DevOps Shack

50 Common Kubernetes Errors With Troubleshooting & Examples

1. **Pod in CrashLoopBackOff State:**
    - **Error:** The pod keeps crashing and restarting.
    - **Troubleshooting:** Check pod logs for errors, ensure required resources are available, check for misconfigurations.
    - **Example:** `kubectl logs <pod-name>`

2. **ImagePullBackOff:**
    - **Error:** Kubernetes is unable to pull the container image.
    - **Troubleshooting:** Verify image name and access permissions, check network connectivity.
    - **Example:** `kubectl describe pod <pod-name>`

3. **Pod Pending:**
    - **Error:** Pod is stuck in the Pending state.
    - **Troubleshooting:** Insufficient resources, node issues, pod scheduling constraints.
    - **Example:** `kubectl describe pod <pod-name>`

4. **Invalid Pod Specification:**
    - **Error:** Pod spec contains invalid configurations.
    - **Troubleshooting:** Review pod YAML file for syntax errors, missing fields, or incorrect values.
    - **Example:** `kubectl apply -f <pod-spec.yaml> --dry-run=client`

5. **Service Unavailable:**
    - **Error:** Service is not reachable.
    - **Troubleshooting:** Check service configuration, endpoint readiness, network policies.
    - **Example:** `kubectl get svc`

6. **Node Not Ready:**
    - **Error:** Node is not ready to accept pods.
    - **Troubleshooting:** Inspect node status, check system logs, monitor resource usage.
    - **Example:** `kubectl describe node <node-name>`

7. **Volume Mount Errors:**
    - **Error:** Issues with mounting volumes in pods.
    - **Troubleshooting:** Verify volume configuration, permissions, and storage availability.
    - **Example:** `kubectl describe pod <pod-name>`

8. **RBAC Permission Denied:**
    - **Error:** User or service account lacks necessary permissions.
    - **Troubleshooting:** Review RBAC roles and bindings, check cluster role permissions.
    - **Example:** `kubectl auth can-i <verb> <resource> --as <user>`

9. **Pod Evicted:**
    - **Error:** Pod is evicted from the node.
    - **Troubleshooting:** Resource constraints, node issues, pod priority configuration.
    - **Example:** `kubectl describe pod <pod-name>`

10. **Network Policy Issues:**
    - **Error:** Network policies are blocking pod communication.
    - **Troubleshooting:** Review network policy configurations, check pod labels and selectors.
    - **Example:** `kubectl describe networkpolicy <policy-name>`

11. **ImageNotFound:**
    - **Error:** Kubernetes cannot find the specified container image.
    - **Troubleshooting:** Verify image name and repository, check image availability.
    - **Example:** `kubectl describe pod <pod-name>`

12. **Init Container Errors:**
    - **Error:** Issues with init containers failing to start or complete.
    - **Troubleshooting:** Check init container logs, verify dependencies, and container startup order.
    - **Example:** `kubectl logs <pod-name> -c <init-container-name>`

13. **Node Out of Disk Space:**
    - **Error:** Node has insufficient disk space.
    - **Troubleshooting:** Free up disk space, resize volumes, or add additional storage.
    - **Example:** `df -h`

14. **Pod Stuck in Terminating State:**
    - **Error:** Pod is stuck terminating and not being removed.
    - **Troubleshooting:** Manually delete pod finalizers, check controller-manager logs.
    - **Example:** `kubectl delete pod <pod-name> --grace-period=0 –force`

15. **Invalid Namespace:**
    - **Error:** Specified namespace does not exist or is misspelled.
    - **Troubleshooting:** Check namespace spelling, create namespace if necessary.
    - **Example:** `kubectl get namespace`

16. **Invalid Pod IP:**
    - **Error:** Pod IP is not assigned or is invalid.
    - **Troubleshooting:** Check networking configurations, restart kubelet service.
    - **Example:** `kubectl describe pod <pod-name>`

17. **DNS Resolution Failure:**
    - **Error:** Pod cannot resolve DNS names.
    - **Troubleshooting:** Verify DNS configurations, check network policies, test DNS resolution.
    - **Example:** `kubectl exec -it <pod-name> -- nslookup <domain>`

18. **CrashLoopBackOff with Custom Controllers:**
    - **Error:** Custom controller-managed pods are in CrashLoopBackOff state.
    - **Troubleshooting:** Check controller logs, review controller implementation, inspect pod resources.
    - **Example:** `kubectl logs <controller-pod-name>`

19. **ConfigMap Errors:**
    - **Error:** Issues with ConfigMap creation or usage in pods.
    - **Troubleshooting:** Verify ConfigMap configurations, check for syntax errors.
    - **Example:** `kubectl describe configmap <configmap-name>`

20. **Pod Security Context Violation:**
    - **Error:** Pod security context constraints are violated.
    - **Troubleshooting:** Review pod security context, check security policies.
    - **Example:** `kubectl describe pod <pod-name>`

21. **Node NotReady Condition:**
    - **Error:** Node is marked as NotReady.
    - **Troubleshooting:** Check node status, inspect kubelet logs, monitor node health.
    - **Example:** `kubectl describe node <node-name>`

22. **PersistentVolumeClaim Pending:**
    - **Error:** PVC is stuck in Pending state.
    - **Troubleshooting:** Check storage class availability, inspect PV/PVC bindings.
    - **Example:** `kubectl describe pvc <pvc-name>`

23. **Scheduler Errors:**
    - **Error:** Issues with pod scheduling.
    - **Troubleshooting:** Inspect scheduler logs, check resource requests/limits.
    - **Example:** `kubectl logs -n kube-system <scheduler-pod-name>`

24. **Missing Resource Quotas:**
    - **Error:** Resource quota limits are exceeded.
    - **Troubleshooting:** Review resource quotas, adjust resource requests/limits.
    - **Example:** `kubectl describe quota <quota-name>`

25. **Container Terminated Unexpectedly:**

- **Error:** Container inside the pod is terminated unexpectedly.
- **Troubleshooting:** Check container logs, inspect container health checks, review application code.
- **Example:** `kubectl logs <pod-name>`

26. **Secret Decryption Error:**

- **Error:** Unable to decrypt secrets.
- **Troubleshooting:** Verify encryption configurations, check secret permissions.
- **Example:** `kubectl describe secret <secret-name>`

27. **Pod Running Slow:**

- **Error:** Pod is taking longer than expected to start or respond.
- **Troubleshooting:** Check pod resource utilization, inspect application performance.
- **Example:** `kubectl top pod <pod-name>`

28. **Node Crashed:**

- **Error:** Node has crashed and is not recoverable.
- **Troubleshooting:** Diagnosenode hardware/software issues, replace node if necessary.
- **Example:** `kubectl describe node <node-name>`

29. **Deployment Rollout Stuck:**

- **Error:** Deployment rollout is stuck or paused.
- **Troubleshooting:** Inspect deployment status, check for conflicts or blocking conditions.
- **Example:** `kubectl rollout status deployment <deployment-name>`

30. **Ingress Controller Errors:**

    - **Error:** Ingress controller is not routing traffic correctly.
    - **Troubleshooting:** Check ingress controller logs, inspect ingress resources, verify DNS resolution.
    - **Example:** `kubectl logs -n <ingress-controller-namespace> <ingress-controller-pod-name>`

31. **Pod Affinity/Anti-Affinity Failures:**

    - **Error:** Pod scheduling based on affinity/anti-affinity rules fails.
    - **Troubleshooting:** Review pod affinity/anti-affinity configurations, check node labels.
    - **Example:** `kubectl describe pod <pod-name>`

32. **Horizontal Pod Autoscaler (HPA) Not Scaling:**

    - **Error:** HPA is not scaling pods as expected.
    - **Troubleshooting:** Inspect HPA configurations, check resource metrics, review pod utilization.
    - **Example:** `kubectl describe hpa <hpa-name>`

33. **Service Account Permissions:**

    - **Error:** Service account lacks necessary permissions to access resources.
    - **Troubleshooting:** Review service account roles and role bindings.
    - **Example:** `kubectl describe sa <service-account-name>`

34. **Pod Disruption Budget Violation:**

    - **Error:** Pod disruption budget constraints are violated.
    - **Troubleshooting:** Review PodDisruptionBudget configurations, check for pod disruptions.
    - **Example:** `kubectl describe pdb <pdb-name>`

35. **Node Resource Exhaustion:**

- **Error:** Node resources (CPU, memory) are exhausted.
- **Troubleshooting:** Monitor node resource utilization, adjust resource quotas.
- **Example:** `kubectl top node`

36. **Custom Resource Definition (CRD) Errors:**

- **Error:** Issues with custom resource definitions.
- **Troubleshooting:** Check CRD configurations, validate CR manifests.
- **Example:** `kubectl get crd`

37. **Pod Security Policy Violation:**

- **Error:** Pod does not comply with pod security policies.
- **Troubleshooting:** Review pod security policy configurations, check for policy violations.
- **Example:** `kubectl describe pod <pod-name>`

38. **Cluster Autoscaler Not Scaling:**

- **Error:** Cluster autoscaler is not scaling nodes as expected.
- **Troubleshooting:** Inspect cluster autoscaler logs, check node utilization, adjust autoscaler configurations.
- **Example:** `kubectl logs -n kube-system <autoscaler-pod-name>`

39. **Pod Resource Contention:**

- **Error:** Pods on the node are contending for resources.
- **Troubleshooting:** Review resource requests/limits, adjust pod scheduling policies.
- **Example:** `kubectl describe pod <pod-name>`

40. **Endpoint Not Ready:**
    - **Error:** Service endpoint is not ready to receive traffic.
    - **Troubleshooting:** Check service health checks, review endpoint status.
    - **Example:** `kubectl describe endpoints <service-name>`

41. **Namespace Resource Quota Exceeded:**
    - **Error:** Resource quota limits in namespace exceeded.
    - **Troubleshooting:** Adjust resource quotas, monitor namespace resource usage.
    - **Example:** `kubectl describe quota -n <namespace-name>`

42. **Node Drain Failure:**
    - **Error:** Node drain operation fails.
    - **Troubleshooting:** Manually evacuate pods from the node, check for stuck processes.
    - **Example:** `kubectl drain <node-name>`

43. **Invalid Service Type:**
    - **Error:** Service type is invalid or unsupported.
    - **Troubleshooting:** Review service type in service manifest.
    - **Example:** `kubectl describe service <service-name>`

44. **Cluster DNS Resolution Failure:**
    - **Error:** Cluster DNS service is not resolving names correctly.
    - **Troubleshooting:** Verify CoreDNS configurations, check for DNS service availability.
    - **Example:** `kubectl get svc -n kube-system`

45. **Pod Affected by Node Maintenance:**

- **Error:** Pod is affected by node maintenance activities.
- **Troubleshooting:** Evacuate pods from the node, ensure node cordoning.
- **Example:** `kubectl drain <node-name>`

46. **Ingress Resource Misconfiguration:**

- **Error:** Ingress resource is misconfigured.
- **Troubleshooting:** Review ingress YAML file, check backend service configurations.
- **Example:** `kubectl describe ingress <ingress-name>`

47. **API Server Unavailable:**

- **Error:** Kubernetes API server is unreachable.
- **Troubleshooting:** Check API server logs, review network connectivity.
- **Example:** `kubectl cluster-info`

48. **Node Affinity Violation:**

- **Error:** Pod scheduling based on node affinity rules fails.
- **Troubleshooting:** Review pod/node labels, inspect node affinity configurations.
- **Example:** `kubectl describe pod <pod-name>`

49. **Pod Priority Preemption:**

- **Error:** Pods with lower priority are preempted by higher-priority pods.
- **Troubleshooting:** Adjust pod priority settings, review preemption policies.
- **Example:** `kubectl describe pod <pod-name>`

50. **Volume Quota Exceeded:**

- **Error:** Persistent volume quota limits exceeded.
- **Troubleshooting:** Adjust storage quotas, monitor persistent volume usage.
- **Example:** `kubectl describe quota <quota-name>`