upGrad
Building Careers of Tomorrow

Kubernetes Cheat Sheet

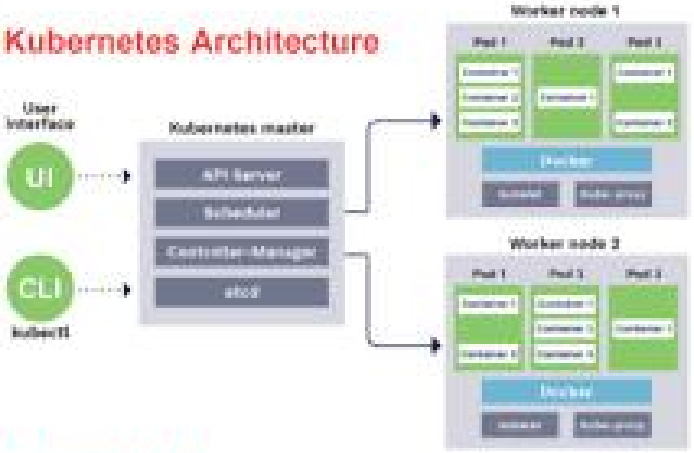
About Kubernetes

Kubernetes is a platform that is designed for managing the life cycle of containerized applications and services completely.

Features

- Maximize resources by making better use of hardware.
- A container orchestrator across multiple hosts.
- Automate the deployment process and updates.
- Able to run a Linux container.
- Auto scaling helps in launching containers on cluster nodes.
- Scaled up and down as per the need.
- Self-healing by replacing, rescheduling, and restarting the dead containers.
- Automated rollbacks and rollouts.
- Load balancing and service discovery.
- Auto restart, auto placement, and auto replication, etc.

Kubernetes Architecture



Components

- API server: Kubernetes API server
- Scheduler: Used for pod scheduling in worker nodes
- Controller: Manages pod replication
- Etcd: A metadata service
- Pod: Group of containers
- Docker: Container based technology, user space of OS
- Kubelet: Container agents that is responsible for maintaining the set of pods
- Kube-proxy: Routes traffic coming into a node from the service

Kubectl Commands

Pods and Container Introspection	
Kubectl describe pod<name>	For describing pod names
Kubectl get pods	For listing all current pods
Kubectl get rc	For listing all replication controllers
Kubectl get rc –namespace="namespace"	For listing replication controllers in a namespace
Kubectl describe rc <name>	For showing the replication controller name
Kubectl get cvc	For listing services
Kubectl describe svc<name>	For showing a service name
Kubectl delete pod<name>	For deleting a pod
Kubectl get nodes -w	For watching nodes continuously

Cluster Introspection	
Kubectl version	For getting version-related information
Kubectl cluster-info	For getting cluster-related information
Kubectl config g view	For getting configuration details
Kubectl describe node<node>	For getting information about a node

Debugging Commands	
Kubectl top pod	For displaying metrics for a pod
Kubectl top node	For displaying metrics for a node
Watch -n 2 cat/var/log/kublet.log	For watching kubelet logs
Kubectl logs -f<name>>[-c< $container>]	For getting logs from the service for the container
Kubectl exec<service><commands>[-c< $container>]	For execution of the command on service by selecting a container

Quick Commands	
Kubectl run<name> — image=<image-name>	For launching a pod with a name and an image
Kubectl create -f <manifest.yaml>	For creating a service described in <manifest.yaml>
Kubectl scale –replicas=<count>rc<name>	For scaling the replication counter to count the number of instances
Expose rc<name> –port=<external>–target-port=<internal>	For mapping the external port to the internal replication port
Kubectl drain<n>– delete-local-data–force–ignore-daemonset	For stopping all pods in <n>
Kubectl create namespace <namespace>	For creating a namespace
Kubectltaintnodes –all-node-role.kuernetes.io/master-	For allowing the master node to run pods

List of Common Objects
All
cm= conf gmaps
Cronjobs
Deploy=deployments
ev= events
jobs
No = nodes
po= pods
Psp= pod security policies
quota= resource quotas
roles
sc= storage classes
clusterroles
crd=custom resource definition
csr= certificate signing requests
ep=end points
ing= ingress
Netpol- network policies
clusterrolebindings
controllerrevisions
cs=component statuses
ds= daemon sets
hpa= horizontal pod autoscaling
limits=limit ranges
ns= namespaces
Pod preset
Pv= persistent volumes
rc= replication controllers
rs= replica sets
secrets
pdb= pod distribution budgets
Pod templates
pvc= persistent volume claims
Role bindings
sa= service accounts
sts= stateful sets

All the basic details of Kubernetes are covered in this. If you are curious to learn more about Kubernetes get in touch with upGrad