Configuring a Raspberry PI and installing the IOTStack

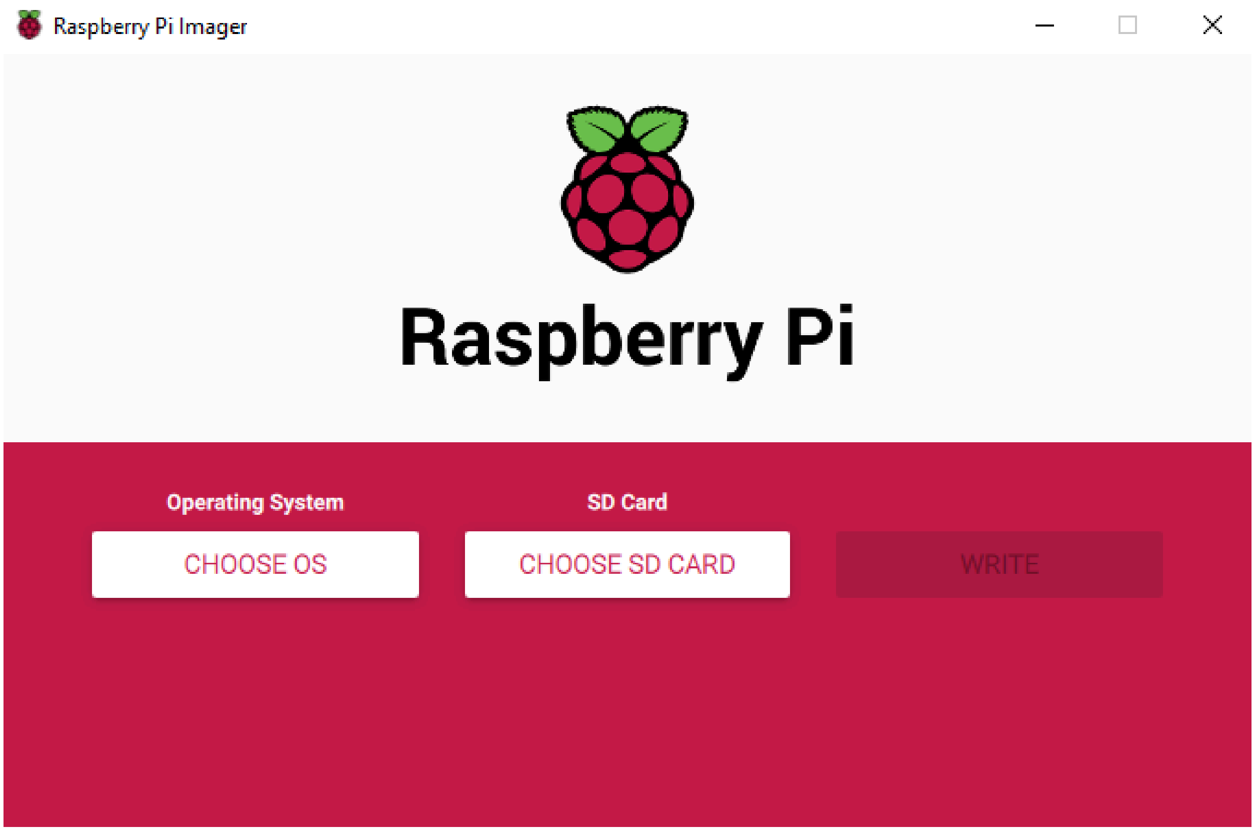
Nov 5, 2024 - Joseph Azar

Shortcuts:

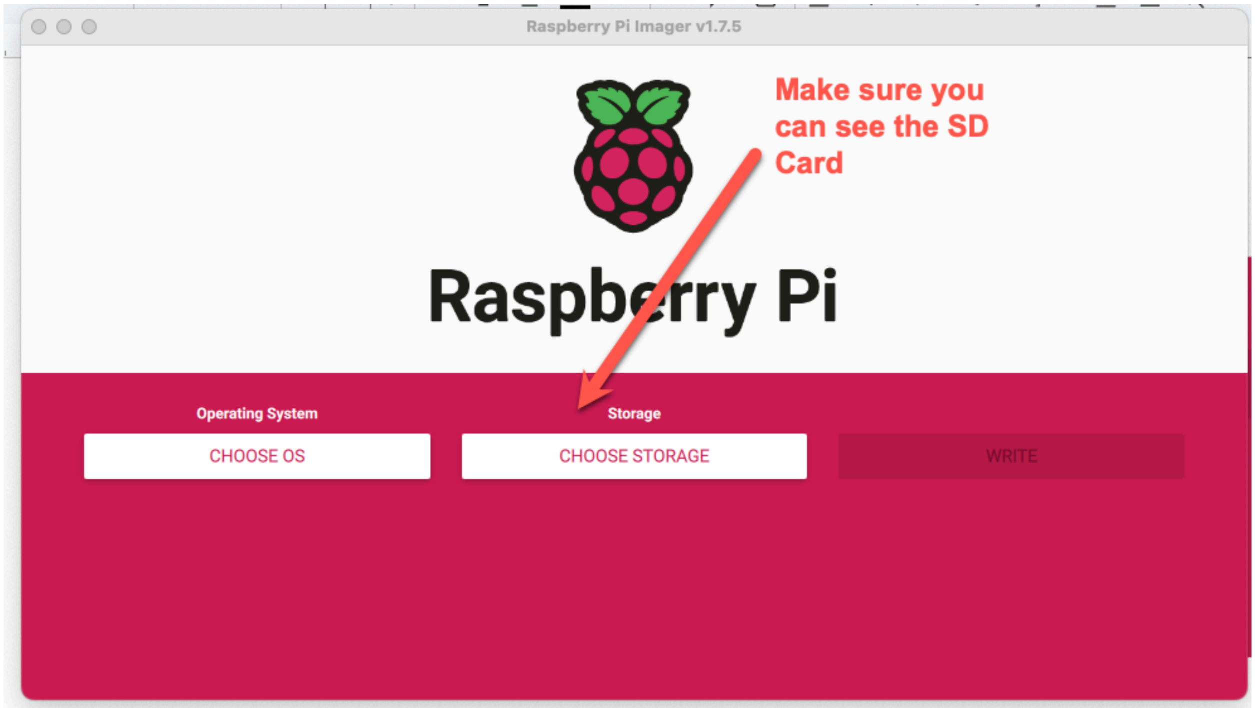
- Webmin: https://<hostname>.local:10000/
- Portainer: http://rpibadevel1.local:9000/
  - Create a password for admin user
- Node-red: http://rpibadevel1.local:1880/
- Grafana: http://rpibadevel1.local:3000/
- Influxdb:
  - docker exec -it influxdb influx
  - show databases
  - create database myhome
  - use myhome

Installing OS on SD CARD

Tool: Raspberry Pi Imager (https://www.raspberrypi.com/software/)

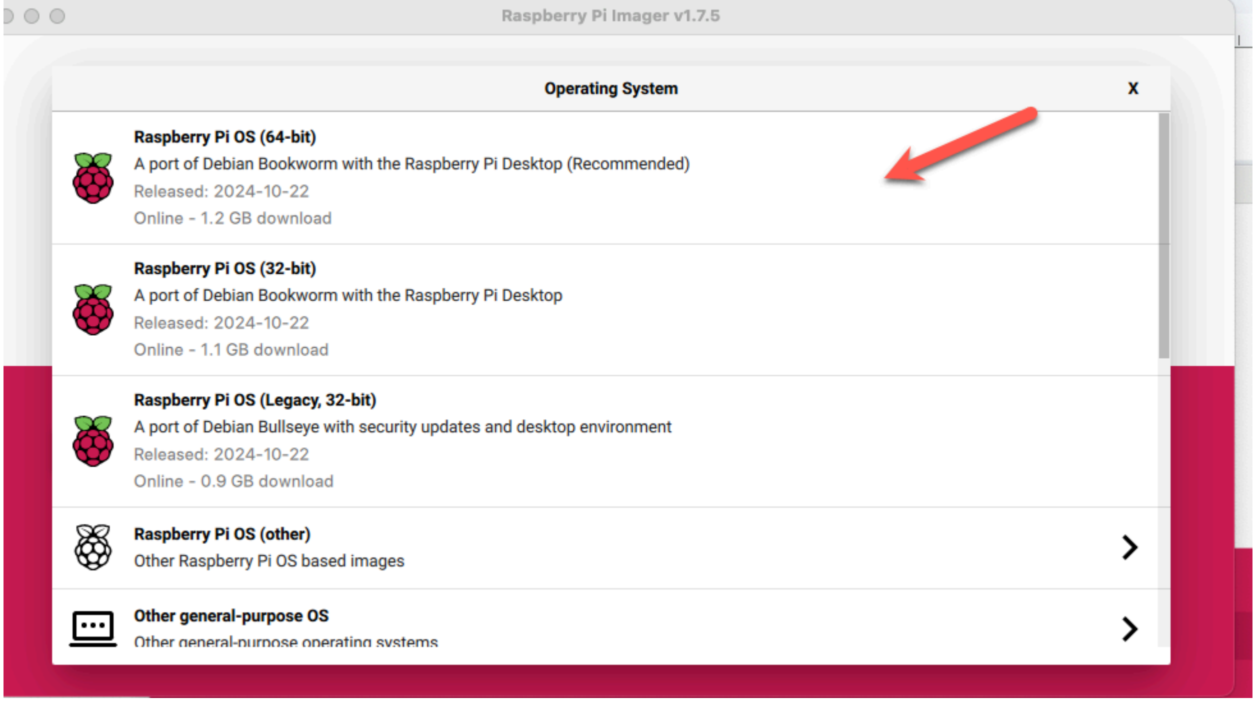


OS: Install Raspbian OS on the SD Card



Make sure you can see the SD Card

Choose the OS with desktop (recommended one)



Raspberry Pi OS (64-bit)
A port of Debian Bookworm with the Raspberry Pi Desktop (Recommended)
Released: 2024-10-22
Online - 1.2 GB download

Raspberry Pi OS (32-bit)
A port of Debian Bookworm with the Raspberry Pi Desktop
Released: 2024-10-22
Online - 1.1 GB download

Raspberry Pi OS (Legacy, 32-bit)
A port of Debian Bullseye with security updates and desktop environment
Released: 2024-10-22
Online - 0.9 GB download

Raspberry Pi OS (other)
Other Raspberry Pi OS based images

Other general-purpose OS
Other general-purpose operating systems

The first time you connect to the Raspberry PI, you will be using SSH.
So, choose a username and a password.

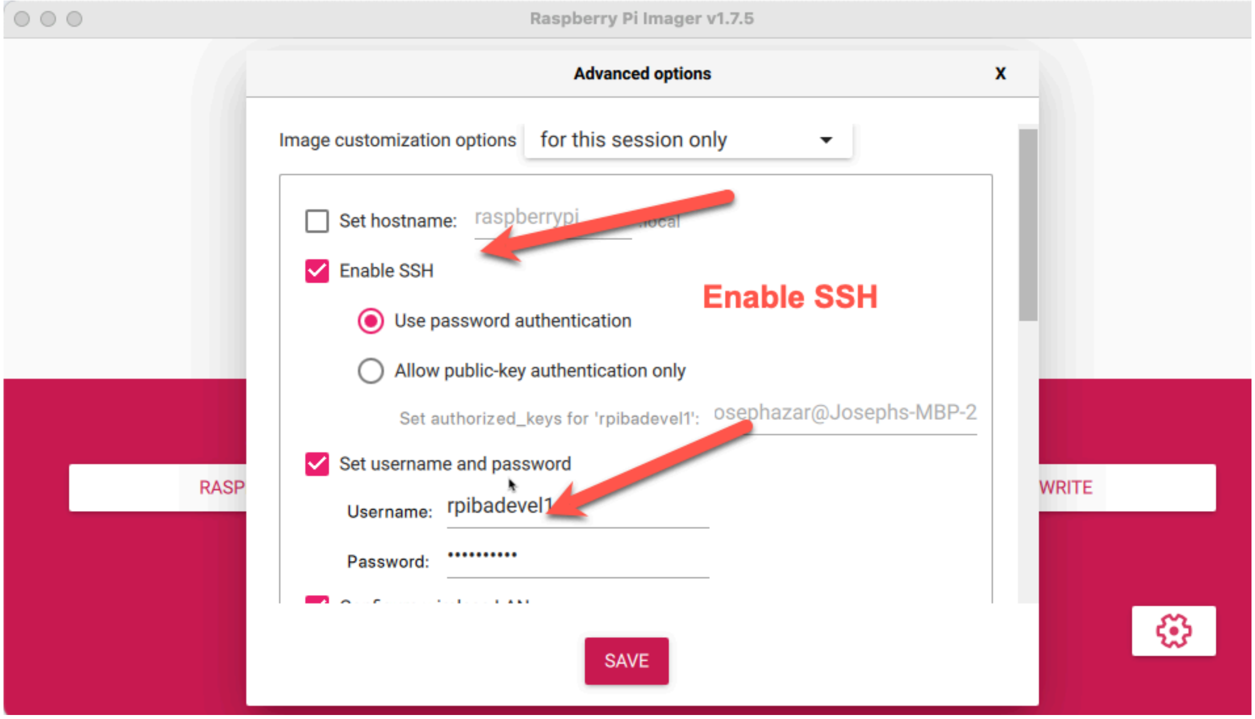


Image customization options: for this session only

☐ Set hostname: raspberrypi .local

☑ Enable SSH

● Use password authentication

○ Allow public-key authentication only

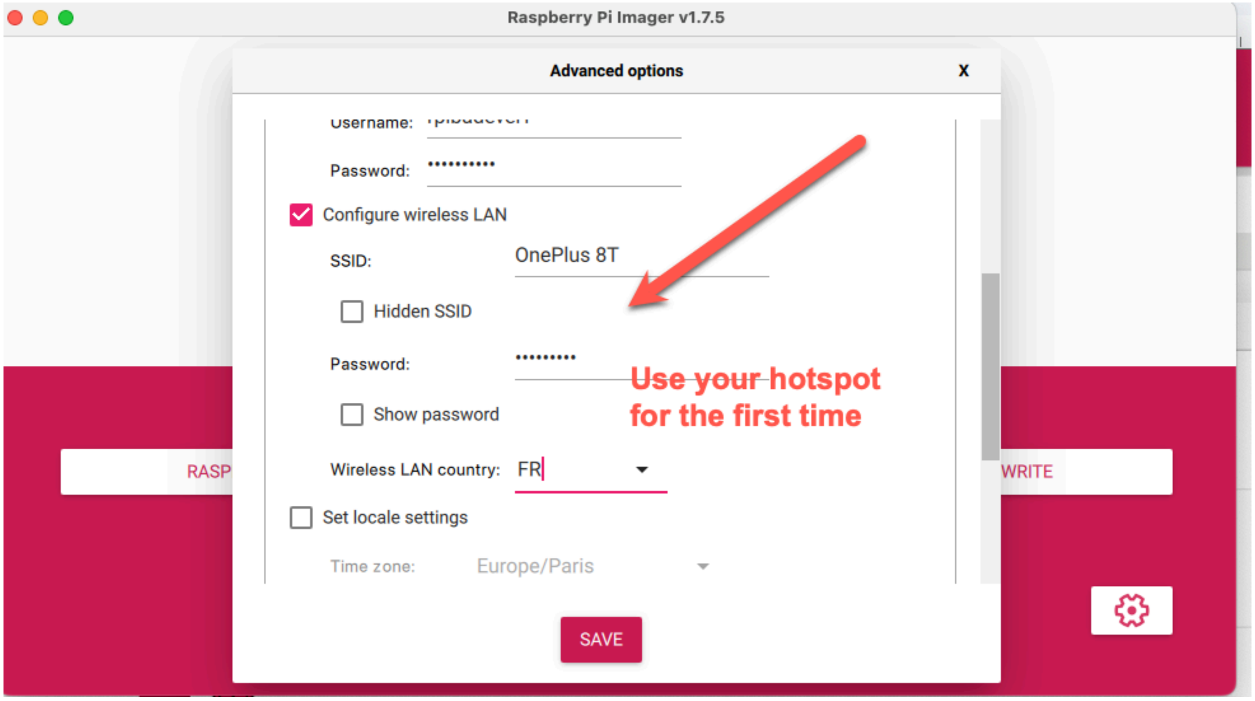
Set authorized_keys for 'rpibadevel1': osephazar@Josephs-MBP-2

☑ Set username and password

Username: rpibadevel1

Password: ••••••••••

Enable SSH



Username: rpibadevel1

Password: ••••••••••

☑ Configure wireless LAN

SSID: OnePlus 8T

□ Hidden SSID

Password: •••••••••

□ Show password

Wireless LAN country: FR

□ Set locale settings

Time zone: Europe/Paris

Use your hotspot for the first time

Connect your PC to the hotspot or Wifi where the RPI is connected.

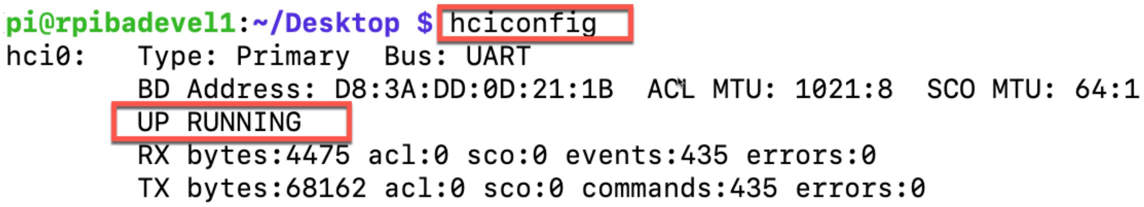
Connect your PC to the RPI

Ssh <username>@<machine>
Ex: ssh pi@rpibadevel1.local (make sure you replace rpibadevel1 with your hostname)
Password: < you should be asked for a password>

Update the system

Open a terminal
Sudo apt update
Sudo apt upgrade -y

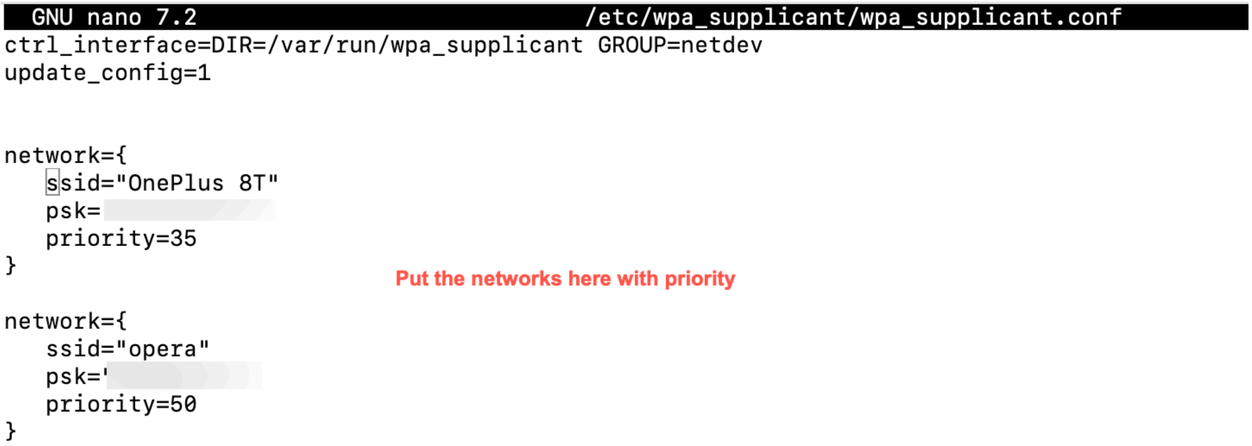
Make sure bluetooth is OK



```
pi@rpibadevel1:~/Desktop $ hciconfig
hci0:   Type: Primary  Bus: UART
        BD Address: D8:3A:DD:0D:21:1B  ACL MTU: 1021:8  SCO MTU: 64:1
        UP RUNNING
        RX bytes:4475 acl:0 sco:0 events:435 errors:0
        TX bytes:68162 acl:0 sco:0 commands:435 errors:0
```

Configure WiFi

sudo nano /etc/wpa_supplicant/wpa_supplicant.conf



```
  GNU nano 7.2                     /etc/wpa_supplicant/wpa_supplicant.conf
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1


network={
   ssid="OnePlus 8T"
   psk=[redacted]
   priority=35
}

network={
   ssid="opera"
   psk='[redacted]
   priority=50
}
```

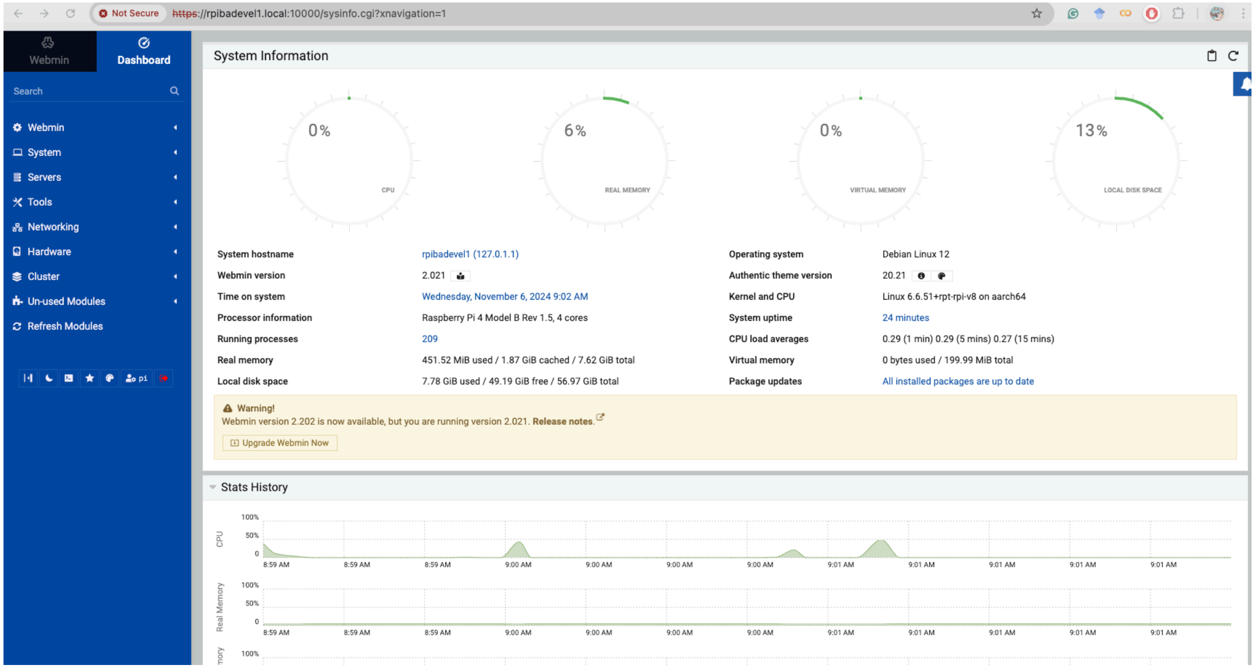
Put the networks here with priority

sudo systemctl restart wpa_supplicant.service
sudo service networking restart

Installing Webmin to monitor the health of the Raspberry PI

$ wget http://prdownloads.sourceforge.net/webadmin/webmin_2.021_all.deb
$ sudo apt --fix-broken install
$ sudo dpkg -i webmin_2.021_all.deb
$ sudo apt -f install
$ service webmin status

From your pc, try to access webmin using your browser:



System Information

System hostname	rpibadevel1 (127.0.1.1)	Operating system	Debian Linux 12
Webmin version	2.021	Authentic theme version	20.21
Time on system	Wednesday, November 6, 2024 9:02 AM	Kernel and CPU	Linux 6.6.51+rpt-rpi-v8 on aarch64
Processor information	Raspberry Pi 4 Model B Rev 1.5, 4 cores	System uptime	24 minutes
Running processes	209	CPU load averages	0.29 (1 min) 0.29 (5 mins) 0.27 (15 mins)
Real memory	451.52 MiB used / 1.87 GiB cached / 7.62 GiB total	Virtual memory	0 bytes used / 199.99 MiB total
Local disk space	7.78 GiB used / 49.19 GiB free / 56.97 GiB total	Package updates	All installed packages are up to date

⚠ Warning!
Webmin version 2.202 is now available, but you are running version 2.021. **Release notes.**

Upgrade Webmin Now

Installing the IoT Stack

Reference: https://sensorsiot.github.io/IOTstack/Basic_setup/

```
sudo apt install -y curl
curl -fsSL https://raw.githubusercontent.com/SensorsIot/IOTstack/master/install.sh | bash
```

From menu.sh, install InfluxDB, MQTT, Grafana, Node-RED & Docker Portainer
Follow this tutorial: https://www.youtube.com/watch?v=_DO2wHI6JWQ&t=441s

```
cd ~/IOTstack
./menu.sh
Choose:
   ●  Node red (Automation)
   ●  Mosquitto (MQTT)
   ●  Grafana (visualization)
   ●  Influxdb (database)
docker-compose up -d
```

You can go through the menu using ./menu.sh and choose the softwares you want to install same as in the youtube video above. Or you can create a `docker-compose.yml file in the ~/IOTStack folder and copy the below into it and do docker-compose up -d.`

Example of docker-compose.yml

```
version: '3.6'

networks:
  default:
    driver: bridge
    ipam:
      driver: default
  nextcloud:
    driver: bridge
    internal: true
    ipam:
      driver: default

services:
  grafana:
    container_name: grafana
    image: grafana/grafana
    restart: unless-stopped
    user: "0"
    ports:
    - "3000:3000"
    environment:
    - TZ=Europe/Paris
    - GF_PATHS_DATA=/var/lib/grafana
    - GF_PATHS_LOGS=/var/log/grafana
    volumes:
    - ./volumes/grafana/data:/var/lib/grafana
    - ./volumes/grafana/log:/var/log/grafana
    healthcheck:
      test: ["CMD", "wget", "-O", "/dev/null", "http://localhost:3000"]
      interval: 30s
      timeout: 10s
      retries: 3
      start_period: 30s

  influxdb:
    container_name: influxdb
    image: "influxdb:1.8"
    restart: unless-stopped
    ports:
    - "8086:8086"
    environment:
    - TZ=Europe/Paris
    - INFLUXDB_HTTP_FLUX_ENABLED=false
    - INFLUXDB_REPORTING_DISABLED=false
    - INFLUXDB_HTTP_AUTH_ENABLED=false
    - INFLUXDB_MONITOR_STORE_ENABLED=FALSE
  # - INFLUX_USERNAME=dba
  # - INFLUX_PASSWORD=supremo
  # - INFLUXDB_UDP_ENABLED=false
  # - INFLUXDB_UDP_BIND_ADDRESS=0.0.0.0:8086
  # - INFLUXDB_UDP_DATABASE=udp
    volumes:
    - ./volumes/influxdb/data:/var/lib/influxdb
    - ./backups/influxdb/db:/var/lib/influxdb/backup
    healthcheck:
      test: ["CMD", "curl", "http://localhost:8086"]
      interval: 30s
      timeout: 10s
      retries: 3
      start_period: 30s

  mosquitto:
    container_name: mosquitto
    build:
      context: ./.templates/mosquitto/.
      args:
      - MOSQUITTO_BASE=eclipse-mosquitto:latest
    restart: unless-stopped
    environment:
    - TZ=Europe/Paris
    ports:
    - "1883:1883"
    volumes:
    - ./volumes/mosquitto/config:/mosquitto/config
    - ./volumes/mosquitto/data:/mosquitto/data
    - ./volumes/mosquitto/log:/mosquitto/log
    - ./volumes/mosquitto/pwfile:/mosquitto/pwfile


  portainer-ce:
    container_name: portainer-ce
```

```
    image: portainer/portainer-ce
    restart: unless-stopped
    ports:
    - "8000:8000"
    - "9000:9000"
    # HTTPS
    - "9443:9443"
    volumes:
    - /var/run/docker.sock:/var/run/docker.sock
    - ./volumes/portainer-ce/data:/data

  nodered:
    container_name: nodered
    build:
      context: ./services/nodered/.
      args:
      - DOCKERHUB_TAG=latest
      - EXTRA_PACKAGES=
    restart: unless-stopped
    user: "0"
    environment:
    - TZ=Europe/Paris
    ports:
    - "1880:1880"
    volumes:
    - ./volumes/nodered/data:/data
    - ./volumes/nodered/ssh:/root/.ssh
    - /var/run/docker.sock:/var/run/docker.sock
    - /var/run/dbus/system_bus_socket:/var/run/dbus/system_bus_socket
    devices:
    - "/dev/vcio:/dev/vcio"
    - "/dev/gpiomem:/dev/gpiomem"
```