100 Kubernetes Errors & Solutions

1. Error: CrashLoopBackOff

- **Solution:**
  - Check the logs of the pod using `kubectl logs <pod-name>`.
  - Identify the cause of the crash (e.g., application bug, misconfiguration).
  - Fix the underlying issue (e.g., correct environment variables, fix code bugs).
  - Restart the pod if necessary using `kubectl delete pod <pod-name>`.

2. Error: ImagePullBackOff

- **Solution:**
  - Ensure the image name is correct in the pod specification.
  - Verify that the image exists in the container registry.
  - Check for network connectivity issues.
  - Use `kubectl describe pod <pod-name>` to get detailed error messages.
  - If using a private registry, ensure proper imagePullSecrets are configured.

3. Error: ErrImagePull

- **Solution:**
  - Check the image name and tag for correctness.
  - Verify access to the container registry.
  - Ensure imagePullSecrets are correctly configured if using a private registry.
  - Look at the events using `kubectl describe pod <pod-name>` for more details.

4. Error: Node Not Ready

- **Solution:**
  - Check the node status using `kubectl get nodes`.
  - Investigate node conditions with `kubectl describe node <node-name>`.
  - Ensure kubelet is running on the node.
  - Check node logs for issues with `journalctl -u kubelet`.
  - Ensure the node has sufficient resources and is not under heavy load.

5. Error: Pod Stuck in Pending State

- **Solution:**
  - Check the pod events using `kubectl describe pod <pod-name>`.
  - Ensure there are sufficient resources (CPU, memory) in the cluster.
  - Verify that the scheduler is running and healthy.
  - Check for node taints that might be preventing scheduling.
  - Ensure the node selector or affinity rules are correctly configured.

6. Error: PersistentVolumeClaim Bound but Pod Cannot Access Volume

- **Solution:**
  - Ensure the PersistentVolume (PV) and PersistentVolumeClaim (PVC) are correctly configured.
  - Check the storage class and provisioner for compatibility.
  - Verify that the PV is in a Bound state.
  - Inspect pod logs for detailed error messages.
  - Ensure the pod has the correct permissions to access the volume.

7. Error: Unauthorized Access to Kubernetes API

- **Solution:**
  - Ensure the API server is accessible and running.
  - Verify that the correct API tokens or certificates are used.
  - Check Role-Based Access Control (RBAC) policies for proper permissions.
  - Use `kubectl auth can-i <verb> <resource>` to test access permissions.
  - Adjust RBAC roles and bindings as necessary.

8. Error: Failed to Create Pod Sandbox

- **Solution:**
  - Check container runtime logs (e.g., Docker, containerd) for errors.

- Verify that the container runtime is running and healthy.
- Ensure the node has sufficient resources (CPU, memory).
- Restart the container runtime service if necessary.
- Check the pod events with `kubectl describe pod <pod-name>` for more details.

9. Error: Service Not Accessible

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the pods backing the service are running and healthy.
  - Check for network policies that might be blocking access.
  - Use `kubectl describe svc <service-name>` to inspect service details.
  - Check endpoints with `kubectl get endpoints <service-name>`.

10. Error: Pod Cannot Mount Secret

- **Solution:**
  - Ensure the secret exists in the same namespace as the pod.
  - Verify the secret name and key in the pod specification.
  - Check the pod events for detailed error messages using `kubectl describe pod <pod-name>`.
  - Ensure the pod has permissions to access the secret.
  - Recreate the secret if it appears to be corrupted.

11. Error: Insufficient CPU/Mem Resources

- **Solution:**
  - Check resource requests and limits in the pod specification.
  - Ensure the cluster has sufficient resources available.
  - Use `kubectl top nodes` and `kubectl top pods` to monitor resource usage.
  - Adjust resource requests and limits as needed.
  - Scale the cluster if necessary.

12. Error: Pods Stuck in Terminating State

- **Solution:**
  - Check the pod events for details using `kubectl describe pod <pod-name>`.
  - Use `kubectl delete pod <pod-name> --grace-period=0 --force` to force delete the pod.
  - Verify the node status and ensure kubelet is running.
  - Investigate potential issues with the underlying container runtime.
  - Check for network or storage issues that might be preventing termination.

13. Error: DNS Resolution Failure

- **Solution:**
  - Verify that the CoreDNS pods are running and healthy using `kubectl get pods -n kube-system`.
  - Check the CoreDNS logs for errors using `kubectl logs <coredns-pod-name> -n kube-system`.
  - Ensure the DNS configuration in the pod is correct.
  - Check for network policies or firewall rules blocking DNS traffic.
  - Restart the CoreDNS pods if necessary.

14. Error: ImagePullSecret Not Working

- **Solution:**
  - Ensure the imagePullSecret is correctly created using `kubectl create secret docker-registry`.
  - Verify the secret is referenced in the pod specification.
  - Check the secret for correct credentials and format.
  - Use `kubectl describe secret <secret-name>` to inspect the secret.
  - Recreate the secret if it appears to be incorrect or corrupted.

15. Error: Invalid Memory/CPU Request

- **Solution:**
  - Ensure resource requests and limits are specified in correct units (e.g., Mi, Gi, m).
  - Check the pod specification for syntax errors.
  - Verify the node has sufficient resources to meet the requests.
  - Adjust the resource requests and limits to realistic values.
  - Validate the configuration using `kubectl apply --dry-run`.

16. Error: PersistentVolume Not Bound

- **Solution:**
  - Ensure the PersistentVolume (PV) and PersistentVolumeClaim (PVC) specifications match.
  - Verify the PV is in an Available state.
  - Check for storage class issues and compatibility.
  - Use `kubectl describe pvc <pvc-name>` to inspect claim details.
  - Recreate the PV and PVC if necessary.

17. Error: Pod Cannot Access ConfigMap

- **Solution:**
  - Ensure the ConfigMap exists in the same namespace as the pod.
  - Verify the ConfigMap name and key in the pod specification.
  - Check the pod events for detailed error messages using `kubectl describe pod <pod-name>`.
  - Ensure the pod has permissions to access the ConfigMap.
  - Recreate the ConfigMap if it appears to be corrupted.

18. Error: Node Disk Pressure

- **Solution:**
  - Check node disk usage using `kubectl describe node <node-name>`.
  - Clean up unused images and containers on the node.
  - Ensure logs and data are not consuming excessive disk space.
  - Add more storage to the node if necessary.
  - Use tools like `du` and `df` to analyze disk usage.

19. Error: Node Memory Pressure

- **Solution:**
  - Monitor node memory usage using `kubectl top nodes`.
  - Check for memory leaks or high memory usage in pods.
  - Adjust pod resource requests and limits.
  - Scale the cluster to add more nodes if necessary.
  - Investigate potential memory-intensive applications.

20. Error: Pod Security Policy Denied

- **Solution:**
  - Ensure the pod security policy (PSP) allows the required permissions.
  - Check the PSP configuration and adjust as necessary.
  - Verify the role bindings and permissions for the service account.
  - Use `kubectl auth can-i` to test permissions.
  - Modify the PSP to allow necessary actions for the pod.

21. Error: Unauthorized Access to Kubelet API

- **Solution:**
  - Verify the correct kubelet API endpoint and credentials.
  - Ensure the client has the necessary permissions.
  - Check the kubelet configuration and logs for errors.
  - Update RBAC roles and bindings to grant access.
  - Secure the kubelet API with proper authentication and authorization.

22. Error: Service IP Conflict

- **Solution:**
  - Check for overlapping IP ranges in service and pod CIDR.
  - Adjust the service IP range in the cluster configuration.
  - Ensure there are no conflicting services with the same IP.
  - Use `kubectl get services` to list all services and their IPs.
  - Reconfigure the network settings if necessary.

23. Error: Port Already in Use

- **Solution:**
  - Ensure the specified port is not already in use by another service.
  - Check the node for conflicting processes using the port.
  - Adjust the service or pod configuration to use a different port.
  - Use tools like `netstat` or `ss`

to check open ports on the node. - Restart the affected service to release the port.

24. Error: PersistentVolume Not Found

- **Solution:**
  - Ensure the PersistentVolume (PV) exists and is in an Available state.
  - Verify the PVC specification and storage class.
  - Check the PV and PVC events for detailed error messages.
  - Use `kubectl describe pv <pv-name>` to inspect the volume.
  - Recreate the PV if necessary.

25. Error: Container Cannot Write to Volume

- **Solution:**
  - Verify the volume mount path and permissions.
  - Ensure the volume is mounted with the correct read/write permissions.
  - Check the container user and group permissions.
  - Use `kubectl exec` to inspect the volume mount inside the container.
  - Adjust the volume permissions as necessary.

26. Error: Node Unschedulable

- **Solution:**
  - Check the node status using `kubectl get nodes`.
  - Use `kubectl describe node <node-name>` to inspect node conditions.
  - Ensure the node is not cordoned or drained.

    - Use `kubectl uncordon <node-name>` to make the node schedulable.
    - Investigate potential resource or configuration issues on the node.

27. Error: Network Policy Blocking Traffic

- **Solution:**
    - Verify the network policy configuration.
    - Ensure the policy allows traffic to and from the necessary pods and services.
    - Use `kubectl describe networkpolicy <policy-name>` to inspect details.
    - Adjust the policy rules to allow required traffic.
    - Test connectivity using tools like `curl` or `ping`.

28. Error: Pod Not Found

- **Solution:**
    - Ensure the pod name and namespace are correct.
    - Use `kubectl get pods` to list all pods in the namespace.
    - Check for typos or incorrect names in the pod specification.
    - Verify the pod has not been deleted or evicted.
    - Recreate the pod if necessary.

29. Error: Readiness Probe Failed

- **Solution:**
    - Check the readiness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is responding correctly at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

30. Error: Liveness Probe Failed

- **Solution:**
    - Check the liveness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is running and healthy at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

31. Error: Node Disk Full

- **Solution:**
    - Check disk usage on the node using `df` and `du` commands.

    - Clean up unused images, containers, and temporary files.
    - Add more storage to the node if necessary.
    - Ensure logs and data are not consuming excessive disk space.
    - Monitor disk usage and set up alerts for high usage.

32. Error: API Server Unreachable

- **Solution:**
    - Check the API server status and logs.
    - Verify network connectivity to the API server endpoint.
    - Ensure the API server process is running.
    - Use `kubectl cluster-info` to get API server details.
    - Restart the API server if necessary.

33. Error: PersistentVolumeClaim Pending

- **Solution:**
    - Check the PersistentVolume (PV) and PersistentVolumeClaim (PVC) specifications.
    - Verify the PVC is correctly bound to a PV.
    - Use `kubectl describe pvc <pvc-name>` to inspect claim details.
    - Ensure the storage class and provisioner are correct.
    - Recreate the PVC if necessary.

34. Error: Pod Evicted

- **Solution:**
    - Check the pod events for eviction details using `kubectl describe pod <pod-name>`.
    - Ensure the node has sufficient resources (CPU, memory).
    - Adjust resource requests and limits for the pod.
    - Investigate potential resource constraints on the node.
    - Recreate the pod on a node with sufficient resources.

35. Error: Service ClusterIP Not Reachable

- **Solution:**
    - Verify the service configuration with `kubectl get svc <service-name>`.
    - Ensure the pods backing the service are running and healthy.
    - Check the endpoints using `kubectl get endpoints <service-name>`.
    - Use `kubectl describe svc <service-name>` to inspect service details.
    - Investigate potential network or DNS issues.

36. Error: Node Out of Disk

- **Solution:**
  - Monitor disk usage on the node using `df` and `du` commands.
  - Clean up unused images, containers, and temporary files.
  - Add more storage to the node if necessary.
  - Ensure logs and data are not consuming excessive disk space.
  - Set up alerts for high disk usage and take preventive measures.

37. Error: Insufficient Permissions

- **Solution:**
  - Check the RBAC roles and bindings for the affected user or service account.
  - Use `kubectl auth can-i <verb> <resource>` to test permissions.
  - Adjust RBAC roles and bindings as necessary.
  - Ensure the user or service account has the required permissions.
  - Reapply the RBAC configuration if necessary.

38. Error: PersistentVolume Already Exists

- **Solution:**
  - Ensure the PersistentVolume (PV) name is unique.
  - Check for existing PVs with the same name using `kubectl get pv`.
  - Use a different name for the new PV.
  - Inspect existing PVs to avoid naming conflicts.
  - Recreate the PV with a unique name if necessary.

39. Error: Node Not Ready

- **Solution:**
  - Check the node status using `kubectl get nodes`.
  - Use `kubectl describe node <node-name>` to inspect node conditions.
  - Ensure the kubelet is running and healthy on the node.
  - Investigate potential resource or configuration issues on the node.
  - Restart the node or kubelet service if necessary.

40. Error: Pod Cannot Communicate with Service

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the pods backing the service are running and healthy.
  - Check the endpoints using `kubectl get endpoints <service-name>`.
  - Use `kubectl describe svc <service-name>` to inspect service details.

    - Investigate potential network or DNS issues.

41. Error: Pod Security Context Denied

- **Solution:**
    - Check the pod security context configuration in the pod specification.
    - Ensure the security context settings are allowed by the policy.
    - Use `kubectl describe pod <pod-name>` to inspect security context details.
    - Adjust the security context settings to comply with the policy.
    - Reapply the pod specification if necessary.

42. Error: Deployment Not Progressing

- **Solution:**
    - Check the deployment status using `kubectl get deployment <deployment-name>`.
    - Use `kubectl describe deployment <deployment-name>` to inspect events and details.
    - Ensure the pods are being created and updated as expected.
    - Investigate potential issues with pod scheduling or readiness.
    - Adjust the deployment strategy or configuration as necessary.

43. Error: Unauthorized Access to Dashboard

- **Solution:**
    - Verify the correct credentials or access tokens for the dashboard.
    - Ensure the user or service account has the necessary permissions.
    - Check the RBAC roles and bindings for dashboard access.
    - Use `kubectl auth can-i` to test permissions for the dashboard user.
    - Adjust the RBAC configuration to grant access if necessary.

44. Error: Pod Failed to Start

- **Solution:**
    - Check the pod events and logs for detailed error messages using `kubectl describe pod <pod-name>` and `kubectl logs <pod-name>`.
    - Ensure the container image is available and correct.
    - Verify the pod specification for any misconfigurations.
    - Investigate potential resource constraints on the node.
    - Recreate the pod if necessary.

45. Error: Service NodePort Not Accessible

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the NodePort is correctly specified and not conflicting with other services.
  - Check for network policies or firewall rules blocking access.
  - Use `kubectl describe svc <service-name>` to inspect service details.
  - Investigate potential node or network issues.

46. Error: Pod Terminated Unexpectedly

- **Solution:**
  - Check the pod events and logs for detailed error messages using `kubectl describe pod <pod-name>` and `kubectl logs <pod-name>`.
  - Ensure the application is not encountering errors or crashes.
  - Investigate potential resource constraints or node issues.
  - Adjust resource requests and limits as necessary.
  - Recreate the pod if necessary.

47. Error: Unauthorized Access to Node Metrics

- **Solution:**
  - Ensure the correct credentials or access tokens for accessing node metrics.
  - Verify the permissions for

the user or service account. - Check the RBAC roles and bindings for node metrics access. - Use `kubectl auth can-i` to test permissions for the metrics user. - Adjust the RBAC configuration to grant access if necessary.

48. Error: PersistentVolume Not Found

- **Solution:**
  - Ensure the PersistentVolume (PV) exists and is in an Available state.
  - Verify the PVC specification and storage class.
  - Check the PV and PVC events for detailed error messages.
  - Use `kubectl describe pv <pv-name>` to inspect the volume.
  - Recreate the PV if necessary.

49. Error: Pod Cannot Access API Server

- **Solution:**
  - Verify network connectivity between the pod and API server.
  - Check the service account and RBAC permissions for the pod.
  - Use `kubectl describe pod <pod-name>` to inspect events and details.

    - Ensure the API server endpoint is correctly specified.
    - Investigate potential network or DNS issues.

50. Error: Pod Not Found

- **Solution:**
    - Ensure the pod name and namespace are correct.
    - Use `kubectl get pods` to list all pods in the namespace.
    - Check for typos or incorrect names in the pod specification.
    - Verify the pod has not been deleted or evicted.
    - Recreate the pod if necessary.

51. Error: Readiness Probe Failed

- **Solution:**
    - Check the readiness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is responding correctly at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

52. Error: Liveness Probe Failed

- **Solution:**
    - Check the liveness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is running and healthy at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

53. Error: Node Disk Full

- **Solution:**
    - Check disk usage on the node using `df` and `du` commands.
    - Clean up unused images, containers, and temporary files.
    - Add more storage to the node if necessary.
    - Ensure logs and data are not consuming excessive disk space.
    - Monitor disk usage and set up alerts for high usage.

54. Error: API Server Unreachable

- **Solution:**
    - Check the API server status and logs.

- Verify network connectivity to the API server endpoint.
- Ensure the API server process is running.
- Use `kubectl cluster-info` to get API server details.
- Restart the API server if necessary.

55. Error: PersistentVolumeClaim Pending

- **Solution:**
  - Check the PersistentVolume (PV) and PersistentVolumeClaim (PVC) specifications.
  - Verify the PVC is correctly bound to a PV.
  - Use `kubectl describe pvc <pvc-name>` to inspect claim details.
  - Ensure the storage class and provisioner are correct.
  - Recreate the PVC if necessary.

56. Error: Pod Evicted

- **Solution:**
  - Check the pod events for eviction details using `kubectl describe pod <pod-name>`.
  - Ensure the node has sufficient resources (CPU, memory).
  - Adjust resource requests and limits for the pod.
  - Investigate potential resource constraints on the node.
  - Recreate the pod on a node with sufficient resources.

57. Error: Service ClusterIP Not Reachable

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the pods backing the service are running and healthy.
  - Check the endpoints using `kubectl get endpoints <service-name>`.
  - Use `kubectl describe svc <service-name>` to inspect service details.
  - Investigate potential network or DNS issues.

58. Error: Node Out of Disk

- **Solution:**
  - Monitor disk usage on the node using `df` and `du` commands.
  - Clean up unused images, containers, and temporary files.
  - Add more storage to the node if necessary.

71. Error: Pod Cannot Access API Server

- **Solution:**
  - Verify network connectivity between the pod and the API server.
  - Check the service account and RBAC permissions for the pod.
  - Use `kubectl describe pod <pod-name>` to inspect events and details.
  - Ensure the API server endpoint is correctly specified.
  - Investigate potential network or DNS issues.

72. Error: Readiness Probe Failed

- **Solution:**
  - Check the readiness probe configuration in the pod specification.
  - Verify the probe endpoint and response criteria.
  - Use `kubectl describe pod <pod-name>` to inspect probe events.
  - Ensure the application is responding correctly at the probe endpoint.
  - Adjust the probe settings or application configuration as needed.

73. Error: Liveness Probe Failed

- **Solution:**
  - Check the liveness probe configuration in the pod specification.
  - Verify the probe endpoint and response criteria.
  - Use `kubectl describe pod <pod-name>` to inspect probe events.
  - Ensure the application is running and healthy at the probe endpoint.
  - Adjust the probe settings or application configuration as needed.

74. Error: Node Disk Full

- **Solution:**
  - Check disk usage on the node using `df` and `du` commands.
  - Clean up unused images, containers, and temporary files.
  - Add more storage to the node if necessary.
  - Ensure logs and data are not consuming excessive disk space.
  - Monitor disk usage and set up alerts for high usage.

75. Error: API Server Unreachable

- **Solution:**
  - Check the API server status and logs.
  - Verify network connectivity to the API server endpoint.
  - Ensure the API server process is running.
  - Use `kubectl cluster-info` to get API server details.
  - Restart the API server if necessary.

76. Error: PersistentVolumeClaim Pending

- **Solution:**
  - Check the PersistentVolume (PV) and PersistentVolumeClaim (PVC) specifications.
  - Verify the PVC is correctly bound to a PV.
  - Use `kubectl describe pvc <pvc-name>` to inspect claim details.
  - Ensure the storage class and provisioner are correct.
  - Recreate the PVC if necessary.

77. Error: Pod Evicted

- **Solution:**
  - Check the pod events for eviction details using `kubectl describe pod <pod-name>`.
  - Ensure the node has sufficient resources (CPU, memory).
  - Adjust resource requests and limits for the pod.
  - Investigate potential resource constraints on the node.
  - Recreate the pod on a node with sufficient resources.

78. Error: Service ClusterIP Not Reachable

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the pods backing the service are running and healthy.
  - Check the endpoints using `kubectl get endpoints <service-name>`.
  - Use `kubectl describe svc <service-name>` to inspect service details.
  - Investigate potential network or DNS issues.

79. Error: Node Out of Disk

- **Solution:**
  - Monitor disk usage on the node using `df` and `du` commands.
  - Clean up unused images, containers, and temporary files.
  - Add more storage to the node if necessary.
  - Ensure logs and data are not consuming excessive disk space.
  - Set up alerts for high disk usage and take preventive measures.

80. Error: Insufficient Permissions

- **Solution:**
  - Check the RBAC roles and bindings for the affected user or service account.
  - Use `kubectl auth can-i <verb> <resource>` to test permissions.

    - Adjust RBAC roles and bindings as necessary.
    - Ensure the user or service account has the required permissions.
    - Reapply the RBAC configuration if necessary.

81. Error: PersistentVolume Already Exists

- **Solution:**
    - Ensure the PersistentVolume (PV) name is unique.
    - Check for existing PVs with the same name using `kubectl get pv`.
    - Use a different name for the new PV.
    - Inspect existing PVs to avoid naming conflicts.
    - Recreate the PV with a unique name if necessary.

82. Error: Node Not Ready

- **Solution:**
    - Check the node status using `kubectl get nodes`.
    - Use `kubectl describe node <node-name>` to inspect node conditions.
    - Ensure the kubelet is running and healthy on the node.
    - Investigate potential resource or configuration issues on the node.
    - Restart the node or kubelet service if necessary.

83. Error: Pod Cannot Communicate with Service

- **Solution:**
    - Verify the service configuration with `kubectl get svc <service-name>`.
    - Ensure the pods backing the service are running and healthy.
    - Check the endpoints using `kubectl get endpoints <service-name>`.
    - Use `kubectl describe svc <service-name>` to inspect service details.
    - Investigate potential network or DNS issues.

84. Error: Pod Security Context Denied

- **Solution:**
    - Check the pod security context configuration in the pod specification.
    - Ensure the security context settings are allowed by the policy.
    - Use `kubectl describe pod <pod-name>` to inspect security context details.
    - Adjust the security context settings to comply with the policy.
    - Reapply the pod specification if necessary.

85. Error: Deployment Not Progressing

- **Solution:**

    - Check the deployment status using `kubectl get deployment <deployment-name>`.
    - Use `kubectl describe deployment <deployment-name>` to inspect events and details.
    - Ensure the pods are being created and updated as expected.
    - Investigate potential issues with pod scheduling or readiness.
    - Adjust the deployment strategy or configuration as necessary.

86. Error: Unauthorized Access to Dashboard

- **Solution:**
    - Verify the correct credentials or access tokens for the dashboard.
    - Ensure the user or service account has the necessary permissions.
    - Check the RBAC roles and bindings for dashboard access.
    - Use `kubectl auth can-i` to test permissions for the dashboard user.
    - Adjust the RBAC configuration to grant access if necessary.

87. Error: Pod Failed to Start

- **Solution:**
    - Check the pod events and logs for detailed error messages using `kubectl describe pod <pod-name>` and `kubectl logs <pod-name>`.
    - Ensure the container image is available and correct.
    - Verify the pod specification for any misconfigurations.
    - Investigate potential resource constraints on the node.
    - Recreate the pod if necessary.

88. Error: Service NodePort Not Accessible

- **Solution:**
    - Verify the service configuration with `kubectl get svc <service-name>`.
    - Ensure the NodePort is correctly specified and not conflicting with other services.
    - Check for network policies or firewall rules blocking access.
    - Use `kubectl describe svc <service-name>` to inspect service details.
    - Investigate potential node or network issues.

89. Error: Pod Terminated Unexpectedly

- **Solution:**
    - Check the pod events and logs for detailed error messages using `kubectl describe pod <pod-name>` and `kubectl logs <pod-name>`.
    - Ensure the application is not encountering errors or crashes.

    - Investigate potential resource constraints or node issues.
    - Adjust resource requests and limits as necessary.
    - Recreate the pod if necessary.

90. Error: Unauthorized Access to Node Metrics

- **Solution:**
    - Ensure the correct credentials or access tokens for accessing node metrics.
    - Verify the permissions for the user or service account.
    - Check the RBAC roles and bindings for node metrics access.
    - Use `kubectl auth can-i` to test permissions for the metrics user.
    - Adjust the RBAC configuration to grant access if necessary.

91. Error: PersistentVolume Not Found

- **Solution:**
    - Ensure the PersistentVolume (PV) exists and is in an Available state.
    - Verify the PVC specification and storage class.
    - Check the PV and PVC events for detailed error messages.
    - Use `kubectl describe pv <pv-name>` to inspect the volume.
    - Recreate the PV if necessary.

92. Error: Pod Cannot Access API Server

- **Solution:**
    - Verify network connectivity between the pod and the API server.
    - Check the service account and RBAC permissions for the pod.
    - Use `kubectl describe pod <pod-name>` to inspect events and details.
    - Ensure the API server endpoint is correctly specified.
    - Investigate potential network or DNS issues.

93. Error: Readiness Probe Failed

- **Solution:**
    - Check the readiness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is responding correctly at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

94. Error: Liveness Probe Failed

- **Solution:**

    - Check the liveness probe configuration in the pod specification.
    - Verify the probe endpoint and response criteria.
    - Use `kubectl describe pod <pod-name>` to inspect probe events.
    - Ensure the application is running and healthy at the probe endpoint.
    - Adjust the probe settings or application configuration as needed.

95. Error: Node Disk Full

- **Solution:**
    - Check disk usage on the node using `df` and `du` commands.
    - Clean up unused images, containers, and temporary files.
    - Add more storage to the node if necessary.
    - Ensure logs and data are not consuming excessive disk space.
    - Monitor disk usage and set up alerts for high usage.

96. Error: API Server Unreachable

- **Solution:**
    - Check the API server status and logs.
    - Verify network connectivity to the API server endpoint.
    - Ensure the API server process is running.
    - Use `kubectl cluster-info` to get API server details.
    - Restart the API server if necessary.

97. Error: PersistentVolumeClaim Pending

- **Solution:**
    - Check the PersistentVolume (PV) and PersistentVolumeClaim (PVC) specifications.
    - Verify the PVC is correctly bound to a PV.
    - Use `kubectl describe pvc <pvc-name>` to inspect claim details.
    - Ensure the storage class and provisioner are correct.
    - Recreate the PVC if necessary.

98. Error: Pod Evicted

- **Solution:**
    - Check the pod events for eviction details using `kubectl describe pod <pod-name>`.
    - Ensure the node has sufficient resources (CPU, memory).
    - Adjust resource requests and limits for the pod.
    - Investigate potential resource constraints on the node.
    - Recreate the pod on a node with sufficient resources.

99. Error: Service ClusterIP Not Reachable

- **Solution:**
  - Verify the service configuration with `kubectl get svc <service-name>`.
  - Ensure the pods backing the service are running and healthy.
  - Check the endpoints using `kubectl get endpoints <service-name>`.
  - Use `kubectl describe svc <service-name>` to inspect service details.
  - Investigate potential network or DNS issues.

100. Error: Node Out of Disk

- **Solution:**
  - Monitor disk usage on the node using `df` and `du` commands.
  - Clean up unused images, containers, and temporary files.
  - Add more storage to the node if necessary.
  - Ensure logs and data are not consuming excessive disk space.
  - Set up alerts for high disk usage and take preventive measures.