Kubernetes Interview Questions and Answers

Q: What is Kubernetes and why do we use it?

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

Q: What is a Pod?

A: A Pod is the smallest deployable unit in Kubernetes, representing one or more containers that share storage, network, and a lifecycle.

Q: How does a Deployment differ from a Pod?

A: A Deployment manages ReplicaSets and Pods, enabling updates and rollbacks; a Pod is just a single instance of a running process.

Q: What is a Service in Kubernetes?

A: A Service is a stable abstraction that exposes a set of Pods as a network service.

Q: What are the types of Services? (ClusterIP, NodePort, LoadBalancer)

A: ClusterIP (default, internal), NodePort (exposes via a static port on each Node), LoadBalancer (provisions an external IP via cloud provider).

Q: What is a ConfigMap?

A: A ConfigMap is used to store non-sensitive configuration data in key-value pairs.

Q: What is a Secret in Kubernetes?

A: A Secret is an object used to store sensitive data like passwords, tokens, or keys.

Q: What does `kubectl run` do?

A: It creates and runs a single Pod based on the provided container image.

Q: Whats the difference between `kubectl apply` and `kubectl create`?

A: `apply` is used for declarative configurations and supports updates; `create` is used for imperative creation only.

Q: How do you scale a Deployment in Kubernetes?

A: Use `kubectl scale deployment <name> --replicas=<n>` or update the replica count in the YAML and apply it.

Q: What is an Ingress? How is it different from a Service?

A: Ingress manages HTTP/HTTPS access to services; it supports routing rules and TLS, while Services expose Pods internally or externally.

Q: How does Kubernetes handle service discovery?

A: Kubernetes uses DNS to map service names to ClusterIP addresses via kube-dns or CoreDNS.

Q: What is a Namespace and why is it used?

A: Namespaces logically divide cluster resources for multi-tenant environments.

Q: How does a Liveness vs. Readiness probe work?

A: Liveness checks if a container is alive; readiness checks if its ready to serve traffic.

Q: How does the Scheduler decide where to place a Pod?

A: Based on resource availability, node selectors, affinities, taints/tolerations, and constraints.

Q: What is the role of a Kubelet?

A: Kubelet ensures that containers described in PodSpecs are running and healthy on a node.

Q: What is the function of kube-proxy?

A: kube-proxy manages network rules on nodes to allow communication to and from services.

Q: How do rolling updates and rollbacks work in Kubernetes?

A: Deployments manage rolling updates using `maxUnavailable` and `maxSurge`, and allow rollbacks via `kubectl rollout undo`.

Q: Whats the difference between StatefulSet and Deployment?

A: StatefulSets maintain sticky identities and ordered deployment; Deployments are for stateless apps.

Q: What is a DaemonSet? Give a real-world use case.

A: DaemonSet ensures a Pod runs on all (or some) nodes. Example: log collection agents like Fluentd.

Q: How does Kubernetes handle networking between Pods across nodes?

A: Through CNI plugins like Calico/Flannel that enable flat networking, allowing Pod-to-Pod communication without NAT.

Q: Explain CNI (Container Network Interface) what plugins have you used?

A: CNI is a spec for configuring network interfaces in containers; common plugins include Calico, Flannel, and Cilium.

Q: What is a sidecar container? Give an example.

A: A sidecar is a helper container in the same Pod that augments the main container. Example: Envoy for service mesh.

Q: How do you secure traffic between services in a cluster?

A: Using NetworkPolicies, TLS encryption, mTLS with service meshes, and RBAC for access control.

Q: How does Kubernetes handle secrets securely? How would you improve it?

A: Secrets are base64-encoded and stored in etcd. Use encryption at rest, RBAC, and external secret managers.

Q: How would you debug a CrashLoopBackOff error?

A: Check pod logs (`kubectl logs`), describe pod, examine events, check probes, and container entrypoints.

Q: What is a taint and a toleration?

A: Taints prevent Pods from scheduling on nodes unless they have matching tolerations.

Q: How do you configure node affinity or anti-affinity?

A: Using `nodeAffinity`, `preferredDuringScheduling` and `requiredDuringScheduling` rules in Pod specs.

Q: Explain how Kubernetes handles horizontal and vertical pod autoscaling.

A: Horizontal scales pod count using CPU/memory metrics; vertical adjusts resource requests/limits of a Pod.

Q: How would you design a highly available Kubernetes cluster?

A: Use multiple control plane nodes, regional failover, load-balanced API server, and persistent storage.