

Raspberry Pi

Home Assistant op de RPi in docker (1)






Agenda

- Doel van deze sessie
 - Boot Image voor RPi maken
 - Docker installeren

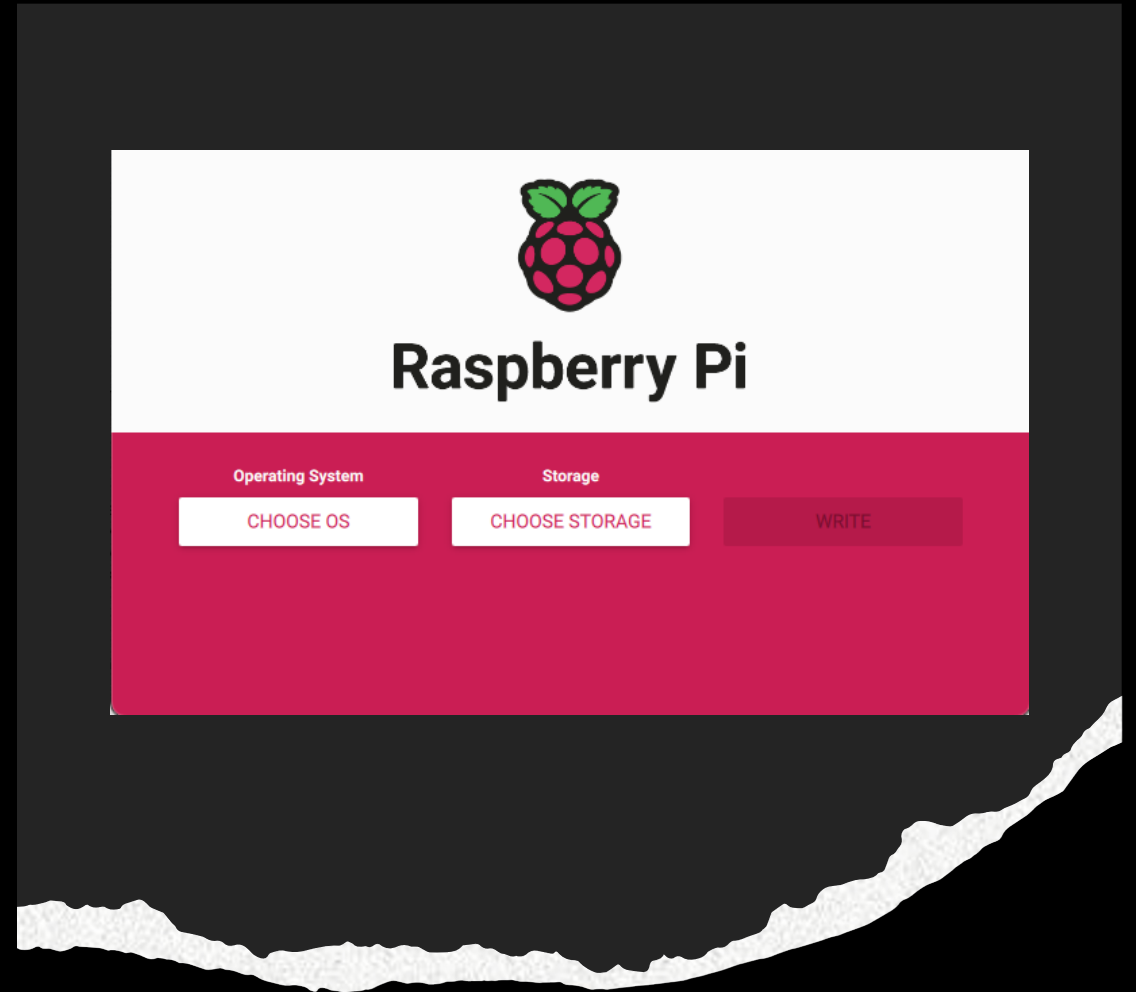
 - Home Assistant via **docker compose**
 - Mosquitto MQTT via **docker compose**
 - Node-RED via **docker compose**
- 



OS op RPi

- Gebruik RPi Imager om OS op SD Card te schrijven
 - Gebruik Ubuntu 22.04 LTS server image
 - Maak gebruik van SSH en WiFi (Advanced Settings)
 - Maak een user-account (Advanced Settings)
- 

Raspberry Pi Imager



<https://www.raspberrypi.com/software/>

Advanced Settings

Advanced options X

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 'nicoo': 📄@Nicos-MacBook-Pro.local

Set username and password

SAVE



Boot RPi van USB



Login m.b.v. terminal/cmd

ssh user@homeassistant.local

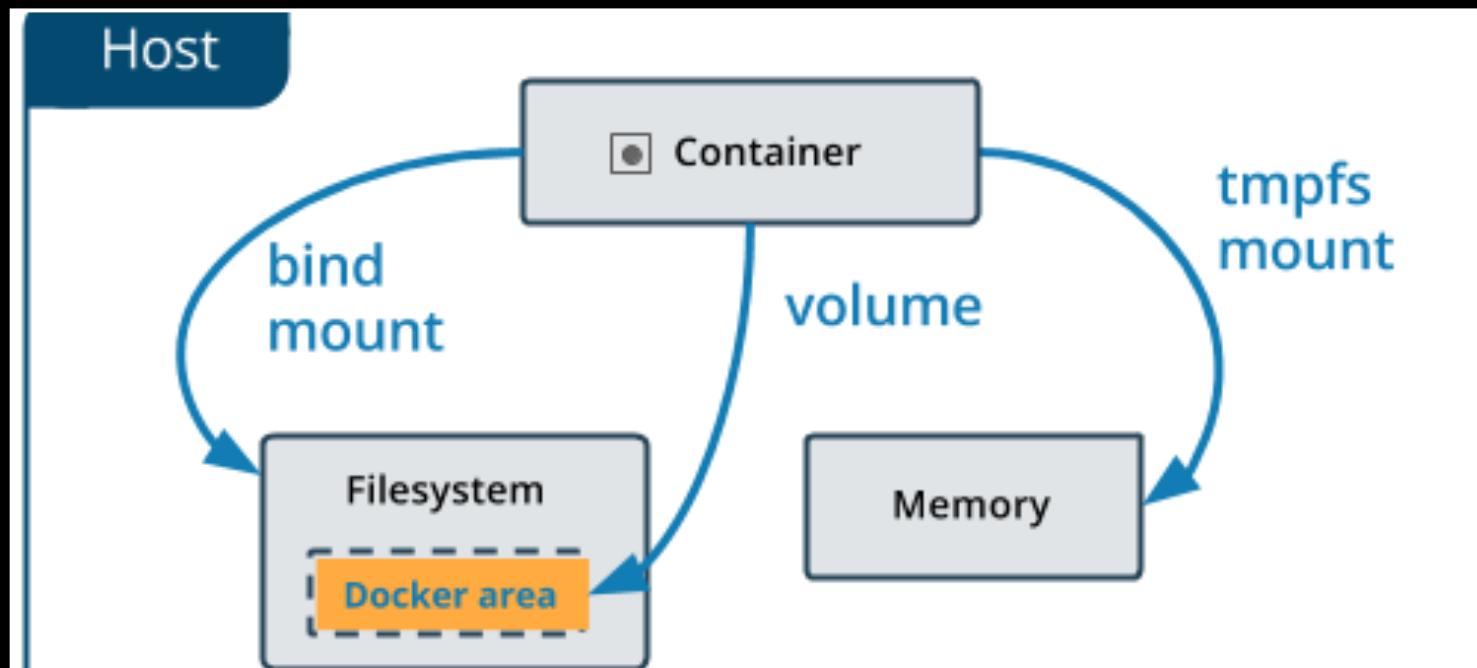


Wat is docker?

- Docker is een open source containerization technologie voor het bouwen en beheren van applicaties in een container.
- Een docker server via dockerd
- Een docker client (CLI) met docker

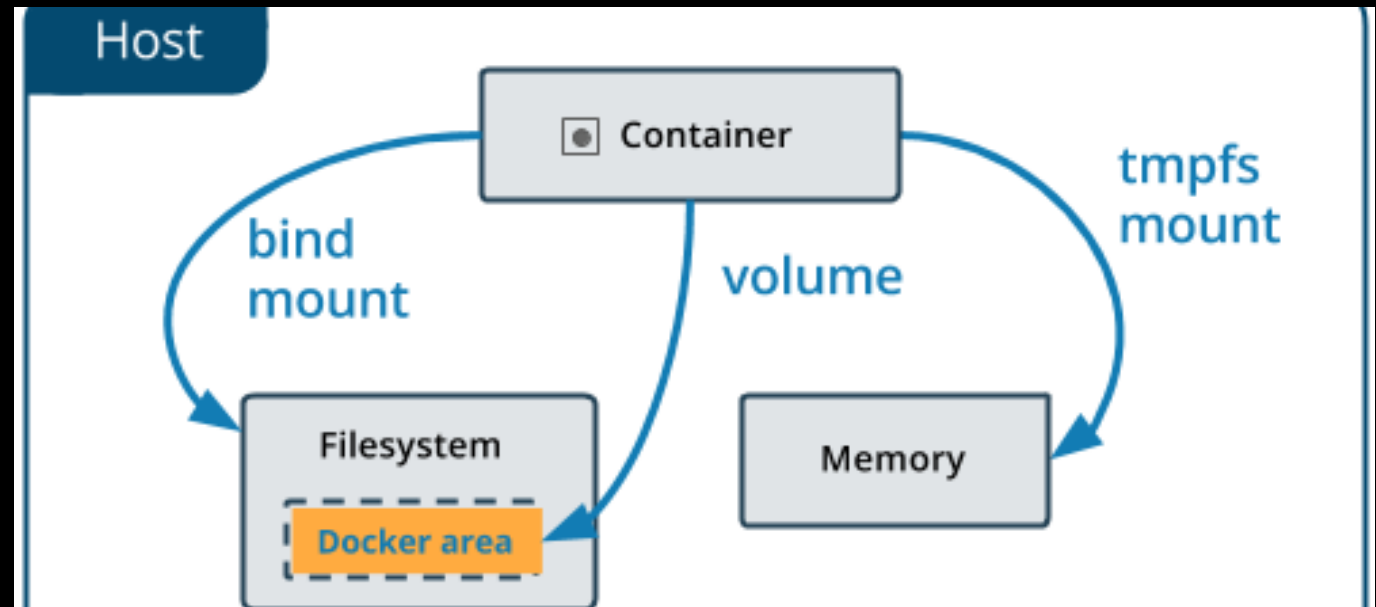
Docker volumes

-v heeft 3 velden,
1e = naam van volume,
2e = path in de container,
3e = optioneel (bv. r/o)



Docker bind mounts

- -v heeft 3 Velden:
- 1e = path op de host,
- 2e = path in de container,
- 3e = optioneel (bv. r/o)



A diagram illustrating Docker networks. It features a large black rectangle with a white border. Inside the rectangle, the text "Docker networks" is written in white. On the left side of the rectangle, there are four white zigzag lines. At the top center of the rectangle, there is a white circle with a black outline. At the bottom right corner of the rectangle, there is a white circle with a black outline and diagonal hatching. A thick white line runs along the bottom edge of the rectangle, extending slightly beyond its right side.

Docker networks

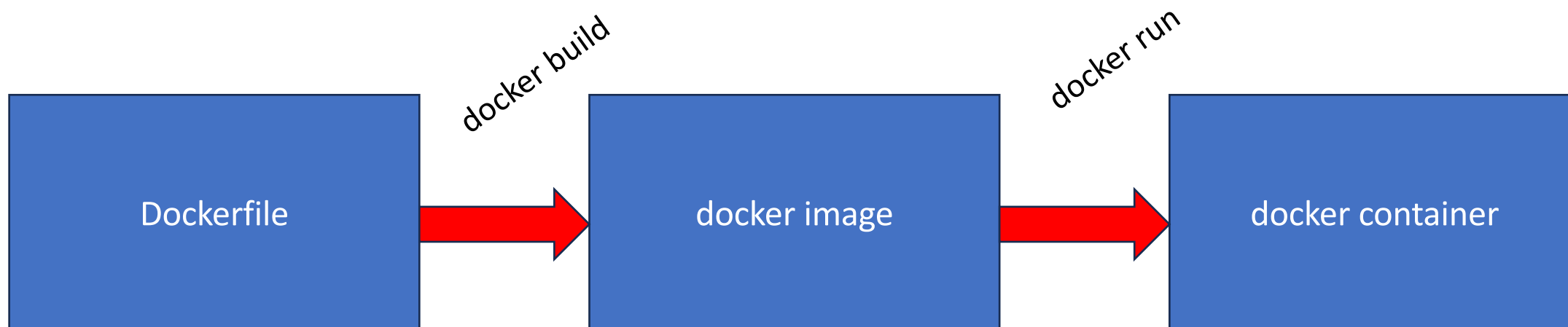
- Bridge
 - Op een enkele RPi is de default om een 'bridge' driver te gebruiken. Bridge is een virtuele netwerk-switch.
- Host
 - Host netwerk driver maakt gebruik van het netwerk van de host (RPi)

Installeren van Docker

Gebruik de aanwijzingen op
docs.docker.com/engine/install/ubuntu



Docker basics






Dockerfile

```
FROM bash
```

```
CMD ["ping","localhost"]
```

```
docker build -t simple .
```

```
docker image ls
```






docker run

```
docker run --name c1 -d bash:latest sh -c 'sleep 1d'
```

```
docker ps
```


```
docker stop c1
```

```
docker ps
```



Home Assistant, MQTT en Node- RED in docker containers

- Home Assistant
 - MQTT berichten-systeem
 - Node-RED voor de automatiseringen van HA

 - Gebruik een docker-compose bestand en maak een service om docker compose op te starten als een **systemd** service
- 

Docker compose

- Voor het samenstellen van meerdere containers met bijbehorende opties.
- Starten van alle services via 'docker compose up'
- Bestaat uit een docker-compose yaml bestand

Vorbereiden mount bind mosquitto

```
persistence true  
persistence_location /mosquitto/data/  
log_dest file /mosquitto/log/mosquitto.log
```

- Maak bestand aan in `/mosquitto/config/mosquitto.conf` met bovenstaande inhoud

docker- compose.yaml

```
version: "3.7"

services:

    #####
    # Home Assistant #
    #####
    homeassistant:

        image: ghcr.io/home-assistant/home-assistant:stable
        container_name: homeassistant

        volumes:

            - haconfig:/config:rw
            - /etc/localtime:/etc/localtime:ro

        network_mode: host

        environment:

            - TZ=Europe/Amsterdam

        privileged: true

        restart: unless-stopped
```

docker- compose.yaml

```
#####  
# mosquitto #  
#####  
mosquitto:  
  image: eclipse-mosquitto  
  container_name: mosquitto  
  volumes:  
    - /mosquitto/config:/mosquitto/config:rw  
    - mosquitto-data:/mosquitto/data:rw  
    - mosquitto-log:/mosquitto/log:rw  
  network_mode: host  
  restart: unless-stopped
```

docker- compose.yaml

```
#####  
  
# node-RED #  
#####  
  
node-red:  
  image: nodered/node-red:latest  
  container_name: node-red  
  environment:  
    - TZ=Europe/Amsterdam  
  volumes:  
    - node-red-data:/data  
  network_mode: host  
  restart: unless-stopped  
  depends_on:  
    - homeassistant  
    - mosquitto  
  
volumes:  
  haconfig:  
    name: homeassistant  
  mosquitto-data:  
    name: mqtt-data  
  mosquitto-log:  
    name: mqtt-log  
  node-red-data:  
    name: node-red
```

Starten van de containers

docker compose up -d

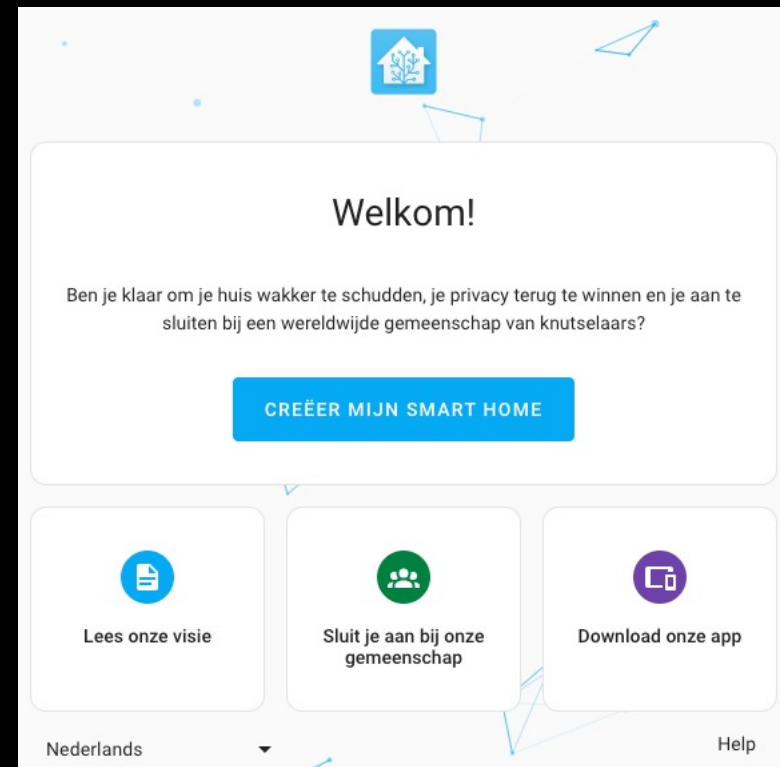
Controle

docker ps



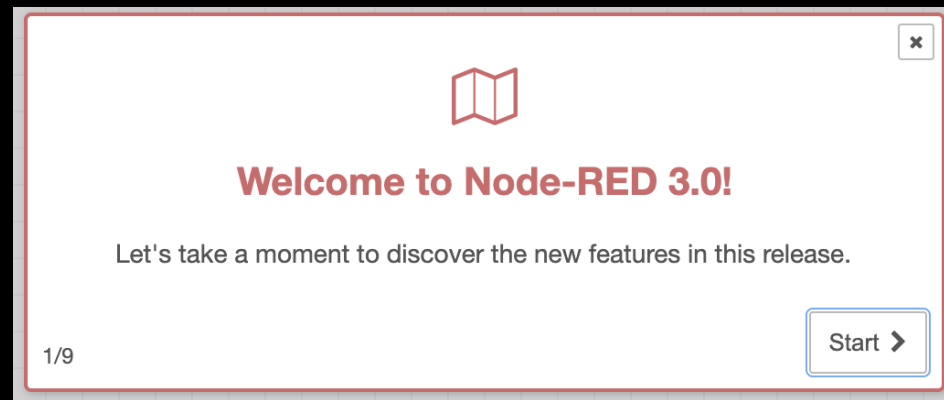
Home Assistant

- <http://192.168.2.80:8123>



Node-RED

<http://192.168.2.80:1880>



Vragen?