**Introduction to Kubernetes**

1. What is Kubernetes?

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

1. What is a container?

A container is a lightweight, standalone, executable software package that includes everything needed to run an application, including code, runtime, system tools, libraries, and settings.

1. What are the benefits of using Kubernetes?

Kubernetes automates application deployment, scaling, and management, making it easy to deploy and manage container-based applications at scale. Other benefits include:  
Simplified application management  
Improved scaling and availability  
Easy deployment and rollback  
Improved resource utilization  
Increased portability and flexibility

1. What is a Kubernetes cluster?
2. A Kubernetes cluster is a set of nodes that run containerized applications managed by the Kubernetes control plane.
3. What is a node in Kubernetes?
4. A node is a worker machine in Kubernetes that runs containerized applications.
5. What is a pod in Kubernetes?
6. A pod is the smallest deployable unit in Kubernetes that represents a single instance of a running process in a container.  
   Kubernetes Architecture
7. What is the Kubernetes control plane?
8. The Kubernetes control plane is a set of components that manages and orchestrates the Kubernetes cluster. It includes the following components:  
   API server  
   etcd  
   kube-scheduler  
   kube-controller-manager  
   cloud-controller-manager
9. What is the API server in Kubernetes?
10. The API server is the front-end interface for the Kubernetes control plane that exposes the Kubernetes API.
11. What is etcd in Kubernetes?
12. etcd is a distributed, reliable, and highly available key-value store used to store the configuration data for the Kubernetes cluster.
13. What is the Kubernetes scheduler?
14. The Kubernetes scheduler is responsible for scheduling pods to run on available nodes in the cluster based on available resources and other scheduling requirements.
15. What is the kube-controller-manager?
16. The kube-controller-manager is responsible for running various controller processes that monitor the state of the cluster and make changes as necessary.
17. What is the cloud-controller-manager?
18. The cloud-controller-manager is responsible for managing integration with cloud providers, such as AWS, GCP, or Azure.
19. What is a Kubernetes worker node?
20. A Kubernetes worker node is a physical or virtual machine that runs containerized applications and services. It includes the following components:  
    Kubelet  
    kube-proxy  
    container runtime
21. What is the kubelet in Kubernetes?
22. The kubelet is an agent that runs on each node and communicates with the Kubernetes API server to manage the container lifecycle.
23. What is the kube-proxy in Kubernetes?
24. The kube-proxy is responsible for managing network routing between pods and services in the Kubernetes cluster.
25. What is a container runtime in Kubernetes?
26. A container runtime is responsible for starting and stopping containers on a node. Examples include Docker, containerd, and CRI-O.
27. Why use namespace in Kubernetes?
28. Namespaces in Kubernetes are used for dividing cluster resources between users. It helps the environment where more than one user spread projects or teams and provides a scope of resources.
29. **Kubernetes Networking**
30. What is a Kubernetes service?
31. A Kubernetes service is an abstraction layer that exposes a set of pods as a network service, allowing them to communicate with each other and with other services outside the cluster.
32. What is a Kubernetes DNS?
33. Kubernetes DNS is a service that provides DNS resolution for services and pods in a Kubernetes cluster.
34. What is a pod network in Kubernetes?
35. A pod network is a network overlay that connects pods in a Kubernetes cluster.
36. What is the Kubernetes CNI (Container Networking Interface)?
37. The Kubernetes CNI is a specification that defines a standardized interface for integrating with container networking plugins.  
    Deploying Applications in Kubernetes
38. What is a Kubernetes deployment?
39. A Kubernetes deployment defines a desired state for a group of replicas of a pod, and manages the rollout and rollback of updates to the pod replicas.
40. What is a Kubernetes pod template?
41. A Kubernetes pod template defines the desired configuration for a pod, including the container image, environment variables, and other settings.
42. What is a Kubernetes replica set?
43. A Kubernetes replica set ensures that a specified number of replicas of a pod are running at any given time.
44. What is a Kubernetes stateful set?
45. A Kubernetes stateful set manages the deployment, scaling, and ongoing state of a set of stateful pods, such as databases or other stateful applications.
46. What is a Kubernetes daemon set?
47. A Kubernetes daemon set ensures that a specific pod runs on all or some nodes in the cluster.
48. What is a Kubernetes job?
49. A Kubernetes job runs a specific task to completion, such as running a batch job or performing a data processing task.
50. **Kubernetes Scheduling and Scaling**
51. What is Kubernetes scheduling?
52. Kubernetes scheduling is the process of assigning a running pod to a node in the cluster.
53. What is Kubernetes scheduling policy?
54. Kubernetes scheduling policy is a set of rules and criteria used to determine which node in the cluster should run a specific pod.
55. What is a Kubernetes affinities?
56. Kubernetes affinities are rules that determine the preferred scheduling of pods based on various factors, such as the existence of a specific data volume or the location of a specific node.
57. What is a Kubernetes anti-affinities?
58. Kubernetes anti-affinities are rules that determine the preferred scheduling of pod based on factors that should be avoided, such as running two replicas of a pod on the same node.
59. What is Kubernetes horizontal pod autoscaling (HPA)?
60. Kubernetes HPA automatically scales the number of replicas of a pod based on the current demand for resources.
61. What is Kubernetes Vertical Pod Autoscaling (VPA)?
62. Kubernetes VPA automatically adjusts the resource requirements of a pod based on the current resource usage.
63. What is Kubernetes cluster autoscaling?
64. Kubernetes cluster autoscaling automatically scales the number of nodes in a cluster based on the current demand for resources.  
    Monitoring and Logging in Kubernetes
65. What is Kubernetes monitoring?
66. Kubernetes monitoring is the process of monitoring the health and performance of a Kubernetes cluster and its applications.
67. What is Kubernetes logging?
68. Kubernetes logging is the process of collecting and analyzing the logs generated by the applications and services running in a Kubernetes cluster.
69. What is Kubernetes Prometheus?
70. Kubernetes Prometheus is an open-source monitoring and alerting toolkit that collects metrics and data from the Kubernetes API server.
71. What is Kubernetes Grafana?
72. Kubernetes Grafana is an open-source data visualization and analysis tool that provides real-time monitoring and analysis of Kubernetes clusters.
73. What is Kubernetes Fluentd?
74. Kubernetes Fluentd is an open-source data collection and forwarding tool that aggregates logs and sends them to a central location for analysis and storage.
75. What is Kubernetes Kibana?
76. Kubernetes Kibana is an open-source data visualization and analysis tool that provides real-time analysis of logs and other data generated by Kubernetes clusters.
77. **Kubernetes Security**
78. What is Kubernetes RBAC (Role-Based Access Control)?
79. Kubernetes RBAC is a method of controlling access to Kubernetes resources based on user roles and permissions.
80. What is Kubernetes TLS (Transport Layer Security)?
81. Kubernetes TLS is a security protocol used to secure client-server communications within a Kubernetes cluster.
82. What is Kubernetes network policies?
83. Kubernetes network policies are rules that control the flow of network traffic between pods and services within a Kubernetes cluster.
84. What is Kubernetes pod security policies?
85. Kubernetes pod security policies are a set of policies that control the security settings for pods deployed in a Kubernetes cluster.
86. What is Kubernetes secrets?
87. Kubernetes secrets are a secure way to store sensitive information, such as passwords, API keys, and other authentication tokens, used by applications running in a Kubernetes cluster.
88. What is Kubernetes pod security context?
89. Kubernetes pod security context provides a way to set security-related attributes on a per-pod basis, such as user and group IDs, and file permissions.
90. **Kubernetes Tools and APIs**
91. What is kubectl?
92. kubectl is the command-line tool used to interact with a Kubernetes cluster.
93. What is the Kubernetes API?
94. The Kubernetes API is a RESTful API used to manage and operate Kubernetes clusters.
95. What is Kubernetes Helm?
96. Kubernetes Helm is a package manager for Kubernetes that helps you deploy, manage and upgrade Kubernetes applications.
97. What is Kubernetes Dashboard?
98. Kubernetes Dashboard is a web-based user interface for managing and monitoring Kubernetes clusters.
99. **Debugging and Troubleshooting in Kubernetes**
100. What is Kubernetes pod readiness probe?
101. Kubernetes pod readiness probe is used to determine if a pod is ready to serve traffic.
102. What is Kubernetes pod liveness probe?
103. Kubernetes pod liveness probe is used to determine if a pod is alive and running.
104. How do you troubleshoot a Kubernetes pod?
105. Troubleshooting a Kubernetes pod involves checking logs, investigating resource utilization, and inspecting the pod status and events.
106. What is Kubernetes kubectl logs?
107. Kubernetes kubectl logs is the command to retrieve the logs generated by a specific pod.
108. What is Kubernetes kubectl describe?
109. Kubernetes kubectl describe is the command to get detailed information about a Kubernetes object, such as a pod, replication controller, or service.  
     Kubernetes Cluster Administration
110. What is Kubernetes cluster management?
111. Kubernetes cluster management involves configuring and maintaining the Kubernetes control plane, worker nodes, and network settings.
112. What is Kubernetes API server authorization?
113. Kubernetes API server authorization controls who can access and perform actions against the Kubernetes API server.
114. What is Kubernetes cluster backup and restore?
115. Kubernetes cluster backup and restore involves backing up and restoring the configuration and data stored in the Kubernetes objects, such as pods, services, and deployments.
116. How does Kubernetes perform a rolling update?
117. Kubernetes performs a rolling update by gradually upgrading the replicas of a pod, ensuring that the application remains available and responsive during the update.
118. **Kubernetes Best Practices**
119. What are the best practices for deploying applications in Kubernetes?
120. Best practices for deploying applications in Kubernetes include:  
     Using declarative deployment methods, such as Deployments or Helm charts  
     Separating concerns between services by deploying them in separate namespaces  
     Using liveness and readiness probes to ensure the health of the application  
     Setting resource limits and requests to ensure adequate resources for the application
121. What are the best practices for Kubernetes cluster security?
122. Best practices for Kubernetes cluster security include:  
     Implementing Role-Based Access Control (RBAC)  
     Using network policies to control traffic within the cluster  
     Restricting external access to cluster components and API servers  
     Implementing secured node access and communication between nodes in the cluster
123. What are the best practices for Kubernetes performance optimization?
124. Best practices for Kubernetes performance optimization include:  
     Setting resource limits and requests to ensure adequate resources for the application  
     Using horizontal and vertical pod autoscaling  
     Optimizing container images for size and performance  
     Monitoring and tuning system and application performance  
     Developing with Kubernetes
125. What is Kubernetes operator?
126. Kubernetes operator is an extension of the Kubernetes API that enables the automation of complex application or cluster management operations.
127. What is Kubernetes custom resource definition?
128. Kubernetes custom resource definition is a way to extend the Kubernetes API with custom resources and APIs that are specific to a particular application or framework.
129. What is Kubernetes CRD controller?
130. Kubernetes CRD controller is used to define the behavior of the custom resources and their interactions with other Kubernetes components.  
     Kubernetes Networking
131. What is Kubernetes Istio?
132. Kubernetes Istio is an open-source service mesh that provides traffic management, observability, and security for microservices-based applications.
133. What is Kubernetes service mesh?
134. Kubernetes service mesh is a dedicated infrastructure layer for managing service-to-service communication within a Kubernetes cluster.
135. What is Kubernetes Ingress?
136. Kubernetes Ingress is an API object that defines rules for directing inbound traffic to Kubernetes services.
137. What is Kubernetes gateway?
138. Kubernetes gateway is a network entry point that manages incoming and outgoing traffic for a service mesh.
139. **Kubernetes Runtime**
140. What is Kubernetes containerd?
141. Kubernetes containerd is a lightweight, non-intrusive container runtime for Kubernetes.
142. What is Kubernetes CRI-O?
143. Kubernetes CRI-O is a container runtime designed specifically for Kubernetes, providing a lightweight and fast container runtime for Kubernetes environments.
144. What is Kubernetes KubeVirt?
145. Kubernetes KubeVirt is an open-source virtual machine runtime for Kubernetes, allowing users to deploy and manage virtual machines alongside Kubernetes workloads.
146. What is Kubernetes Kata Containers?
147. Kubernetes Kata Containers is a secure container runtime option for Kubernetes, providing hardware-implemented isolation to ensure security and isolation between containers.
148. **Kubernetes Cloud-Native Development**
149. What is Kubernetes cloud-native development?
150. Kubernetes cloud-native development is a software development methodology that maximizes the use of Kubernetes to build, deploy, and manage cloud-native applications.
151. What is Kubernetes software development kit (SDK)?
152. Kubernetes software development kit (SDK) is a set of tools and libraries that help developers build, deploy and manage cloud-native applications on Kubernetes.
153. What is Kubernetes Helm?
154. Kubernetes Helm is a package manager for Kubernetes that provides templating and deployment automation for cloud-native applications.
155. **Miscellaneous**  
     What is the difference between a deployment and a stateful set in Kubernetes?
156. Deployments are used for stateless applications, while stateful sets are used for stateful applications, such as databases or other applications that require persistent and stable storage.
157. What is Kubernetes Configuration Management?
158. Kubernetes Configuration Management is the automated management of configuration files and settings across a Kubernetes cluster.
159. What is Kubernetes container orchestration?
160. Kubernetes container orchestration is the automated process of deploying, scaling, and managing containerized applications in a Kubernetes cluster.
161. What is Kubernetes containerization?
162. Kubernetes containerization is the process of packaging an application and all its dependencies into a container for deployment and management.
163. What is Kubernetes cloud deployment?
164. Kubernetes cloud deployment is the deployment of Kubernetes clusters on cloud platforms, such as AWS, Azure, or GCP.
165. What is Kubernetes on-premises deployment?
166. Kubernetes on-premises deployment is the deployment of Kubernetes clusters on private or enterprise servers and data centers.