Raspberry Pi domotica farm

Table of Contents

Ubuntu-setup

Install ubuntu on pi

- ➔ Raspberry pi sd

- ➔ set password (asked during installation)
- ➔ apt update
- ➔ apt upgrade

Change ip to static address

On ubuntu these are set by netplan, modify /etc/netplan/50-cloud-init.yaml

```
vi /etc/netplan/50-cloud-init.yaml
# This file is generated from information provided by the datasource.  Changes
# to it will not persist across an instance reboot.  To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  ethernets:
    eth0:
      dhcp4: no
      addresses:
       - 192.168.0.140/12
      gateway4: 192.168.0.1
      nameservers:
        addresses: [109.88.203.3, 62.197.111.140]
      optional: true
  version: 2
~
~
```

Apply the changes:

```
root@ubuntu:~# netplan apply
root@ubuntu:~#
```

Change the hostname

```
root@ubuntu:~# hostnamectl set-hostname colom-master
root@ubuntu:~#
```

When you exit the session and reconnect the hostname should be changed in the lines

```
root@colom-master:~#
```

Docker Install

When ubuntu is configured, installation of docker. All actions for docker need to run as root: Suggest changing to root user:

```
sudo su -
```

Installation of docker with apt, don't use snap etc as it does not need to be isolated.

```
apt install docker.io
```

```
ubuntu@colom-master:~$ sudo su -
root@colom-master:~# apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils cgroupfs-mount containerd dns-root-data dnsmasq-base libidn11 pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils cgroupfs-mount containerd dns-root-data dnsmasq-base docker.io libidn11 pigz runc ubuntu-fan
0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.
Need to get 48.3 MB of archives.
After this operation, 271 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ports.ubuntu.com/ubuntu-ports focal/universe arm64 pigz arm64 2.4-1 [47.8 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 bridge-utils arm64 1.6-2ubuntu1 [30.6 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports focal/universe arm64 cgroupfs-mount all 1.4 [6320 B]
Get:4 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 runc arm64 1.0.0~rc10-0ubuntu1 [2249 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 containerd arm64 1.3.3-0ubuntu2 [19.7 MB]
Get:6 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 dns-root-data all 2019052802 [5300 B]
Get:7 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 libidn11 arm64 1.33-2.2ubuntu2 [45.3 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 dnsmasq-base arm64 2.80-1.1ubuntu1 [294 kB]
Get:9 http://ports.ubuntu.com/ubuntu-ports focal-updates/universe arm64 docker.io arm64 19.03.8-0ubuntu1.20.04 [25.9 MB]
Get:10 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 ubuntu-fan all 0.12.13 [34.5 kB]
Fetched 48.3 MB in 4s (13.1 MB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 99387 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_arm64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.6-2ubuntu1_arm64.deb ...
Unpacking bridge-utils (1.6-2ubuntu1) ...
Selecting previously unselected package cgroupfs-mount.
Preparing to unpack .../2-cgroupfs-mount_1.4_all.deb ...
Unpacking cgroupfs-mount (1.4) ...
Selecting previously unselected package runc.
Preparing to unpack .../3-runc_1.0.0~rc10-0ubuntu1_arm64.deb ...
Unpacking runc (1.0.0~rc10-0ubuntu1) ...
Selecting previously unselected package containerd.
Preparing to unpack .../4-containerd_1.3.3-0ubuntu2_arm64.deb ...
Unpacking containerd (1.3.3-0ubuntu2) ...
Selecting previously unselected package dns-root-data.
Preparing to unpack .../5-dns-root-data_2019052802_all.deb ...
Unpacking dns-root-data (2019052802) ...
Selecting previously unselected package libidn11:arm64.
Preparing to unpack .../6-libidn11_1.33-2.2ubuntu2_arm64.deb ...
Unpacking libidn11:arm64 (1.33-2.2ubuntu2) ...
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../7-dnsmasq-base_2.80-1.1ubuntu1_arm64.deb ...
Unpacking dnsmasq-base (2.80-1.1ubuntu1) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../8-docker.io_19.03.8-0ubuntu1.20.04_arm64.deb ...
Unpacking docker.io (19.03.8-0ubuntu1.20.04) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../9-ubuntu-fan_0.12.13_all.deb ...
Unpacking ubuntu-fan (0.12.13) ...
Setting up runc (1.0.0~rc10-0ubuntu1) ...
Setting up dns-root-data (2019052802) ...
Setting up libidn11:arm64 (1.33-2.2ubuntu2) ...
Setting up bridge-utils (1.6-2ubuntu1) ...
Setting up pigz (2.4-1) ...
Setting up cgroupfs-mount (1.4) ...
Setting up containerd (1.3.3-0ubuntu2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker.io (19.03.8-0ubuntu1.20.04) ...
Adding group `docker' (GID 119) ...
Done.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
docker.service is a disabled or a static unit, not starting it.
Setting up dnsmasq-base (2.80-1.1ubuntu1) ...
Setting up ubuntu-fan (0.12.13) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Processing triggers for systemd (245.4-4ubuntu3.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
root@colom-master:~#
```

Enable docker

```
systemctl enable docker
```

```
root@colom-master:~# systemctl enable --now docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
root@colom-master:~#
```

Check the version:

```
docker --version
```

```
root@colom-master:~# docker --version
Docker version 19.03.8, build afacb8b7f0
root@colom-master:~#
```

Test docker with the typical hello-world

```
docker run hello-world
```

```
root@colom-master:~# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
256ab8fe8778: Pull complete
Digest: sha256:4cf9c47f86df71d48364001ede3a4fcd85ae80ce02ebad74156906caff5378bc
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (arm64v8)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

root@colom-master:~#
```

Kubernetes Install

Kubernetes is only helpful if we want multiple nodes (or when the web-gui is required)

Install master

To install Kubernetes we'll need the curl and apt-transport-https packages (necessary to get a key to add the Kubernetes repositories)

```
apt install apt-transport-https curl
```

```
root@colom-master:/var/lib/docker# apt install apt-transport-https curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.68.0-1ubuntu2.2).
curl set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1708 B of archives.
After this operation, 160 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ports.ubuntu.com/ubuntu-ports focal-updates/universe arm64 apt-transport-https all 2.0.2ubuntu0.1 [1708 B]
Fetched 1708 B in 0s (16.8 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 99747 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.0.2ubuntu0.1_all.deb ...
Unpacking apt-transport-https (2.0.2ubuntu0.1) ...
Setting up apt-transport-https (2.0.2ubuntu0.1) ...
root@colom-master:/var/lib/docker#
```

Add the repository keys.

```
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add
```

```
root@colom-master:/var/lib/docker# curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add
OK
root@colom-master:/var/lib/docker#
```

Add the repository.

```
apt-add-repository "deb http://apt.kubernetes.io/ kubernetes-xenial main"
```

```
root@colom-master:/var/lib/docker# apt-add-repository "deb http://apt.kubernetes.io/ kubernetes-xenial main"
Hit:1 http://ports.ubuntu.com/ubuntu-ports focal InRelease
Hit:2 http://ports.ubuntu.com/ubuntu-ports focal-updates InRelease
Hit:3 http://ports.ubuntu.com/ubuntu-ports focal-backports InRelease
Hit:4 http://ports.ubuntu.com/ubuntu-ports focal-security InRelease
Get:5 https://packages.cloud.google.com/apt kubernetes-xenial InRelease [8993 B]
Get:6 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 Packages [39.2 kB]
Fetched 48.2 kB in 2s (24.7 kB/s)
Reading package lists... Done
root@colom-master:/var/lib/docker#
```

If you see the below error then the steps above to add the key where not successful.

```
Err:5 https://packages.cloud.google.com/apt kubernetes-xenial InRelease
  The following signatures couldn't be verified because the public key is not available: NO_PUBKEY 6A030B21BA07F4FB
Reading package lists... Done
W: GPG error: https://packages.cloud.google.com/apt kubernetes-xenial InRelease: The following signatures couldn't be verified because the public key is not available: NO_PUBKEY 6A030B21BA07F4FB
E: The repository 'http://apt.kubernetes.io kubernetes-xenial InRelease' is not signed.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
root@colom-master:/var/lib/docker#
```

Swap cannot be activated

```
swapoff -a
```

```
root@colom-master:/var/lib/docker# swapoff -a
```

Standard install didn't activate any swap file. This can be validated in /etc/fstab

Install the packages required for Kubernetes

```
apt install kubeadm kubelet kubectl kubernetes-cni
```

```
root@colom-master:/var/lib/docker# apt install kubeadm kubelet kubectl kubernetes-cni
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  conntrack cri-tools ebtables socat
Suggested packages:
  nftables
The following NEW packages will be installed:
  conntrack cri-tools ebtables kubeadm kubectl kubelet kubernetes-cni socat
0 upgraded, 8 newly installed, 0 to remove and 0 not upgraded.
Need to get 61.1 MB of archives.
After this operation, 276 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 conntrack arm64 1:1.4.5-2 [28.8 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 ebtables arm64 2.0.11-3build1 [77.7 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 socat arm64 1.7.3.3-2 [315 kB]
Get:4 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 cri-tools arm64 1.13.0-01 [7964 kB]
Get:5 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 kubernetes-cni arm64 0.8.7-00 [23.1 MB]
Get:6 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 kubelet arm64 1.19.2-00 [15.9 MB]
Get:7 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 kubectl arm64 1.19.2-00 [7072 kB]
Get:8 https://packages.cloud.google.com/apt kubernetes-xenial/main arm64 kubeadm arm64 1.19.2-00 [6601 kB]
Fetched 61.1 MB in 8s (7730 kB/s)
Selecting previously unselected package conntrack.
(Reading database ... 99751 files and directories currently installed.)
Preparing to unpack .../0-conntrack_1%3a1.4.5-2_arm64.deb ...
Unpacking conntrack (1:1.4.5-2) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.13.0-01_arm64.deb ...
Unpacking cri-tools (1.13.0-01) ...
Selecting previously unselected package ebtables.
Preparing to unpack .../2-ebtables_2.0.11-3build1_arm64.deb ...
Unpacking ebtables (2.0.11-3build1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../3-kubernetes-cni_0.8.7-00_arm64.deb ...
Unpacking kubernetes-cni (0.8.7-00) ...
Selecting previously unselected package socat.
Preparing to unpack .../4-socat_1.7.3.3-2_arm64.deb ...
Unpacking socat (1.7.3.3-2) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.19.2-00_arm64.deb ...
Unpacking kubelet (1.19.2-00) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../6-kubectl_1.19.2-00_arm64.deb ...
Unpacking kubectl (1.19.2-00) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../7-kubeadm_1.19.2-00_arm64.deb ...
Unpacking kubeadm (1.19.2-00) ...
Setting up conntrack (1:1.4.5-2) ...
Setting up kubectl (1.19.2-00) ...
Setting up ebtables (2.0.11-3build1) ...
Setting up socat (1.7.3.3-2) ...
Setting up cri-tools (1.13.0-01) ...
Setting up kubernetes-cni (0.8.7-00) ...
```

```
Setting up kubelet (1.19.2-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubelet.service.
Setting up kubeadm (1.19.2-00) ...
Processing triggers for man-db (2.9.1-1) ...
root@colom-master:/var/lib/docker#
```

Specific system settings:

```
vi /boot/firmware/cmdline.txt
```

Old settings:

```
net.ifnames=0 dwc_otg.lpm_enable=0 console=serial0,115200 console=tty1 root=LABEL=writable rootfstype=ext4 elevator=deadline rootwait fixrtc
```

New Settings:

```
net.ifnames=0 dwc_otg.lpm_enable=0 console=serial0,115200 console=tty1 root=LABEL=writable rootfstype=ext4 elevator=deadline rootwait fixrtc cgroup_enable=cpuset cgroup_enable=memory cgroup_memory=1
```

So the following was added at the end of the line:

cgroup_enable=cpuset cgroup_enable=memory cgroup_memory=1

reboot the system

Initialize Kubernetes → only on the master

```
kubeadm init
```

```
ubuntu@colom-master:~$ kubeadm init
W0921 20:59:46.336019    2563 kubelet.go:200] cannot automatically set CgroupDriver when starting the Kubelet: cannot execute 'docker info -f {{.CgroupDriver}}': exit status 2
W0921 20:59:47.114310    2563 configset.go:348] WARNING: kubeadm cannot validate component configs for API groups [kubelet.config.k8s.io kubeproxy.config.k8s.io]
[init] Using Kubernetes version: v1.19.2
[preflight] Running pre-flight checks
error execution phase preflight: [preflight] Some fatal errors occurred:
        [ERROR IsPrivilegedUser]: user is not running as root
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-preflight-errors=...`
To see the stack trace of this error execute with --v=5 or higher
ubuntu@colom-master:~$ sudo su -
root@colom-master:~# kubeadm init
W0921 20:59:58.775402    2631 configset.go:348] WARNING: kubeadm cannot validate component configs for API groups [kubelet.config.k8s.io kubeproxy.config.k8s.io]
[init] Using Kubernetes version: v1.19.2
[preflight] Running pre-flight checks
        [WARNING IsDockerSystemdCheck]: detected "cgroupfs" as the Docker cgroup driver. The recommended driver is "systemd". Please follow the guide at https://kubernetes.io/docs/setup/cri/
        [WARNING SystemVerification]: missing optional cgroups: hugetlb
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action in beforehand using 'kubeadm config images pull'
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [colom-master kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.local] and IPs [10.96.0.1 192.168.0.140]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [colom-master localhost] and IPs [192.168.0.140 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [colom-master localhost] and IPs [192.168.0.140 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
[kubeconfig] Writing "controller-manager.conf" kubeconfig file
```

```
[kubeconfig] Writing "scheduler.conf" kubeconfig file
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Starting the kubelet
[control-plane] Using manifest folder "/etc/kubernetes/manifests"
[control-plane] Creating static Pod manifest for "kube-apiserver"
[control-plane] Creating static Pod manifest for "kube-controller-manager"
[control-plane] Creating static Pod manifest for "kube-scheduler"
[etcd] Creating static Pod manifest for local etcd in "/etc/kubernetes/manifests"
[wait-control-plane] Waiting for the kubelet to boot up the control plane as static Pods from directory "/etc/kubernetes/manifests". This can take up to 4m0s
[kubelet-check] Initial timeout of 40s passed.
[apiclient] All control plane components are healthy after 41.010249 seconds
[upload-config] Storing the configuration used in ConfigMap "kubeadm-config" in the "kube-system" Namespace
[kubelet] Creating a ConfigMap "kubelet-config-1.19" in namespace kube-system with the configuration for the kubelets in the cluster
[upload-certs] Skipping phase. Please see --upload-certs
[mark-control-plane] Marking the node colom-master as control-plane by adding the label "node-role.kubernetes.io/master=''"
[mark-control-plane] Marking the node colom-master as control-plane by adding the taints [node-role.kubernetes.io/master:NoSchedule]
[bootstrap-token] Using token: 2xh64p.eo66cpxf1e5qikqv
[bootstrap-token] Configuring bootstrap tokens, cluster-info ConfigMap, RBAC Roles
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to get nodes
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to post CSRs in order for nodes to get long term certificate credentials
[bootstrap-token] configured RBAC rules to allow the csrapprover controller automatically approve CSRs from a Node Bootstrap Token
[bootstrap-token] configured RBAC rules to allow certificate rotation for all node client certificates in the cluster
[bootstrap-token] Creating the "cluster-info" ConfigMap in the "kube-public" namespace
[kubelet-finalize] Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 192.168.0.140:6443 --token 2xh64p.eo66cpxf1e5qikqv \
    --discovery-token-ca-cert-hash sha256:f8d287a1a9555ada1e3b6a83ec30cedb5927d3958d9fe50fcb92b873019d7841
root@colom-master:~#
```

kubeadm-join -> for worker nodes

```
kubeadm join 192.168.0.140:6443 --token 2xh64p.eo66cpxf1e5qikqv \
    --discovery-token-ca-cert-hash sha256:f8d287a1a9555ada1e3b6a83ec30cedb5927d3958d9fe50fcb92b873019d7841
```

Tokens do expire – so if you need to generate a new one on the master

```
kubeadm token create
kubeadm token list
```

modify the token, but leave the cert as is

Allow user to use Kubernetes commands:

Run the following commands for root (if really required) and e.g. ubuntu user

```
mkdir -p $HOME/.kube

sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Check if running:

```
kubectl cluster-info
```

```
root@colom-master:~# kubectl cluster-info
Kubernetes master is running at https://192.168.0.140:6443
```

Kubernetes Container Network Interface

We must deploy a Container Network Interface (CNI) based Pod network (calico, flannel,canal or weave-net).

Reading tip : https://rancher.com/blog/2019/2019-03-21-comparing-kubernetes-cni-providers-flannel-calico-canal-and-weave/

```
kubectl apply -f https// docs.projectcalico.org/v3.14/manifests/calico.yam
```

```
root@colom-master:~# kubectl apply -f https://docs.projectcalico.org/v3.14/manifests/calico.yaml
configmap/calico-config created
Warning: apiextensions.k8s.io/v1beta1 CustomResourceDefinition is deprecated in v1.16+, unavailable in v1.22+; use apiextensions.k8s.io/v1 CustomResourceDefinition
customresourcedefinition.apiextensions.k8s.io/bgpconfigurations.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/bgppeers.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/blockaffinities.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/clusterinformations.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/felixconfigurations.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/globalnetworkpolicies.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/globalnetworksets.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/hostendpoints.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/ipamblocks.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/ipamconfigs.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/ipamhandles.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/ippools.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/kubecontrollersconfigurations.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/networkpolicies.crd.projectcalico.org created
customresourcedefinition.apiextensions.k8s.io/networksets.crd.projectcalico.org created
clusterrole.rbac.authorization.k8s.io/calico-kube-controllers created
clusterrolebinding.rbac.authorization.k8s.io/calico-kube-controllers created
clusterrole.rbac.authorization.k8s.io/calico-node created
clusterrolebinding.rbac.authorization.k8s.io/calico-node created
daemonset.apps/calico-node created
serviceaccount/calico-node created
deployment.apps/calico-kube-controllers created
serviceaccount/calico-kube-controllers created
```

Check the nodes

Kubernetes Metrics Server

Metrics-Server useful link: https://github.com/kubernetes-sigs/metrics-server

```
kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/download/v0.3.7/components.yaml
```

```
root@colom-master:~#  kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/download/v0.3.7/components.yaml
clusterrole.rbac.authorization.k8s.io/system:aggregated-metrics-reader created
clusterrolebinding.rbac.authorization.k8s.io/metrics-server:system:auth-delegator created
rolebinding.rbac.authorization.k8s.io/metrics-server-auth-reader created
Warning: apiregistration.k8s.io/v1beta1 APIService is deprecated in v1.19+, unavailable in v1.22+; use apiregistration.k8s.io/v1 APIService
apiservice.apiregistration.k8s.io/v1beta1.metrics.k8s.io created
serviceaccount/metrics-server created
deployment.apps/metrics-server created
service/metrics-server created
clusterrole.rbac.authorization.k8s.io/system:metrics-server created
clusterrolebinding.rbac.authorization.k8s.io/system:metrics-server created
root@colom-master:~# kubectl get nodes all-
```

Solved: Master could not run pods (taint)

```
kube-system   metrics-server-68b849498d-6ksbc        0/1     Pending   0          17m
```

Check the service:

```
kubectl get apiservice v1beta1.metrics.k8s.io -o yaml
```

```
root@colom-master:~# kubectl get apiservice v1beta1.metrics.k8s.io -o yaml
apiVersion: apiregistration.k8s.io/v1
kind: APIService
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |

{"apiVersion":"apiregistration.k8s.io/v1beta1","kind":"APIService","metadata":{"annotations":{},"name":"v1beta1.metrics.k8s.io"},"spec":{"g
roup":"metrics.k8s.io","groupPriorityMinimum":100,"insecureSkipTLSVerify":true,"service":{"name":"metrics-server","namespace":"kube-
system"},"version":"v1beta1","versionPriority":100}}
  creationTimestamp: "2020-09-22T07:17:06Z"
  name: v1beta1.metrics.k8s.io
  resourceVersion: "81490"
  selfLink: /apis/apiregistration.k8s.io/v1/apiservices/v1beta1.metrics.k8s.io
  uid: 1ac3c233-d79b-4ce3-a166-e93af93d4d76
spec:
  group: metrics.k8s.io
  groupPriorityMinimum: 100
  insecureSkipTLSVerify: true
  service:
    name: metrics-server
    namespace: kube-system
    port: 443
  version: v1beta1
  versionPriority: 100
status:
  conditions:
  - lastTransitionTime: "2020-09-22T07:17:06Z"
    message: endpoints for service/metrics-server in "kube-system" have no addresses
      with port name ""
    reason: MissingEndpoints
    status: "False"
    type: Available
```

Delete the service (did not work as it was not created correctly)

```
root@colom-master:~# kubectl delete apiservice v1beta1.metrics.k8s.io
```

The pod metrics-server kept being in pending state. The root cause could be found by using the describe command:

```
root@colom-master:~# kubectl describe pod metrics-server-68b849498d-xnc7k -n kube-system
Name:         metrics-server-68b849498d-xnc7k
Namespace:    kube-system
Priority:     0
Node:         colom-master/192.168.0.140
Start Time:   Tue, 22 Sep 2020 11:08:47 +0000
Labels:       k8s-app=metrics-server
              pod-template-hash=68b849498d
Annotations:  cni.projectcalico.org/podIP: 172.16.63.74/32
              cni.projectcalico.org/podIPs: 172.16.63.74/32
Status:       Running
IP:           172.16.63.74
IPs:
  IP:           172.16.63.74
Controlled By:  ReplicaSet/metrics-server-68b849498d
Containers:
  metrics-server:
    Container ID:  docker://59f5c561d24700ae6aaca134e0ae7f483116010a6b14eacb7ae3e5b5f4a15cce
    Image:         k8s.gcr.io/metrics-server/metrics-server:v0.3.7
    Image ID:      docker-pullable://k8s.gcr.io/metrics-server/metrics-
server@sha256:eec279de92328954ec69e9c2ef920861de28d31bb14b5290b53b5ef3dfa96502
    Port:          4443/TCP
    Host Port:     0/TCP
    Args:
      --cert-dir=/tmp
      --secure-port=4443
```

```
    State:          Running
      Started:      Tue, 22 Sep 2020 11:08:50 +0000
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /tmp from tmp-dir (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from metrics-server-token-tlfqf (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  tmp-dir:
    Type:       EmptyDir (a temporary directory that shares a pod's lifetime)
    Medium:
    SizeLimit:  <unset>
  metrics-server-token-tlfqf:
    Type:       Secret (a volume populated by a Secret)
    SecretName:  metrics-server-token-tlfqf
    Optional:    false
QoS Class:       BestEffort
Node-Selectors:  kubernetes.io/os=linux
Tolerations:     node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason            Age              From               Message
  ----     ------            ----             ----               -------
  Warning  FailedScheduling  90s (x12 over 16m)  default-scheduler  0/1 nodes are available: 1 node(s) had taint {node-
role.kubernetes.io/master: }, that the pod didn't tolerate.
  Normal   Scheduled         9s               default-scheduler  Successfully assigned kube-system/metrics-server-68b849498d-xnc7k to colom-
master
  Normal   Pulled            7s               kubelet            Container image "k8s.gcr.io/metrics-server/metrics-server:v0.3.7" already present on
machine
  Normal   Created           6s               kubelet            Created container metrics-server
  Normal   Started           6s               kubelet            Started container metrics-server
```

Kubernetes Dashboard

Install

Dashboard install: https://computingforgeeks.com/how-to-install-kubernetes-dashboard-with-nodeport/

```
        kubectl apply -f
https://raw.githubusercontent.com/kubernetes/dashboard/master/aio/deploy/recommended.yaml
```

```
ubuntu@colom-master:~$ kubectl apply -f
https://raw.githubusercontent.com/kubernetes/dashboard/master/aio/deploy/recommended.yaml
namespace/kubernetes-dashboard created
serviceaccount/kubernetes-dashboard created
service/kubernetes-dashboard created
secret/kubernetes-dashboard-certs created
secret/kubernetes-dashboard-csrf created
secret/kubernetes-dashboard-key-holder created
configmap/kubernetes-dashboard-settings created
role.rbac.authorization.k8s.io/kubernetes-dashboard created
```

```
clusterrole.rbac.authorization.k8s.io/kubernetes-dashboard created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
clusterrolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
deployment.apps/kubernetes-dashboard created
service/dashboard-metrics-scraper created
deployment.apps/dashboard-metrics-scraper created
ubuntu@colom-master:~$
```

Configure

Changing to NodePort to open up outside the cluster

If we check the service we see the port is only opened internally

```
ubuntu@colom-master:~$ kubectl get services -n kubernetes-dashboard
NAME                        TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
dashboard-metrics-scraper   ClusterIP   10.98.108.182   <none>        8000/TCP   21m
kubernetes-dashboard        ClusterIP   10.111.187.95   <none>        443/TCP    21m
```

Edit the service to change the type to NodePort

```
kubectl edit service kubernetes-dashboard -n kubernetes-dashboard
```

```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: v1
kind: Service
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","kind":"Service","metadata":{"annotations":{},"labels":{"k8s-app":"kubernetes-dashboard"},"name":"kubernetes-dashboard","namespace":"kubernetes-dashboard"},"spec":{"ports":[{"port":443,"targetPort":8443}],"selector":{"k8s-app":"kubernetes-dashboard"}}}
  creationTimestamp: "2020-09-24T17:18:15Z"
  labels:
    k8s-app: kubernetes-dashboard
  name: kubernetes-dashboard
  namespace: kubernetes-dashboard
  resourceVersion: "540506"
  selfLink: /api/v1/namespaces/kubernetes-dashboard/services/kubernetes-dashboard
  uid: 2067c99a-4177-4ac6-bba1-d88567b8fc3a
spec:
  clusterIP: 10.111.187.95
  externalTrafficPolicy: Cluster
  ports:
  - nodePort: 30115
    port: 443
    protocol: TCP
    targetPort: 8443
  selector:
    k8s-app: kubernetes-dashboard
  sessionAffinity: None
  type: NodePort
status:
    loadBalancer: {}
```

The dashboard should now be reachable from the webbrowser of any computer on the network

Remote link : https://192.168.0.140:30115/#/login

Service Account

```
ubuntu@colom-master:~$ vi admin-sa.yml
ubuntu@colom-master:~$ kubectl apply -f admin-sa.yml
serviceaccount/dashboard-admin created
ubuntu@colom-master:~$ vi admin-rbac.yml
ubuntu@colom-master:~$ kubectl apply -f admin-rbac.yml
clusterrolebinding.rbac.authorization.k8s.io/dashboard-admin created
ubuntu@colom-master:~$
```

NFS Server

Install

Install the required package (currently I'll install on the master, maybe best to install on separate system together with a desktop?)

```
apt install nfs-common nfs-kernel-server
```

```
root@colom-master:~# apt install nfs-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-5.4.0-1015-raspi linux-image-5.4.0-1015-raspi linux-modules-5.4.0-1015-raspi linux-raspi-headers-5.4.0-1015
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  keyutils libnfsidmap2 libtirpc-common libtirpc3 rpcbind
Suggested packages:
  watchdog
The following NEW packages will be installed:
  keyutils libnfsidmap2 libtirpc-common libtirpc3 nfs-common rpcbind
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 391 kB of archives.
After this operation, 1411 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 libtirpc-common all 1.2.5-1 [7632 B]
Get:2 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 libtirpc3 arm64 1.2.5-1 [74.1 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 rpcbind arm64 1.2.5-8 [41.1 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 keyutils arm64 1.6-6ubuntu1 [43.6 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 libnfsidmap2 arm64 0.25-5.1ubuntu1 [27.1 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports focal-updates/main arm64 nfs-common arm64 1:1.3.4-2.5ubuntu3.3 [197 kB]
Fetched 391 kB in 0s (1845 kB/s)
Selecting previously unselected package libtirpc-common.
(Reading database ... 132446 files and directories currently installed.)
Preparing to unpack .../0-libtirpc-common_1.2.5-1_all.deb ...
Unpacking libtirpc-common (1.2.5-1) ...
Selecting previously unselected package libtirpc3:arm64.
Preparing to unpack .../1-libtirpc3_1.2.5-1_arm64.deb ...
Unpacking libtirpc3:arm64 (1.2.5-1) ...
Selecting previously unselected package rpcbind.
Preparing to unpack .../2-rpcbind_1.2.5-8_arm64.deb ...
Unpacking rpcbind (1.2.5-8) ...
Selecting previously unselected package keyutils.
Preparing to unpack .../3-keyutils_1.6-6ubuntu1_arm64.deb ...
Unpacking keyutils (1.6-6ubuntu1) ...
Selecting previously unselected package libnfsidmap2:arm64.
Preparing to unpack .../4-libnfsidmap2_0.25-5.1ubuntu1_arm64.deb ...
Unpacking libnfsidmap2:arm64 (0.25-5.1ubuntu1) ...
```

```
Selecting previously unselected package nfs-common.
Preparing to unpack .../5-nfs-common_1%3a1.3.4-2.5ubuntu3.3_arm64.deb ...
Unpacking nfs-common (1:1.3.4-2.5ubuntu3.3) ...
Setting up libtirpc-common (1.2.5-1) ...
Setting up keyutils (1.6-6ubuntu1) ...
Setting up libnfsidmap2:arm64 (0.25-5.1ubuntu1) ...
Setting up libtirpc3:arm64 (1.2.5-1) ...
Setting up rpcbind (1.2.5-8) ...
Created symlink /etc/systemd/system/multi-user.target.wants/rpcbind.service → /lib/systemd/system/rpcbind.service.
Created symlink /etc/systemd/system/sockets.target.wants/rpcbind.socket → /lib/systemd/system/rpcbind.socket.
Setting up nfs-common (1:1.3.4-2.5ubuntu3.3) ...

Creating config file /etc/idmapd.conf with new version
Adding system user `statd' (UID 114) ...
Adding new user `statd' (UID 114) with group `nogroup' ...
Not creating home directory `/var/lib/nfs'.
Created symlink /etc/systemd/system/multi-user.target.wants/nfs-client.target → /lib/systemd/system/nfs-client.target.
Created symlink /etc/systemd/system/remote-fs.target.wants/nfs-client.target → /lib/systemd/system/nfs-client.target.
nfs-utils.service is a disabled or a static unit, not starting it.
Processing triggers for systemd (245.4-4ubuntu3.2) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.1) ...
root@colom-master:~#
```

USB-Drive

```
fdisk -l
```

fdisk command shows that the usb drive is on /dev/sda (in red below) and linux filesystem on /dev/sda1 (blue below)

```
root@colom-master:~# fdisk -l
Disk /dev/loop0: 48.46 MiB, 50798592 bytes, 99216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop1: 63.6 MiB, 66674688 bytes, 130224 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop2: 48.8 MiB, 51154944 bytes, 99912 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop3: 62.96 MiB, 65998848 bytes, 128904 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop4: 26.26 MiB, 27533312 bytes, 53776 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop5: 25.96 MiB, 27201536 bytes, 53128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mmcblk0: 29.74 GiB, 31914983424 bytes, 62333952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xab86aefd

Device       Boot  Start     End  Sectors  Size Id Type
/dev/mmcblk0p1 *     2048   526335   524288  256M  c W95 FAT32 (LBA)
/dev/mmcblk0p2     526336 62333918 61807583 29.5G 83 Linux


Disk /dev/sda: 476.96 GiB, 512110190592 bytes, 1000215216 sectors
Disk model: D370S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: C8D96B29-E04A-4838-9144-490B650B9D7F

Device     Start       End   Sectors  Size Type
/dev/sda1   2048 1000215182 1000213135  477G Linux filesystem
root@colom-master:~#
```

Create a mount point where the drive will be mounted to. For the domotica farm domodata

```
mkdir /domodata
```

Modify fstab to mount on boot

```
vi /etc/fstab
```

```
LABEL=writable /      ext4   defaults      0 0
LABEL=system-boot     /boot/firmware vfat   defaults      0     1
/dev/sda1     /domodata ext4  defaults      0     0
```

/dev/sda1 line is added

```
mount -a
```

nfs-export

Good read on share options: https://www.golinuxcloud.com/unix-linux-nfs-mount-options-example/

```
vi /etc/exports
```

exportfs -r
exportfs -v

NFS Provisioner

https://opensource.com/article/20/6/kubernetes-nfs-client-provisioning

Useful system packages

Locate

```
root@colom-master:~# apt install mlocate
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  nocache
The following NEW packages will be installed:
  mlocate
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 48.1 kB of archives.
After this operation, 250 kB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports focal/main arm64 mlocate arm64 0.26-3ubuntu3 [48.1 kB]
Fetched 48.1 kB in 0s (241 kB/s)
Selecting previously unselected package mlocate.
(Reading database ... 99844 files and directories currently installed.)
Preparing to unpack .../mlocate_0.26-3ubuntu3_arm64.deb ...
Unpacking mlocate (0.26-3ubuntu3) ...
Setting up mlocate (0.26-3ubuntu3) ...
update-alternatives: using /usr/bin/mlocate to provide /usr/bin/locate (locate) in auto mode
Adding group `mlocate' (GID 120) ...
Done.
Initializing mlocate database; this may take some time... done
Processing triggers for man-db (2.9.1-1) ...
root@colom-master:~#
```

Kubernetes Commands

Cheatsheet

Cheatsheet: https://kubernetes.io/docs/reference/kubectl/cheatsheet/

Configuration

Reset Kubernetes

```
sudo kubeadm reset
sudo rm -rf /etc/cni/net.d
```

Taint mode

```
kubectl taint node colom-master node-role.kubernetes.io/master:NoSchedule-
```

To make the node dedicated again

```
kubectl taint node colom-master dedicated-
```

Monitoring

Get cluster info

```
kubectl cluster-info
```

Check the cluster nodes

```
kubectl get nodes
```

Openhab

Needs host network: hostNetwork: true

```
kind: Deployment
apiVersion: apps/v1
metadata:
  name: openhab
  labels:
    k8s-app: openhab
  annotations:
    description: openhab
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: openhab
  template:
    metadata:
      name: openhab
      labels:
        k8s-app: openhab
      annotations:
        description: openhab
    spec:
      volumes:
        - name: openhab-userdata
```

```
        nfs:
          server: 192.168.0.140
          path: /domodata/openhab/userdata
      - name: openhab-conf
        nfs:
          server: 192.168.0.140
          path: /domodata/openhab/conf
      - name: openhab-addons
        nfs:
          server: 192.168.0.140
          path: /domodata/openhab/addons
      - name: localtime
        hostPath:
          path: /etc/localtime
      - name: timezone
        hostPath:
          path: /etc/timezone
    hostNetwork: true
    containers:
      - name: openhab
        image: 'openhab/openhab:2.5.9'
        ports:
          - containerPort: 8080
            hostPort: 8080
        volumeMounts:
          - mountPath: "/openhab/userdata"
            name: openhab-userdata
          - mountPath: "/openhab/conf"
            name: openhab-conf
          - mountPath: "/openhab/addons"
            name: openhab-addons
          - mountPath: "/etc/localtime:ro"
            name: localtime
          - mountPath: "/etc/timezone:ro"
            name: timezone
        env:
          - name: OPENHAB_HTTP_PORT
            value: "8080"
          - name: OPENHAB_HTTPS_PORT
            value: "8443"
          - name: CRYPTO_POLICY
            value: "unlimited"
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
        imagePullPolicy: IfNotPresent
        securityContext:
          privileged: false
    nodeSelector:
      kubernetes.io/hostname: colom-worker
    restartPolicy: Always
    terminationGracePeriodSeconds: 30
    dnsPolicy: ClusterFirstWithHostNet
    securityContext: {}
    schedulerName: default-scheduler
  strategy:
    type: Recreate
  revisionHistoryLimit: 10
  progressDeadlineSeconds: 600
```

Nodered (rules based on flows)

Influxdb

Commands in the pod

Influx
show series on openhab

Curl commands (can be used to test connectivity from within another pod)

curl -G "http://influxdb:8086/query?pretty=true" --data-urlencode "q=show series on openhab"

Grafana

Usefull link: https://grafana.com/docs/grafana/latest/installation/docker/#image-variants

Omada (tp-link wifi control)