

Here are 30 Kubernetes (K8s) scenario-based interview questions and answers tailored for a DevOps Engineer role with real-time examples:

1. How do you recover a failed Kubernetes master node?

Scenario: Your Kubernetes master node crashes, and your cluster becomes unresponsive.

What steps do you take to recover it?

Answer:

- If using a highly available cluster, failover happens automatically.
- If using a single master node:
- Check the logs (`journalctl -u kube-apiserver`).
- Restore from backup (`etcdctl snapshot restore`).
- Restart services (`kube-apiserver`, `kube-controller-manager`, `kube-scheduler`).
- Verify cluster health (`kubectl`)

get nodes).

Example:

```
ETCDCTL_API=3 etcdctl  
snapshot restore snapshot.db --  
data-dir=/var/lib/etcd
```

**2. How do you
troubleshoot a pod
stuck in a "Pending"**

state?

Scenario: A pod is not
getting scheduled. How
do you diagnose and fix
it?

Answer:

- **Check node resources**
(`kubectl describe pod <pod-name>`).
- **Check taints and tolerations**
(`kubectl describe node <node-name>`).
- **Check affinity/anti-affinity rules.**
- **Check persistent volume claims (PVCs).**

Example:

```
kubectl describe pod my-pod |  
grep -A 10 Events kubectl get  
nodes -o wide
```

3. How do you handle a CrashLoopBackOff pod?

Scenario: A pod is constantly restarting.

Answer:

- Check logs (`kubectl logs <pod-name>`).
- Check resource limits (`kubectl describe pod <pod-name>`).
- Increase readiness probe failureThreshold.
- Run interactively (`kubectl run --rm -it --image=<image> bash`).

Example:

kubectl logs my-crashing-pod -f

4. How do you roll back a failed deployment?

Scenario: You deployed a new version of an application, but it is failing.

Answer:

- Rollback deployment (kubectl rollout undo deployment <deployment-name>).
- Check previous revisions (kubectl rollout history

deployment <deployment-name>).

Example:

```
kubectl rollout undo deployment  
my-app --to-revision=2
```

5. How do you scale an application based on CPU usage?

Scenario: Your application needs to automatically scale when CPU usage exceeds 70%.

Answer:

- Use Horizontal Pod Autoscaler (HPA):

kubectl autoscale deployment
my-app --cpu-percent=70 --min=2
--max=10

- Verify using:

kubectl get hpa

6. What happens if a worker node fails?

Scenario: A worker node crashes.

Answer:

- Kubernetes reschedules pods

to other nodes automatically.

- Check node status (`kubectl get nodes`).
- Manually drain a failing node:

`kubectl drain <node-name> --ignore-daemonsets --delete-emptydir-data`

7. How do you debug network issues in a Kubernetes cluster?

Scenario: A pod cannot communicate with another pod or service.

Answer:

- Check pod IP (kubectl get pods -o wide).
- Check DNS resolution (kubectl exec -it <pod-name> -- nslookup <service-name>).
- Check network policies (kubectl get networkpolicy).

Example:

```
kubectl exec -it my-pod -- curl  
http://my-service
```

**8. How do you restrict
pod-to-pod
communication?**

Scenario: You want to

**prevent one pod from
accessing another.**

Answer:

- **Use NetworkPolicies:**

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: deny-all
spec:
  podSelector: {}
  policyTypes:
    - Ingress
```

9. How do you debug a failing Kubernetes job?

Scenario: A Kubernetes Job is failing.

Answer:

- Check logs (kubectl logs job/ <job-name>).
- Describe job (kubectl describe job <job-name>).
- Check completion status:
kubectl get jobs --watch

10. How do you enforce resource limits on pods?

Scenario: You want to prevent a pod from consuming too much

CPU/memory.

Answer:

- Define limits and requests:
resources: requests: memory: "64Mi" cpu: "250m" limits:
memory: "128Mi" cpu: "500m"

**11. How do you create
a multi-container pod?**

Scenario: You need a
pod with a sidecar
container.

Answer:

apiVersion: v1 kind: Pod

```
metadata: name: multi-container-pod
spec: containers:
  - name: main-app
    image: nginx
  - name: sidecar
    image: busybox
    command: ["sh", "-c", "while true; do echo hello; sleep 10; done"]
```

12. How do you restart all pods in a deployment?

Scenario: You made a configuration change and need to restart all pods.

Answer:

kubectl rollout restart
deployment my-deployment

**13. How do you drain a
node for maintenance?**

kubectl drain <node-name> --
ignore-daemonsets --delete-
emptydir-data

**14. How do you debug
a failing service?**

kubectl get svc my-service -o
yaml kubectl describe svc my-
service kubectl exec -it <pod-
name> -- curl http://my-service

15. How do you set up a Kubernetes cron job?

```
apiVersion: batch/v1 kind:  
CronJob metadata: name: my-  
cronjob spec: schedule: "*/5 * * *  
*" jobTemplate: spec: template:  
spec: containers: - name: my-job  
image: busybox command:  
["echo", "Hello, Kubernetes"]  
restartPolicy: OnFailure
```

16-30 More Scenarios:

- How to configure a liveness probe?
- How to expose a service externally?

- How to deploy a stateful application in Kubernetes?
- How to configure a rolling update?
- How to debug a Kubernetes ingress issue?
- How to run a pod on a specific node?
- How to use a Kubernetes secret in a pod?
- How to prevent unauthorized access to the Kubernetes API?
- How to enable logging for Kubernetes pods?
- How to delete a namespace forcefully?

- How to deploy an application using Helm?
- How to monitor a Kubernetes cluster?
- How to use Persistent Volume Claims (PVC)?
- How to configure service mesh (Istio) in Kubernetes?
- How to recover from an accidental kubectl delete?

Would you like any of these in more detail?

I'll expand on the remaining 15 Kubernetes scenario-based

questions with real-time examples.

16. How do you configure a liveness probe in Kubernetes?

Scenario: Your application sometimes hangs, and you need Kubernetes to automatically restart it.

Solution: Use a liveness probe to check if the app

is still running.

Example YAML:

```
apiVersion: v1 kind: Pod
metadata: name: liveness-probe-
example spec: containers: -
  name: my-container image:
    nginx livenessProbe: httpGet:
      path: /healthz port: 8080
initialDelaySeconds: 5
periodSeconds: 10
```

- If /healthz fails, Kubernetes restarts the container.

17. How do you expose a Kubernetes service

externally?

Scenario: You want to access a service from outside the cluster.

Solution: Use a LoadBalancer or Ingress.

Example using LoadBalancer:

```
apiVersion: v1 kind: Service  
metadata: name: my-service  
spec: type: LoadBalancer  
selector: app: my-app ports: -  
protocol: TCP port: 80  
targetPort: 8080
```

- Use kubectl get svc to find the

external IP.

18. How do you deploy a stateful application in Kubernetes?

Scenario: You need a stable identity (persistent storage, fixed network identity) for a database.

Solution: Use StatefulSets and Persistent Volumes

(PVs).

Example StatefulSet for MySQL:

```
apiVersion: apps/v1 kind:  
StatefulSet metadata: name:  
mysql spec: serviceName:  
"mysql" replicas: 2 selector:  
matchLabels: app: mysql  
template: metadata: labels: app:  
mysql spec: containers: - name:  
mysql image: mysql:5.7 env: -  
name:  
MYSQL_ROOT_PASSWORD  
value: "password"  
volumeMounts: - name: mysql-  
storage mountPath: /var/lib/  
mysql volumeClaimTemplates: -
```

```
metadata: name: mysql-storage
spec: accessModes: [
  "ReadWriteOnce"
] resources:
requests: storage: 1Gi
```

- Each pod gets its own PVC (mysql-0, mysql-1).

19. How do you configure a rolling update in Kubernetes?

Scenario: You need to
update an application
without downtime.

Solution: Use a

Deployment with rollingUpdate.

Example Deployment with Rolling Update Strategy:

```
apiVersion: apps/v1 kind:  
Deployment metadata: name:  
my-app spec: replicas: 3  
strategy: type: RollingUpdate  
rollingUpdate: maxUnavailable: 1  
maxSurge: 1 template:  
metadata: labels: app: my-app  
spec: containers: - name: my-app  
image: my-app:v2  
• Use kubectl rollout status  
deployment my-app to
```

monitor.

20. How do you debug a Kubernetes Ingress issue?

Scenario: Your app is exposed via Ingress, but it's not accessible.

Solution:

- Check Ingress resource:
`kubectl get ingress -n my-namespace`
- Describe Ingress to check events:
`kubectl describe ingress my-ingress`

- Check logs of the ingress controller (e.g., Nginx):
kubectl logs -n ingress-nginx <nginx-controller-pod>
- Test the backend service:
kubectl port-forward svc/my-service 8080:80 curl http://localhost:8080

21. How do you run a pod on a specific node?

Scenario: You need to schedule a pod on a specific node for performance or

compliance reasons.

**Solution: Use
nodeSelector or
nodeAffinity.**

Example using nodeSelector:

```
apiVersion: v1 kind: Pod
metadata: name: specific-node-
pod spec: nodeSelector:
  kubernetes.io/hostname: node-1
containers: - name: my-container
image: nginx
```

**22. How do you use a
Kubernetes Secret in a
pod?**

Scenario: Your application needs to use sensitive information like API keys.

Solution: Store credentials in a Secret and mount it in a pod.

Create Secret:

```
kubectl create secret generic my-secret --from-literal=username=admin --from-literal=password=secret
```

Use Secret in a Pod:

```
env: - name: DB_USER
```

```
valueFrom: secretKeyRef: name:  
my-secret key: username
```

23. How do you prevent unauthorized access to the Kubernetes API?

Scenario: You need to
secure the Kubernetes
API.

Solution:

- Enable RBAC: kubectl auth
can-i create pods --as=user1
- Use RoleBindings for least

```
privilege access: apiVersion:  
rbac.authorization.k8s.io/v1  
kind: Role metadata:  
namespace: default name:  
pod-reader rules: - apiGroups:  
["""] resources: ["pods"] verbs:  
["get", "watch", "list"]
```

24. How do you enable logging for Kubernetes pods?

Scenario: You need to
collect logs for
debugging and
monitoring.

Solution: Use kubectl logs, Fluentd, or ELK.

Check logs of a specific pod:

```
kubectl logs my-pod -f
```

Forward logs to Elasticsearch using Fluentd:

```
apiVersion: v1 kind: ConfigMap  
metadata: name: fluentd-config  
data: fluent.conf: | <match **>  
    @type elasticsearch host  
    elasticsearch-svc </match>
```

25. How do you delete a namespace forcefully?

```
kubectl delete namespace my-  
namespace --grace-period=0 --  
force
```

**26. How do you deploy
an application using
Helm?**

```
helm install my-release bitnami/  
nginx
```

**27. How do you
monitor a Kubernetes
cluster?**

**Solution: Use
Prometheus + Grafana**

```
helm install prometheus  
prometheus-community/kube-  
prometheus-stack
```

28. How do you use Persistent Volume Claims (PVC)?

**Scenario: Your app
needs persistent storage.**

Create PVC:

```
apiVersion: v1 kind:  
PersistentVolumeClaim  
metadata: name: my-pvc spec:  
accessModes: [  
"ReadWriteOnce" ] resources:
```

requests: storage: 1Gi

29. How do you configure service mesh (Istio) in Kubernetes?

```
istioctl install --set profile=demo  
kubectl label namespace default  
istio-injection=enabled
```

30. How do you recover from an accidental kubectl delete?

- If deleted pod belongs to a deployment, it will auto-restart.

- If deleted a namespace, check backups (e.g., Velero).
- If deleted resources manually, check etcd snapshots.

Would you like a deep dive into any of these?