Kubernetes MCQs – Questions & Explanations

Questions

1. Which Kubernetes component is the "single source of truth" for cluster state? A. kubelet B. kube-scheduler C. etcd D. kube-proxy
2. What happens if the kube-apiserver goes down? A. Pods stop running B. Existing workloads run, but no new changes can be applied C. Scheduler takes over API tasks D. kube-proxy replaces apiserver
3. What is the role of the kube-scheduler? A. Runs pods B. Decides node placement for pods C. Manages network rules D. Stores cluster state
4. Which best describes the kubelet? A. Stores cluster data B. Assigns pods to nodes C. Ensures containers on a node run as specified D. Manages traffic routing
5. What does kube-proxy do? A. Schedules pods to nodes B. Stores cluster metadata C. Implements Services by routing traffic to pod IPs D. Monitors pod health
6. Which describes the function of a CNI plugin like Calico or Flannel? A. Runs containers B. Provides pod networking across nodes C. Schedules workloads D. Monitors logs
7. Which Service type is only reachable inside the cluster? A. ClusterIP B. NodePort C. LoadBalancer D. ExternalName
8. What role does CoreDNS play in Kubernetes? A. Stores logs B. Provides pod-to-pod networking C. Resolves service names to ClusterIPs D. Runs system metrics
9. How are ConfigMaps and Secrets different? A. ConfigMaps are for sensitive data, Secrets for non-sensitive B. Secrets can be encrypted, ConfigMaps are plain configs C. ConfigMaps are stored in files, Secrets are not D. ConfigMaps can't be mounted into pods
10. What is the difference between a Deployment and a StatefulSet? A. Deployments are for stateless apps; StatefulSets provide stable identity and storage B. StatefulSets scale faster C. Deployments only support 1 replica D. StatefulSets don't need volumes
11. When an external client accesses a NodePort Service, which component first processes the packet on the node? A. kubelet B. kube-proxy C. Ingress D. container runtime
12. In a LoadBalancer Service, what is the correct sequence? A. LB → NodePort → kube-proxy → Pod IP B. LB → kubelet → Pod C. LB → kube-proxy → Service → Pod D. LB → Ingress → Pod

13. How does kube-proxy decide which pod to send a request to? A. By hashing client IP B. Round-robin or IPVS load-balancing C. Randomly chosen by kubelet D. DNS chooses pod directly

14. A request reaches a pod but no response returns. What is the likely issue? A. Service not created B. Readiness probe failed C. Misconfigured CNI return path D. kube-proxy missing

15. What is the correct trajectory for an in-cluster pod-to-service request? A. Pod → DNS (CoreDNS) → ClusterIP → kube-proxy → Pod B. Pod → kubelet → Pod directly C. Pod → kube-proxy → DNS → Pod D. Pod → Ingress → ClusterIP → Pod

16. In Kubernetes, a Pod can contain: A. Exactly 1 container B. Multiple containers sharing the same network namespace C. Only sidecar containers D. Containers that must run on different nodes

17. What is the default restart policy for a Kubernetes Pod? A. Never B. OnFailure C. Always D. Manual

18. Which object in Kubernetes ensures a set of identical Pods are always running? A. Pod B. ReplicaSet C. Service D. ConfigMap

19. Kubernetes Service of type `ClusterIP` is accessible: A. Only within the cluster B. From external network C. Only from Pod's namespace D. Nowhere unless exposed

20. Which Kubernetes controller is responsible for rolling updates and rollbacks? A. ReplicaSet B. Deployment C. StatefulSet D. DaemonSet

21. Which Kubernetes component is responsible for scheduling Pods to nodes? A. Controller Manager B. Scheduler C. Kubelet D. API Server

22. ConfigMap vs Secret: which statement is correct? A. Both are encrypted by default B. Secrets are base64-encoded, ConfigMaps store plain text C. ConfigMaps require RBAC, Secrets do not D. Secrets cannot be mounted in Pods

23. If a Pod crashes, which component detects it and attempts to restart it? A. Scheduler B. Kubelet C. Controller Manager D. etcd

24. Which Kubernetes object ensures only one Pod is scheduled per node? A. DaemonSet B. StatefulSet C. Job D. Deployment

25. Kubernetes Ingress is used to: A. Load balance between Pods within a namespace B. Manage traffic rules and expose services externally C. Replace Service objects completely D. Create persistent storage

---

Answers with Explanations

1. **C. etcd** – etcd is a distributed key-value store used as the **single source of truth** for all cluster state.
2. **B. Existing workloads run, but no new changes can be applied** – kube-apiserver handles API requests; if it goes down, workloads continue but no changes are possible.
3. **B. Decides node placement for pods** – kube-scheduler assigns pods to nodes based on resources and constraints.
4. **C. Ensures containers on a node run as specified** – kubelet ensures that pods defined in specs are running on its node.
5. **C. Implements Services by routing traffic to pod IPs** – kube-proxy maintains rules for service-to-pod traffic.
6. **B. Provides pod networking across nodes** – CNI plugins like Calico/Flannel enable cross-node pod networking.
7. **A. ClusterIP** – default service type, accessible only inside the cluster.
8. **C. Resolves service names to ClusterIPs** – CoreDNS provides internal DNS resolution.
9. **B. Secrets can be encrypted, ConfigMaps are plain configs** – Secrets are base64-encoded and can be encrypted; ConfigMaps are plain text.
10. **A. Deployments are for stateless apps; StatefulSets provide stable identity and storage** – StatefulSets give pods persistent IDs and volumes.
11. **B. kube-proxy** – kube-proxy processes the traffic first when using NodePort.
12. **A. LB → NodePort → kube-proxy → Pod IP** – external traffic flows through LoadBalancer → NodePort → kube-proxy → Pod.
13. **B. Round-robin or IPVS load-balancing** – kube-proxy distributes traffic across pods.
14. **C. Misconfigured CNI return path** – if requests reach pods but responses don't return, it's often a CNI networking issue.
15. **A. Pod → DNS (CoreDNS) → ClusterIP → kube-proxy → Pod** – internal service communication uses DNS + service cluster IP.
16. **B. Multiple containers sharing the same network namespace** – pods can have sidecar containers sharing storage and networking.
17. **C. Always** – default pod restart policy ensures high availability.
18. **B. ReplicaSet** – ReplicaSet ensures the desired number of identical pods are running.
19. **A. Only within the cluster** – ClusterIP is internal-only.
20. **B. Deployment** – Deployments handle rollouts and rollbacks for pods.
21. **B. Scheduler** – schedules pods onto nodes.
22. **B. Secrets are base64-encoded, ConfigMaps store plain text** – main difference in storage format.
23. **B. Kubelet** – detects crashed pods and restarts them.
24. **A. DaemonSet** – ensures exactly one pod per node.

25. **B. Manage traffic rules and expose services externally** – Ingress manages external HTTP/HTTPS traffic into cluster services.

Would you like me to also **format this into a neat Markdown table** (Questions | Correct Answer | Explanation) so it's easier for quick revision?