**Prometheus, kind, kubectl, Argo CD, Helm installation and configuration**

**1. Creating and Managing Kubernetes Cluster with Kind**

clear # Clear the terminal screen

kind create cluster --config=config.yml # Create a 3-node Kubernetes cluster

kubectl cluster-info --context kind-kind # Show cluster information

kubectl get nodes # List all nodes in the cluster

kind get clusters # Display Kind clusters

**2. Installing kubectl (Linux)**

curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl

chmod +x ./kubectl

sudo mv ./kubectl /usr/local/bin

kubectl version --short --client # Verify kubectl installation

**3. Managing Docker and Kubernetes Pods**

docker ps # List running Docker containers

kubectl get pods -A # List all Kubernetes pods across namespaces

**4. Installing Argo CD**

kubectl create namespace argocd # Create Argo CD namespace

kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

kubectl get svc -n argocd # Check Argo CD services

kubectl patch svc argocd-server -n argocd -p '{"spec": {"type": "NodePort"}}'

kubectl port-forward -n argocd service/argocd-server 8443:443 &

**5. Deleting Kubernetes Cluster**

kind delete cluster --name=kind # Delete Kind cluster

**6. Installing Kubernetes Dashboard**

kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml

kubectl -n kubernetes-dashboard create token admin-user

**7. Argo CD Initial Admin Password**

kubectl get secret -n argocd argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d && echo

**8. Install HELM**

curl -fsSL -o get\_helm.sh https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3

chmod 700 get\_helm.sh

./get\_helm.sh

**9. Install Kube Prometheus Stack**

helm repo add prometheus-community https://prometheus-community.github.io/helm-charts

helm repo add stable https://charts.helm.sh/stable

helm repo update

kubectl create namespace monitoring

helm install kind-prometheus prometheus-community/kube-prometheus-stack --namespace monitoring \

--set prometheus.service.nodePort=30000 \

--set prometheus.service.type=NodePort \

--set grafana.service.nodePort=31000 \

--set grafana.service.type=NodePort \

--set alertmanager.service.nodePort=32000 \

--set alertmanager.service.type=NodePort \

--set prometheus-node-exporter.service.nodePort=32001 \

--set prometheus-node-exporter.service.type=NodePort

kubectl get svc -n monitoring

kubectl get namespace

**Port Forwarding**

kubectl port-forward svc/kind-prometheus-kube-prome-prometheus -n monitoring 9090:9090 --address=0.0.0.0 &

kubectl port-forward svc/kind-prometheus-grafana -n monitoring 31000:80 --address=0.0.0.0 &

**10. Prometheus Queries**

sum(rate(container\_cpu\_usage\_seconds\_total{namespace="default"}[1m])) / sum(machine\_cpu\_cores) \* 100

sum(container\_memory\_usage\_bytes{namespace="default"}) by (pod)

sum(rate(container\_network\_receive\_bytes\_total{namespace="default"}[5m])) by (pod)

sum(rate(container\_network\_transmit\_bytes\_total{namespace="default"}[5m])) by (pod)