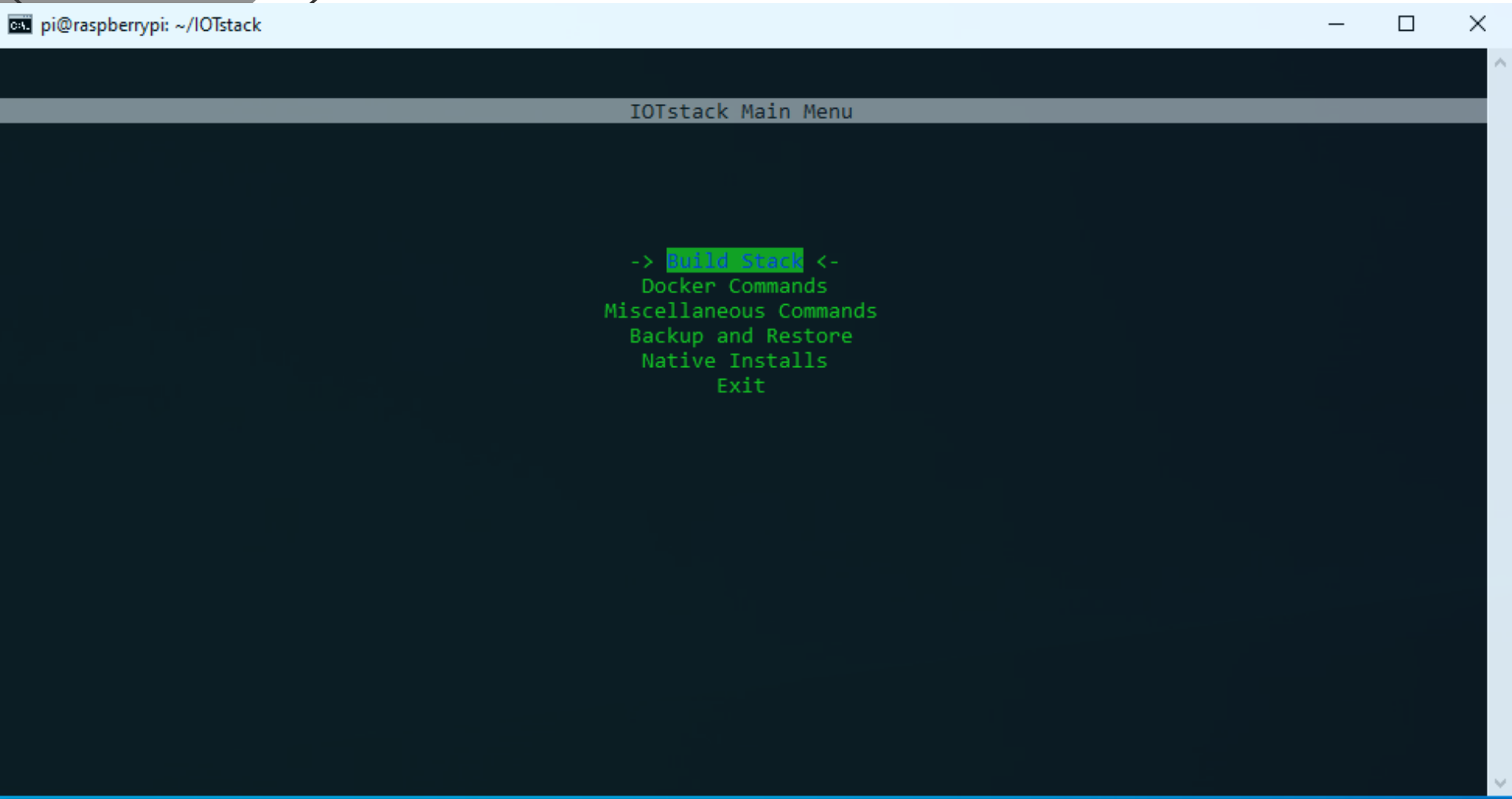


Install dan run docker

Connect to Rpi ssh pi@raspberrypi.local

1. Install curl:
sudo apt install -y curl
Run the following command:
curl -fsSL https://raw.githubusercontent.com/SensorsIot/IOTstack/master/install.sh | bash

2. Run the menu and choose your containers:
cd ~/IOTstack
./menu.sh
Bring up your stack:
cd ~/IOTstack
docker-compose up -d


pi@raspberrypi: ~/IOTstack
IOTstack Main Menu
-> Build Stack <-
Docker Commands
Miscellaneous Commands
Backup and Restore
Native Installs
Exit

Service to install

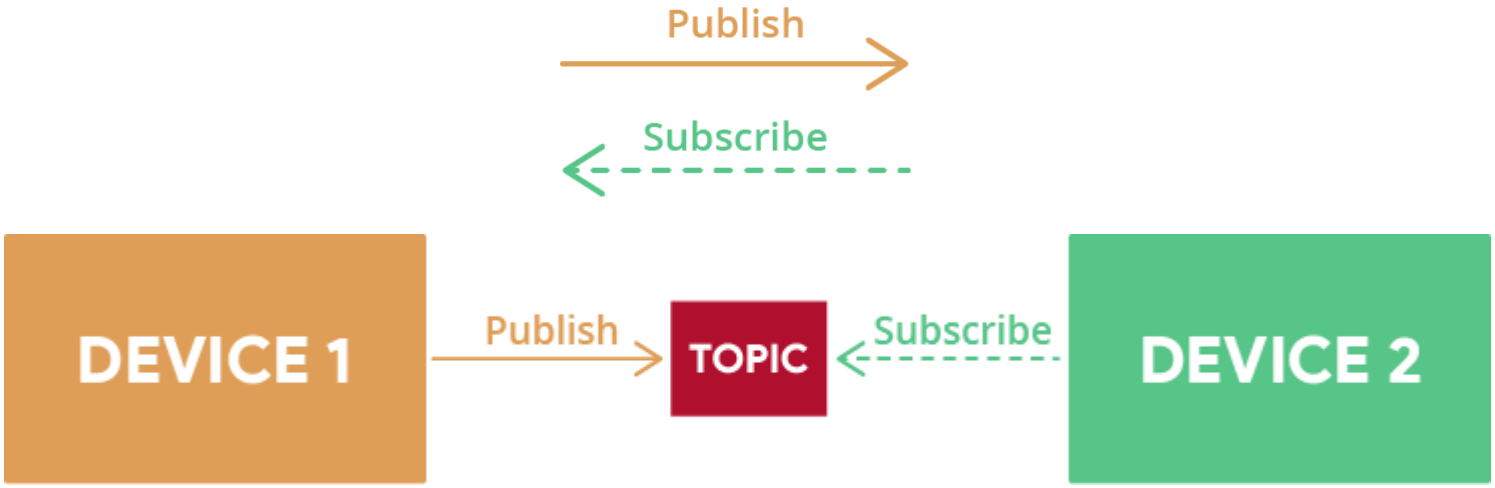
- Nodered
- Portainer-ce
- Mosquito
- Influxdb
- Grafana

MQTT SERVER

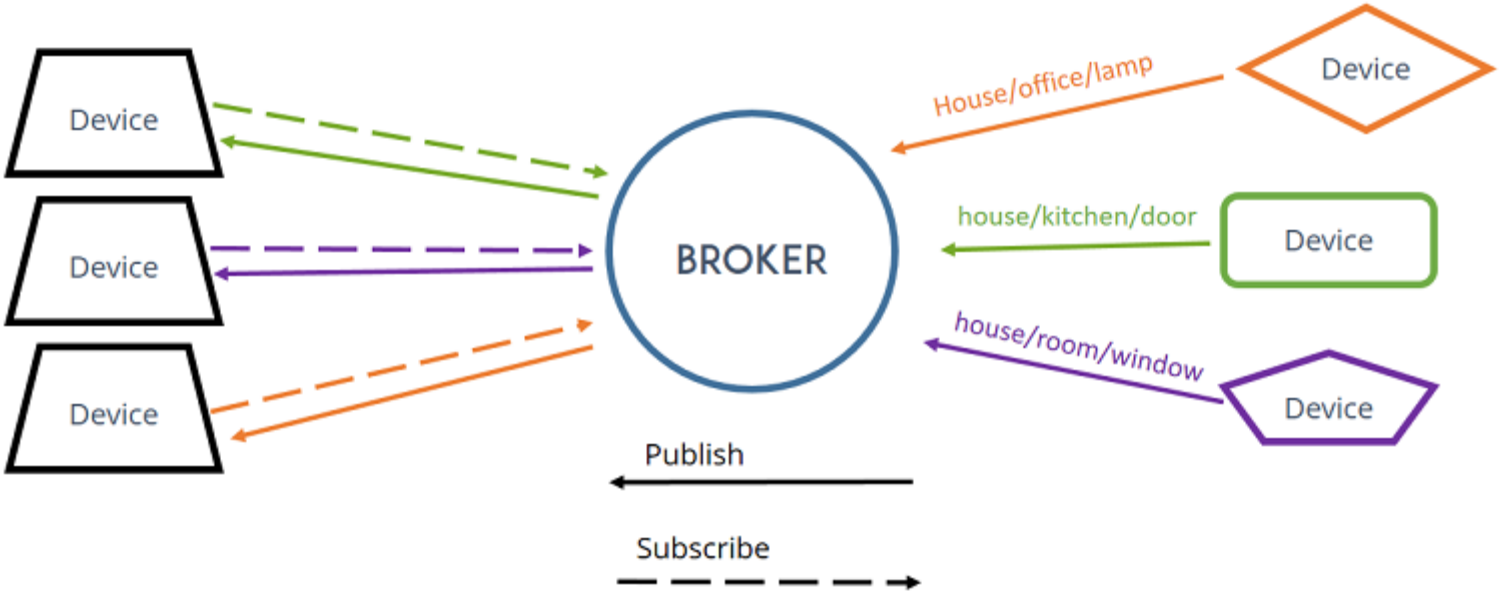
Mosquito

MQTT

MQTT stands for Message Queuing Telemetry Transport.



MQTT Broker


Device
Device
Device
BROKER
House/office/lamp
Device
house/kitchen/door
Device
house/room/window
Device
Publish
Subscribe

Tentang Topic

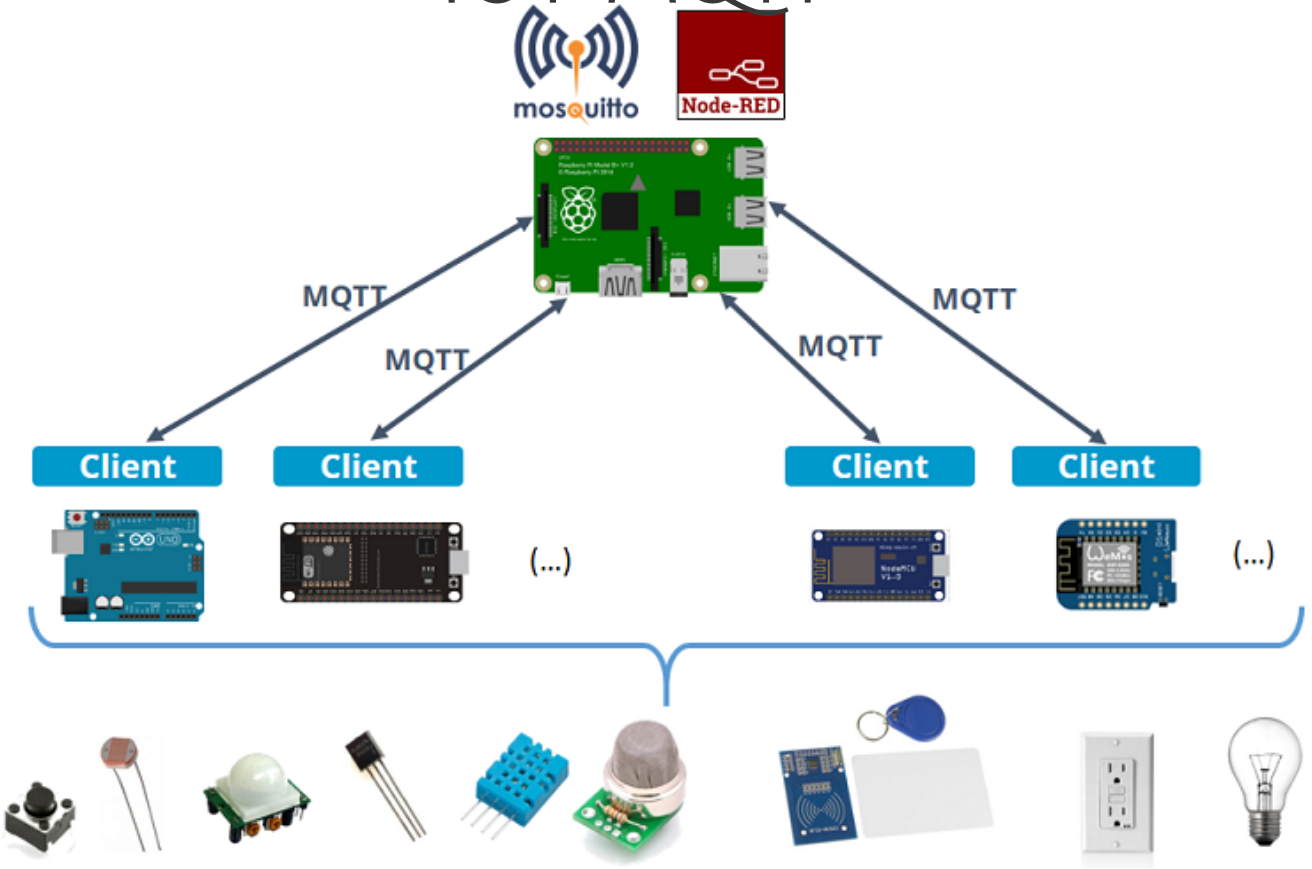
- Struktur dibatasi “/” misal
  - /workshop/esp32/sensor/status
  - /workshop/esp32/sensor/control
  - /workshop/esp32/sensordht11
  - /workshop/room/alarm

- Wild card topic dan subtopic
  - # (hash character) – multi level wildcard
  - + (plus character) -single level wildcard
- Case sensitive

Mosquitto

TOPIC	VALUE
$SYS/broker/version	mosquitto version 1.4.10
$SYS/broker/timestamp	Fri, 22 Dec 2017 08:19:25 +0000
$SYS/broker/uptime	2251623 seconds
$SYS/broker/clients/total	2
$SYS/broker/clients/inactive	0
$SYS/broker/clients/disconnected	0
$SYS/broker/clients/active	2
$SYS/broker/clients/connected	2
$SYS/broker/clients/expired	0
$SYS/broker/clients/maximum	3
$SYS/broker/messages/stored	61
$SYS/broker/messages/received	419773

IOT MQTT

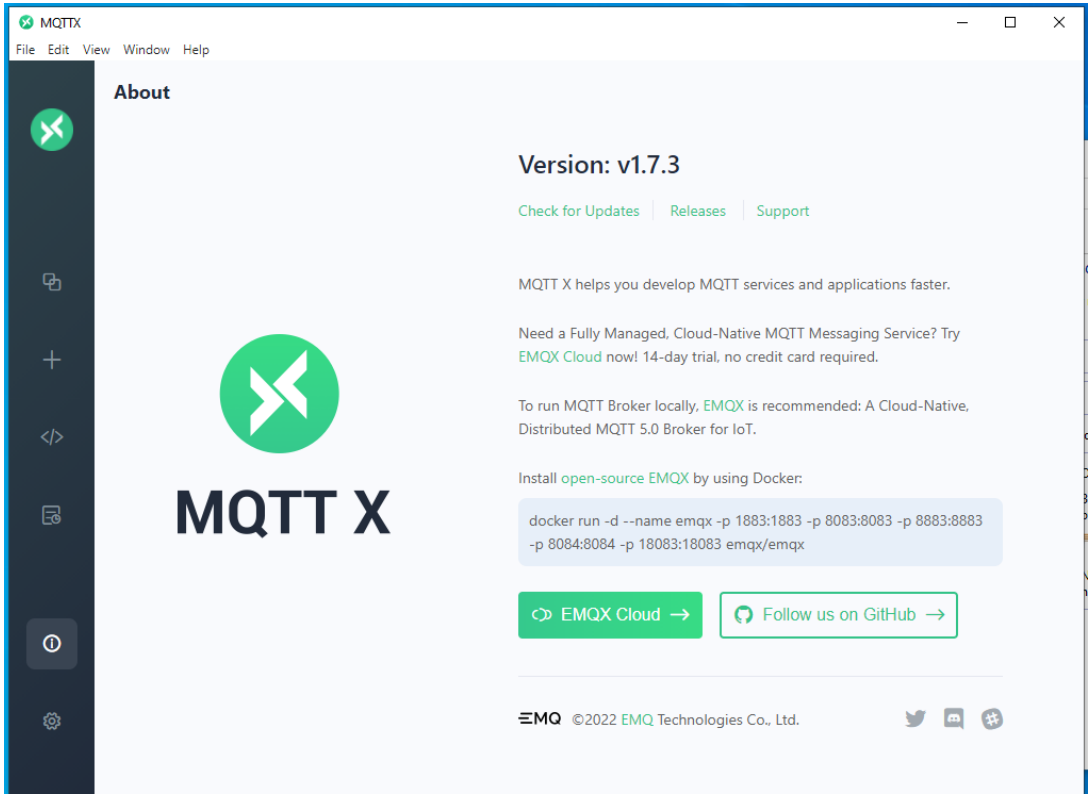

mosquitto
Node-RED
MQTT
MQTT
MQTT
MQTT
Client
Client
Client
Client
(...)
(...)

ESP MQTT

```
// WiFi
const char *ssid = “MOBILESUITE"; // Enter your WiFi name
const char *password = "88888888.";  // Enter WiFi password

// MQTT Broker
const char *mqtt_broker = "broker.emqx.io";
const char *topic = “workshop/esp32/sensor/";
const char *mqtt_username = "emqx";
const char *mqtt_password = "public";
const int mqtt_port = 1883;
```

```
lib_deps =
    ;adafruit/DHT sensor library@^1.4.3
    ;adafruit/Adafruit Unified Sensor@^1.1.5
    knolleary/PubSubClient@^2.8
```

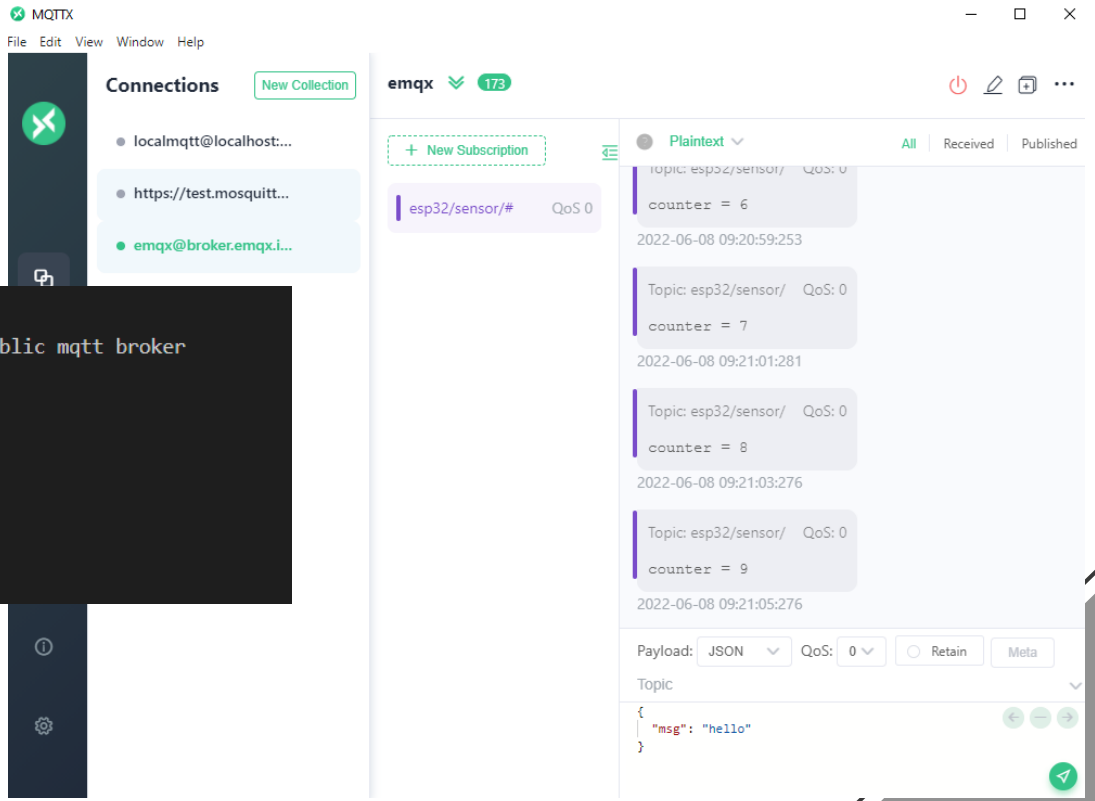
MQTT Client


MQTTX
File Edit View Window Help
About
Version: v1.7.3
Check for Updates | Releases | Support
MQTT X helps you develop MQTT services and applications faster.
Need a Fully Managed, Cloud-Native MQTT Messaging Service? Try EMQX Cloud now! 14-day trial, no credit card required.
To run MQTT Broker locally, EMQX is recommended: A Cloud-Native, Distributed MQTT 5.0 Broker for IoT.
Install open-source EMQX by using Docker:
docker run -d --name emqx -p 1883:1883 -p 8083:8083 -p 8883:8883 -p 8084:8084 -p 18083:18083 emqx/emqx
MQTT X
EMQX Cloud →
Follow us on GitHub →
©2022 EMQ Technologies Co., Ltd.

TEST MQTT

FROM ESP32

```
Connecting to WiFi..
Connected to the WiFi network
The client esp32-client-7C:9E:BD:48:61:CC connects to the public mqtt broker
Public emqx mqtt broker connected
Message arrived in topic: esp32/sensor/
Message:counter = 0
-----------------------
Message arrived in topic: esp32/sensor/
Message:counter = 1
-----------------------
Message arrived in topic: esp32/sensor/
Message:counter = 2
```

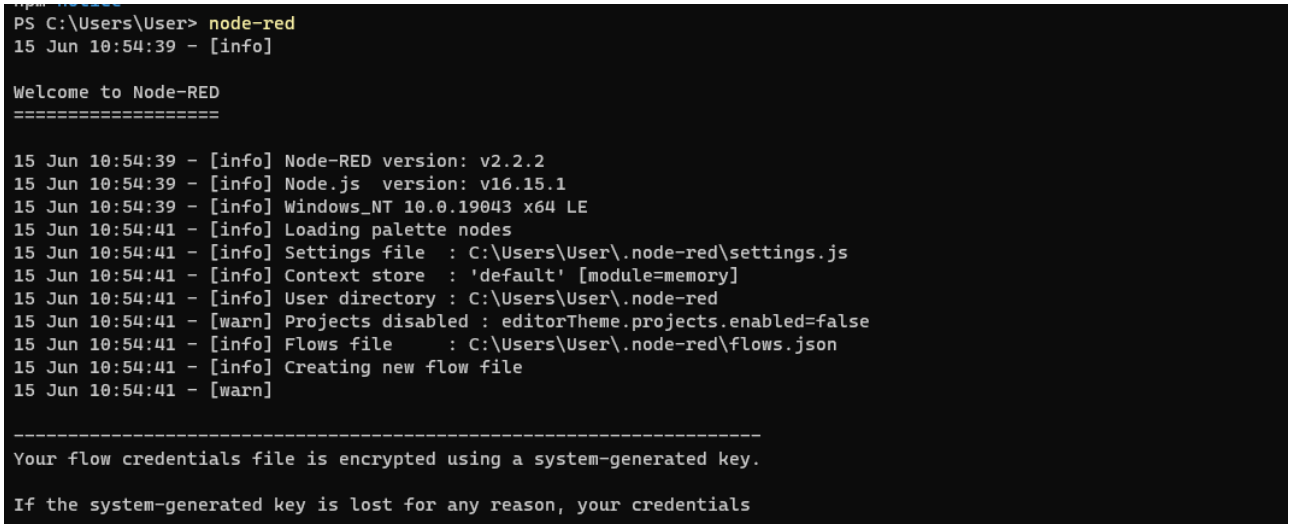


Node red dashboard

Dashboard mqtt mengunakan node-red dashboard

Install node red in pc

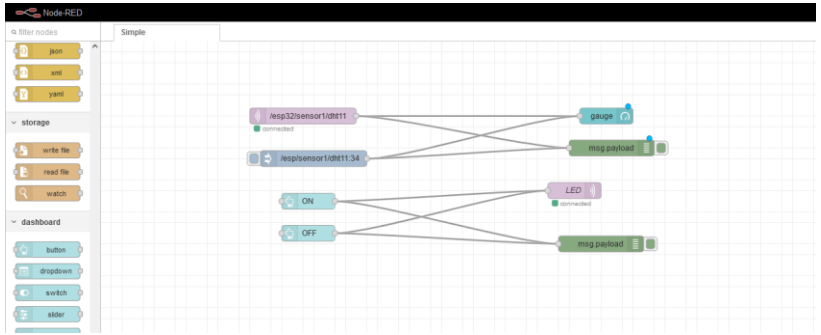
https://nodered.org/docs/getting-started/local

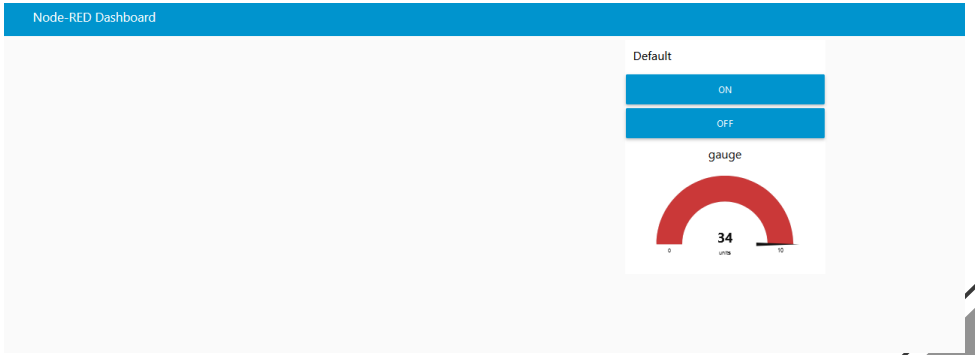


AKSES WEB

Node red flow
http://localhost:1880

Node red UI
http://localhost:1880/ui





Setting Node MQTT

```
client.subscribe(topic);
# dalam callback subscibe
if (payload[0]==1) {
    digitalWrite(18, HIGH);
 } else if (payload[0]==0) {
    digitalWrite(18, LOW);
 }
```

```
sprintf(str, “%i", datasensor );
client.publish(topic, str);
```

THANKS

Do you have any question?

hasbiida@gmail.com