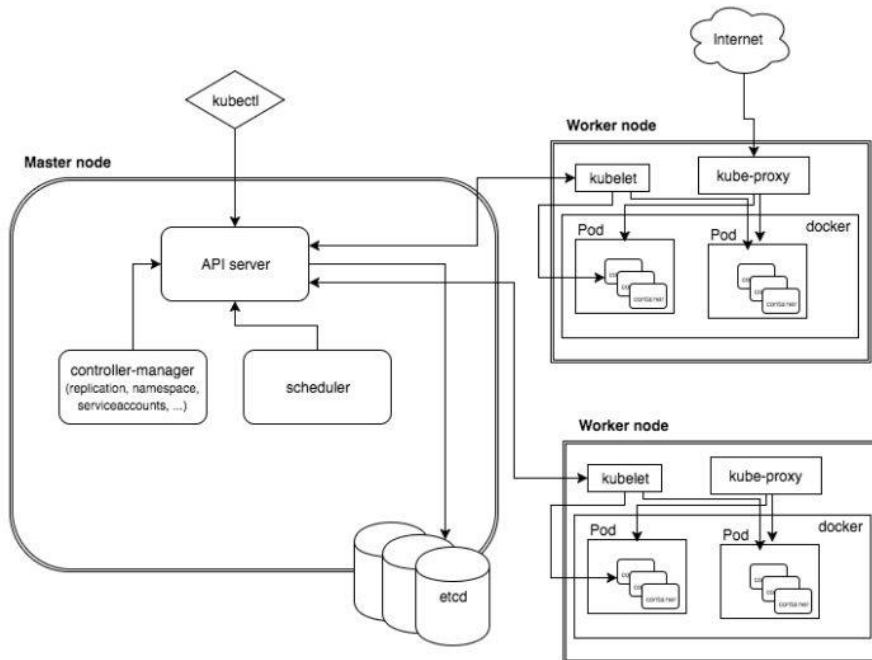**Worker Node Components :**

Kubelet :

The Kubelet is one of the most important components in Kubernetes. Basically, it's an agent that runs on each node and is responsible for watching the API Server for pods that are bound to its node and making sure those pods are running (it talks to the Docker daemon using the API over the Docker socket to manipulate containers lifecycle). It then reports back to the API Server the status of changes regarding those pods.

The main Kubelet responsibilities include:

- Run the pods containers.
- Report the status of the node and each pod to the API Server.
- Run container probes.
- Retrieve container metrics from cAdvisor, aggregate and expose them through the Kubelet Summary API for components (such as Heapster) to consume.

Kube Proxy :

The Kube Proxy runs on each node and is responsible for watching the API Server for changes on services and pods definitions to maintain the entire network configuration up to date, ensuring that one pod can talk to another pod, one node can talk to another node, one container can talk to another container, and so on. Besides, it

exposes Kubernetes services and manipulates iptables rules to trap access to services IPs and redirect them to the correct backends (that's why you can access a NodePort service using any node IP; even if the node you hit is not the one you are looking for, this node will already be set up with the appropriate iptables rules to redirect your request to the correct backend). This provides a highly-available load-balancing solution with low performance overhead.

What is Minikube?

Minikube is a utility you can use to run Kubernetes (k8s) on your local machine. It creates a single node cluster contained in a virtual machine (VM). This cluster lets you demo Kubernetes operations without requiring the time and resource-consuming installation of full-blown K8s.

All you need is Docker (or similarly compatible) container or a Virtual Machine environment, and Kubernetes is a single command away: minikube start.

Recommended Min Hardware Config Required :
2 CPUs or more
2GB of free memory
20GB of free disk space
Internet connection
Container or virtual machine manager, such as: **Docker, Hyperkit, Hyper-V, KVM, Parallels, Podman, VirtualBox, or VMWare**

Kubectl :

The Kubernetes command-line tool, kubectl, allows you to run commands against Kubernetes clusters. You can use kubectl to deploy applications, inspect and manage cluster resources, and view logs. For more information including a complete list of kubectl operations, see the kubectl reference documentation.

kubectl is installable on a variety of Linux platforms, macOS and Windows.

Steps to Config Kubernetes Cluster using MINIKUBE :

1)Install Virtual Box

2)Install KUBECTL

3)Install MINIKUBE

4)Start MINIKUBE as Non Root User

5) Ensure these all we are doing on Base Operating system **(Note on Guest OS)**

Installing Virtual Box on Base OS :

#yum install wget vim –y **(Installing wget & Vim softwares)**
#wget https://download.virtualbox.org/virtualbox/rpm/rhel/virtualbox.repo -P /etc/yum.repos.d/ **(Configuring Virtual Box Repository)**

#yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm **(Configuring EPEL Repository)**

#yum update

#yum install binutils kernel-devel kernel-headers libgomp make patch gcc glibc-headers glibc-devel dkms -y

#Yum install VirtualBox-6.1 **(Installing Virtual Box)**
#which virtualbox
#reboot
#/sbin/vboxconfig **(Stopping & STarting the Virtual Box Services)**
#systemctl status vboxconfig **(Verify Virtual box Service is Runnin)**

Installing KUBECTL :

#curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl

#chmod +x ./kubectl

#mv ./kubectl /usr/local/bin/kubectl **(Move the binary in to your PATH)**

Best to ensure the version you installed is up-to-date:
#kubectl version --client

Installing MINIKUBE :

#curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64 && chmod +x minikube

#mkdir -p /usr/local/bin/

#Install minikube /usr/local/bin/

$minikube version

$minikube status

Confirm Installation : (From here you need to do with non root user)

$minikube start --driver=<driver_name>

                virtualbox | vmware (As per your Hypervisor)

NOTE :

1)By default Minikube used 2GB of Ram, 2 V CPU & 20GB of Disk space by default. It is possible to create MiniKube VM with our custom required Hardware configurations

$minikube --memory 8192 --cpus 6 start

After Installing Test it :

$minikube status **(Displays the status of MiniKube)**

$kubectl config view **(Displays Kubernetes Cluster configuration Information)**

$kubectl get no **(Displays the nodes available in the Cluster along with its state)**

$kubectl get po **(Displaying the pods Created)**

$kubectl delete po <POD NAME> **(Deleting the Specific POD)**

$minikube stop **(Stopping the Minikube VM)**

$minikube delete **(Deleting the MiniKube VM)**

$minikube status **(Verify the VM is Deleted or Not)**