Kubernetes Interview Questions and Answers (1–25)

## 1. What is Kubernetes?

Kubernetes is an open-source container orchestration platform that automates deployment, scaling, and management of containerized applications.

## 2. What are the main components of Kubernetes architecture?

Kubernetes architecture has two major components:

1. Control Plane (API Server, Scheduler, Controller Manager, etcd)

2. Nodes (Kubelet, Kube Proxy, Container Runtime)

## 3. What is a Pod in Kubernetes?

A Pod is the smallest and simplest unit in Kubernetes. It encapsulates one or more containers, storage resources, and a network IP.

## 4. What is the difference between a Pod and a Container?

A container runs a single process in isolation. A pod can host one or more containers that share the same network and storage context.

## 5. What is a ReplicaSet in Kubernetes?

A ReplicaSet ensures a specified number of pod replicas are running at any time.

## 6. What is a Deployment in Kubernetes?

A Deployment provides declarative updates to applications. It manages ReplicaSets and enables rollout, rollback, and scaling.

## 7. What is a Service in Kubernetes?

A Service is an abstraction that defines a logical set of pods and a policy by which to access them.

## 8. What are the types of Services in Kubernetes?

1. ClusterIP

2. NodePort

3. LoadBalancer

4. ExternalName

## 9. What is the purpose of etcd in Kubernetes?

etcd is a distributed key-value store that holds all cluster data and configuration. It's the single source of truth for Kubernetes.

## 10. What is the role of the Kubelet?

Kubelet is an agent running on each node. It ensures the containers described in PodSpecs are running and healthy.

## 11. What is the purpose of the kube-proxy?

kube-proxy maintains network rules on each node to allow communication to your pods. It uses iptables or IPVS for routing.

## 12. What is the function of the Scheduler in Kubernetes?

The Scheduler assigns newly created pods to nodes based on resource requirements and policies.

## 13. What is the Controller Manager?

The Controller Manager runs controller processes to handle routine tasks like node management, replication, etc.

## 14. What is a Namespace in Kubernetes?

Namespaces provide a mechanism for isolating groups of resources within a single cluster.

## 15. What are ConfigMaps and Secrets?

ConfigMaps: Store non-confidential data.

Secrets: Store confidential data like passwords, tokens, and keys.

## 16. What’s the role of the kubectl command-line tool?

kubectl allows users to interact with Kubernetes clusters. It supports creating, updating, and inspecting resources.

## 17. What is a DaemonSet and when would you use it?

DaemonSet ensures a pod runs on all (or some) nodes. Used for log collection, monitoring, or networking services.

## 18. Explain the concept of Taints and Tolerations. Explain in detail about the effects in taint.

Taints are applied to nodes to repel pods. Tolerations are applied to pods to let them be scheduled on tainted nodes.

Effects: NoSchedule, PreferNoSchedule, NoExecute.

## 19. How do you secure a Kubernetes cluster?

1. Use RBAC for access control

2. Apply NetworkPolicies

3. Encrypt Secrets

4. Use PodSecurity Standards

5. Regular patching

## 20. What’s a Helm chart and how does it help with Kubernetes deployments?

Helm charts package Kubernetes resources into a single unit. It enables repeatable deployment, versioning, and configuration.

## 21. Explain about: I. Resources (Requests and Limits) II. Taints and Tolerations

I. Requests are minimum guaranteed resources; Limits are maximum allowed.

II. Taints repel pods; Tolerations allow pods to schedule on tainted nodes.

## 22. Can you write a pod definition file and a deployment file?

Pod YAML:

apiVersion: v1

kind: Pod

metadata:

name: mypod

spec:

containers:

- name: nginx-container

image: nginx

ports:

- containerPort: 80

Deployment YAML:

apiVersion: apps/v1

kind: Deployment

metadata:

name: myapp-deployment

spec:

replicas: 3

selector:

matchLabels:

app: myapp

template:

metadata:

labels:

app: myapp

spec:

containers:

- name: myapp-container

image: nginx

ports:

- containerPort: 80

## 23. How can you scale up and scale down the pods?

Use kubectl scale or edit deployment YAML.

Autoscaling: kubectl autoscale deployment --min=2 --max=10 --cpu-percent=80

## 24. When a user enters `kubectl get pods` what happens?

kubectl contacts the API server, which authenticates the user, fetches pod data from etcd, and returns it to kubectl.

## 25. Explain about any 3 controllers in Kubernetes control manager.

1. ReplicaSet Controller: Ensures desired number of pods.

2. Deployment Controller: Manages rollouts/rollbacks.

3. Node Controller: Manages node health and pod eviction.