

Kubernetes Basics and Concepts (General/Popular)

1. What is Kubernetes?
2. What are the main components of a Kubernetes cluster?
3. What is a node in Kubernetes?
4. Explain the role of the master node.
5. What is a Pod?
6. How is a Pod different from a container?
7. What is a ReplicaSet?
8. What is a Deployment?
9. What is a Namespace?
10. What is a Service?
11. What types of Services exist in Kubernetes?
12. What is a ConfigMap?
13. What is a Secret?
14. What is a DaemonSet?
15. What is a StatefulSet?
16. What is a Job?
17. What is a CronJob?
18. How does Kubernetes achieve high availability?
19. How do you expose a Pod to the outside world?
20. What is kube-proxy?

Cluster Architecture and Internals

21. What is etcd and what role does it play in Kubernetes?
22. What is the kube-apiserver?
23. What is the kube-scheduler?
24. What is the kube-controller-manager?
25. How does the control plane communicate with worker nodes?
26. What is kubelet?
27. What is a container runtime? Name a few.
28. What is CNI and why is it important?
29. What happens when you run `kubectl get pods`?
30. What is the purpose of kube-dns/CoreDNS?

YAML, Manifests, and Declarative Configurations

31. What are the main sections of a Pod YAML file?
32. How do you define environment variables in a Pod manifest?
33. How do you mount a volume in a Pod?
34. What is the difference between `args` and `command` in Pod spec?
35. How do you define resource limits and requests?
36. How do you use ConfigMap in a deployment?
37. How do you use Secret in a deployment?

- 38. What is the difference between an `initContainer` and a regular container?
 - 39. How do you create a multi-container Pod?
 - 40. How do you use liveness and readiness probes?
-

Deployments, Rollouts, and Updates

- 41. How do you update a deployment?
 - 42. What is a rolling update?
 - 43. How do you perform a rollback?
 - 44. What is a canary deployment?
 - 45. How do you perform a blue-green deployment in Kubernetes?
 - 46. How does Kubernetes handle rolling restarts?
 - 47. What is a strategy for zero-downtime deployment?
 - 48. What happens if a deployment update fails?
 - 49. How do you monitor rollout status?
 - 50. What is the difference between `kubectl apply` and `kubectl create`?
-

Networking and Services

- 51. What is a ClusterIP service?
- 52. What is a NodePort service?
- 53. What is a LoadBalancer service?

- 54. How do you access a ClusterIP service from outside the cluster?
 - 55. What is an Ingress?
 - 56. What is an Ingress controller?
 - 57. How do you secure communication between Pods?
 - 58. What is network policy?
 - 59. How do you limit traffic between namespaces?
 - 60. How do you debug a service that is not accessible?
-

Volumes, Storage, and Data Persistence

- 61. What is a PersistentVolume (PV)?
 - 62. What is a PersistentVolumeClaim (PVC)?
 - 63. How do you use dynamic provisioning for PVs?
 - 64. What types of storage backends are supported in Kubernetes?
 - 65. What is the difference between emptyDir and hostPath?
 - 66. How do you backup and restore PV data?
 - 67. How do you resize a PVC?
 - 68. What is a StorageClass?
 - 69. How do you attach a volume to a specific Pod?
 - 70. What happens to data in a Pod's local storage if the Pod is deleted?
-

Configuration, Secrets, and Environment Management

- 71. How do you manage environment-specific configurations in Kubernetes?
 - 72. How do you inject a secret into a container as an environment variable?
 - 73. How do you rotate secrets in Kubernetes?
 - 74. How do you manage sensitive data in GitOps workflows?
 - 75. How do you update a ConfigMap or Secret without restarting the Pod?
 - 76. What is projected volume?
 - 77. How do you use downward API?
 - 78. How do you set up application configuration for blue-green deployments?
 - 79. How do you manage application configuration at scale?
 - 80. How do you troubleshoot when configuration changes are not reflected?
-

Scaling, Auto-Scaling, and Resource Management

- 81. What is the difference between horizontal and vertical pod autoscaling?
- 82. How do you enable HPA (Horizontal Pod Autoscaler)?
- 83. What metrics can be used for HPA?
- 84. How do you scale up/down deployments manually?
- 85. How does cluster autoscaler work?
- 86. What is resource quota?
- 87. What happens when resource limits are not set?
- 88. How do you handle pod eviction due to resource pressure?
- 89. How do you diagnose OOMKilled pods?

90. How do you ensure fair resource allocation in multi-tenant clusters?

Monitoring, Logging, and Troubleshooting

91. How do you monitor Kubernetes clusters?

92. What tools do you use for logging in Kubernetes?

93. How do you access Pod logs?

94. How do you aggregate logs from multiple Pods?

95. How do you monitor application health?

96. How do you collect metrics from the cluster?

97. How do you set up alerts for failed pods?

98. What is the recommended approach for centralized monitoring?

99. How do you debug a crashing Pod?

100. How do you troubleshoot DNS issues in a cluster?

Security, RBAC, and Access Control

101. What is RBAC?

102. How do you create a service account?

103. How do you bind a role to a user or group?

104. What is a ClusterRole and ClusterRoleBinding?

105. How do you restrict user access to specific namespaces?

- 106. How do you audit API access in Kubernetes?
 - 107. How do you secure communication between cluster components?
 - 108. What is pod security policy?
 - 109. How do you enforce network policies?
 - 110. How do you manage secrets securely?
-

Upgrades, Maintenance, and Disaster Recovery

- 111. How do you upgrade a Kubernetes cluster?
 - 112. How do you backup and restore etcd?
 - 113. What steps do you follow before a cluster upgrade?
 - 114. What happens to running Pods during a node drain?
 - 115. How do you evict Pods from a node before maintenance?
 - 116. How do you recover from a lost etcd quorum?
 - 117. How do you migrate workloads between clusters?
 - 118. How do you upgrade application workloads with minimal downtime?
 - 119. How do you handle failed node recovery?
 - 120. What is the process for scaling down unused nodes safely?
-

CI/CD, GitOps, and Workflow Integration

- 121. How do you deploy applications to Kubernetes using CI/CD?

- 122. What is GitOps?
 - 123. How do you implement GitOps with Kubernetes?
 - 124. How do you handle image versioning in deployments?
 - 125. How do you perform blue-green or canary deployments in CI/CD?
 - 126. How do you automate Kubernetes manifest generation?
 - 127. How do you validate manifests in CI?
 - 128. How do you roll back a failed deployment via CI?
 - 129. How do you handle secrets in a GitOps pipeline?
 - 130. What are some popular tools for Kubernetes CI/CD?
-

Cloud Providers and Managed Kubernetes

- 131. What is GKE/EKS/AKS?
- 132. How does managed Kubernetes differ from self-hosted?
- 133. How do you use cloud provider integrations (e.g., load balancers, storage)?
- 134. How do you authenticate to a managed Kubernetes cluster?
- 135. How do you provision clusters as code?
- 136. How do you manage IAM roles in cloud Kubernetes?
- 137. How do you use node pools or node groups?
- 138. How do you configure cluster auto-upgrade in the cloud?
- 139. What are cloud-specific limitations or differences?
- 140. How do you monitor cloud spend for Kubernetes workloads?

Scenario-Based & Real-World Problems (Troubleshooting and Design)

Day-to-Day and Real-World Challenges

141. Your Pod is in CrashLoopBackOff. How do you debug and resolve this?
142. Your deployment rollout is stuck. How do you troubleshoot?
143. Service is not reachable from outside. What are your debugging steps?
144. Pod is stuck in Pending state. What are possible reasons and solutions?
145. You need to enforce that only certain images are allowed in your cluster. How would you do this?
146. An app requires persistent storage that must survive Pod rescheduling. What's your approach?
147. You need to restrict access to a namespace for a team. How would you do it?
148. During node maintenance, some Pods are not moving—why and how to resolve?
149. How would you migrate a database from one cluster to another with minimal downtime?
150. How do you handle configuration drift between environments?

Extra Real-World Scenarios & Behavioral Questions

- You are asked to rotate all secrets cluster-wide after a security breach. How do you proceed?
- An upgrade is scheduled but you can't afford any downtime. What steps do you take?
- You notice resource starvation on some nodes. How do you rebalance workloads?

- You have hundreds of microservices—how do you maintain YAML manifests at scale?
 - A new developer accidentally deleted a namespace in production. How do you prevent this in the future?
 - Users report latency only at peak times—what's your investigation process?
 - A service is accessible within the cluster but not from outside. What could be wrong?
 - Your cluster frequently hits the Pod limit per node. What steps do you take?
 - How do you securely inject runtime secrets into running Pods without restart?
 - Explain your approach to Kubernetes cost optimization.
-

Preparation Advice

- **Practice hands-on:** Try solving these in your own test cluster.
- **Scenario questions:** Always answer with *what you would check*, *commands you'd run* (`kubectl/logs`), and *what you would look for*.
- **Behavioral:** Prepare short stories for real outages or challenges you've handled.
- **Best practices:** Always mention things like monitoring, least-privilege, version control for manifests, and disaster recovery.