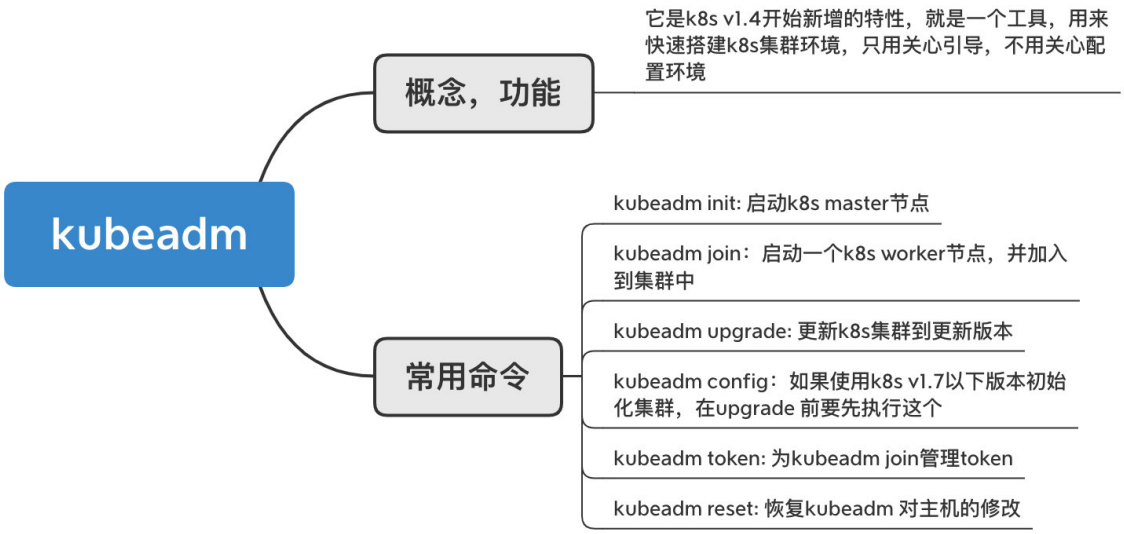
1. kubeadm knowledge map



Kubeadm is a tool to implement the k8s environment quickly. Also you don't need to care about the configure environment, just know how to bootstrap it. Master the basically command to use kubeadm.

2. Kubeadm practice

2.1 The practice environment

Google cloud instance.
Linux system.

2.2 Environment configure

- Install container (Docker)

```
<!--
# Install containerd
## Set up the repository
### Install packages to allow apt to use a repository over HTTPS
apt-get update && apt-get install -y apt-transport-https ca-certificates curl software-properties-
common

### Add Docker’s official GPG key
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-key add -

### Add Docker apt repository.
add-apt-repository \
    "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) \
    stable"

## Install containerd
apt-get update && apt-get install -y containerd.io

# Configure containerd
mkdir -p /etc/containerd
```

```
containerd config default > /etc/containerd/config.toml
-->
# 安装 containerd
## 设置仓库
### 安装软件包以允许 apt 通过 HTTPS 使用存储库
apt-get update && apt-get install -y apt-transport-https ca-certificates curl software-properties-
common

### 安装 Docker 的官方 GPG 密钥
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-key add -

### 新增 Docker apt 仓库。
add-apt-repository \
    "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) \
    stable"

## 安装 containerd
apt-get update && apt-get install -y containerd.io

# 配置 containerd
mkdir -p /etc/containerd
containerd config default > /etc/containerd/config.toml

<!--
# Restart containerd
systemctl restart containerd
-->
# 重启 containerd
systemctl restart containerd
```

- Install kubeadm, kubectl and kubelet

```
apt-get update && apt-get install -y apt-transport-https curl
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add -
cat <<EOF >/etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
apt-get update
apt-get install -y kubelet kubeadm kubectl
apt-mark hold kubelet kubeadm kubectl
```

2.3 kubeadm practice

- Initialize the master node.

```
kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=all
```

pod-network-cidr, means identify the pod ip range, also we use the flannel network design solution.

Ignore, means ignore the error when startup. Because when start up kubeadm, perhaps hit the hardware uncomfortable.

- After start up successful, will generate the kubeadm token, this token can be used to join other nodes. You can use below command to view the token.

```
kubeadm token list
```

- Configure the kubectl.

As we know, kubectl is the command tool to control kubernetes cluster. When we switch to the master node, we need to configure the kubectl.

```
mkdir -p $HOME/.kube
cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
chown $(id -u):$(id -g) $HOME/.kube/config
echo export KUBECONFIG=~/.kube/config>> ~/.bashrc
source ~/.bashrc
```

- Install the pod network add-on. The pods can communicate each other after install the pod network. Also we use the flannel network mode.

```
kubectl apply -f
https://raw.githubusercontent.com/coreos/flannel/2140ac876ef134e0ed5af15c65e414cf26827915/Documenta
tion/kube-flannel.yml
```

- Join other nodes to the cluster.

```
sudo kubeadm join 10.128.0.2:6443 --token 5dhzcw.h7aih16mg982ms2o --discovery-token-ca-cert-hash
sha256:e9e6843a6ae6fc5fb8acb9f116bc58d1c1e0f30d1da9bfe3bf151319c3788d57 --ignore-preflight-
errors=all
```

- Clean up the environment

  After deploy, you can clean up the environment.

```
sudo kubeadm reset
```

3. Additional

Actually there are many issues when you follow the steps.

Unsolved problems:

- After execute the kubeadm join command, the terminal show it was added successful, but in fact, the new node isn't existing in the node list.
