**INSTALLATION AND USING KIND**

In the root directory create a file named install\_kind.sh

***vim install\_kind.sh***

#!/bin/bash

[ $(uname -m) = x86\_64 ] && curl -Lo ./kind https://kind.sigs.k8s.io/dl/v0.29.0/kind-linux-amd64

chmod +x ./kind

sudo mv ./kind /usr/local/bin/kind

VERSION="v1.30.0"

URL="https://dl.k8s.io/release/${VERSION}/bin/linux/amd64/kubectl"

INSTALL\_DIR="/usr/local/bin"

curl -LO "$URL"

chmod +x kubectl

sudo mv kubectl $INSTALL\_DIR/

kubectl version --client

rm -f kubectl

rm -rf kind

echo "kind & kubectl installation complete."

***chmod 777 install\_kind.sh***

***./install\_kind.sh***

***mkdir kind-cluster***

***sudo systemctl enable –now docker***

***vim config.yml***

kind: Cluster

apiVersion: kind.x-k8s.io/v1alpha4

nodes:

- role: control-plane

image: kindest/node:v1.31.2

- role: worker

image: kindest/node:v1.31.2

- role: worker

image: kindest/node:v1.31.2

- role: worker

image: kindest/node:v1.31.2

***kubectl get nodes***

If both Kubernetes contexts are installed for example “kind” and “minikube” then do not give “extraportmapping” in the config.yml file of kind

To check all the contexts installed use kubectl

***kubectl config get-contexts***

To select a particular context use

***kubectl config use-context “context\_name”***

***kubectl get nodes***