

# Day 08: Learning Kubernetes via Troubleshooting: **Worker Node Failure**

## Troubleshooting a Worker Node Failure

### Reasons for Node Failure:

- There are several reasons why a worker node in a Kubernetes cluster might fail, including:
- The node may run out of CPU, memory, or disk space.
- **The kubelet service may crash or become unresponsive.**
- Connectivity problems between the node and the rest of the cluster.
- Physical issues with the server or VM hosting the node.
- Misconfigurations in the node settings or Kubernetes components.

## Steps to Troubleshoot

### Check Node Status

```
controlplane ~ → kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
controlplane        Ready     control-plane   24m   v1.30.0
node01              NotReady  <none>         23m   v1.30.0
```

### Describe the Problematic Node

```
Conditions:
  Type                    Status    LastHeartbeatTime           LastTransitionTime        Reason                    Message
  ----                    -
NetworkUnavailable       False     Tue, 06 Aug 2024 16:41:32 +0000 Tue, 06 Aug 2024 16:41:32 +0000 FlannelIsUp              Flannel is running on this node
MemoryPressure           Unknown   Tue, 06 Aug 2024 16:41:55 +0000 Tue, 06 Aug 2024 16:46:54 +0000 NodeStatusUnknown        Kubelet stopped posting node status.
DiskPressure             Unknown   Tue, 06 Aug 2024 16:41:55 +0000 Tue, 06 Aug 2024 16:46:54 +0000 NodeStatusUnknown        Kubelet stopped posting node status.
PIDPressure              Unknown   Tue, 06 Aug 2024 16:41:55 +0000 Tue, 06 Aug 2024 16:46:54 +0000 NodeStatusUnknown        Kubelet stopped posting node status.
Ready                    Unknown   Tue, 06 Aug 2024 16:41:55 +0000 Tue, 06 Aug 2024 16:46:54 +0000 NodeStatusUnknown        Kubelet stopped posting node status.
Addresses:
```

### Verify Kubelet Status

The kubelet is a critical component that manages pods and communicates with the control plane. Ensure it's running properly.

```
ow: NodeNotReady

controlplane ~ → ssh node01

node01 ~ → service kubelet status
kubelet.service - kubelet: The Kubernetes Node Agent
Loaded: loaded (/lib/systemd/system/kubelet.service; enabled; vendor preset: enabled)
Drop-In: /usr/lib/systemd/system/kubelet.service.d
└─10-kubeadm.conf
Active: inactive (dead) since Tue 2024-08-06 16:46:13 UTC; 29min ago
Docs: https://kubernetes.io/docs/
Process: 2553 ExecStart=/usr/bin/kubelet $KUBELET_KUBECONFIG_ARGS $KUBELET_CONFIG_ARGS $K
Main PID: 2553 (code=exited, status=0/SUCCESS)
```

Here, we discovered that the kubelet is not working; it is in a stopped state. To resolve this, we need to start the kubelet service. After starting the kubelet, we should exit the node and verify the node status using `kubectl get nodes`. This will confirm if the node has returned to a Ready state.

```
node01 ~ → service kubelet status
• kubelet.service - kubelet: The Kubernetes Node Agent
  Loaded: loaded (/lib/systemd/system/kubelet.service; enabled; vendor preset: enabled)
  Drop-In: /usr/lib/systemd/system/kubelet.service.d
           └─10-kubeadm.conf
  Active: active (running) since Tue 2024-08-06 17:17:33 UTC; 4s ago
  Docs: https://kubernetes.io/docs/
  Main PID: 13584 (kubelet)
  Tasks: 27 (limit: 251379)
  Memory: 40.2M
  CGroup: /system.slice/kubelet.service
           └─13584 /usr/bin/kubelet --bootstrap-kubeconfig=/etc/kubernetes/bootstrap-kubele>

Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.319992 13584 apiserver.go:52] "Watchi>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.323432 13584 topology_manager.go:215]>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.323550 13584 topology_manager.go:215]>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.337036 13584 desired_state_of_world_p>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362592 13584 reconciler_common.go:247>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362650 13584 reconciler_common.go:247>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362675 13584 reconciler_common.go:247>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362833 13584 reconciler_common.go:247>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362862 13584 reconciler_common.go:247>
Aug 06 17:17:34 node01 kubelet[13584]: I0806 17:17:34.362935 13584 reconciler_common.go:247>
lines 1-22/22 (END)
```

Yes, it is finally started. Exit from the node and run `kubectl get nodes`.

```
NAME          STATUS    ROLES          AGE    VERSION
controlplane  Ready    control-plane  38m    v1.30.0
node01        Ready    <none>         37m    v1.30.0

controlplane ~ →
```

Our node is in ready state now.

## Conclusion:

By following these troubleshooting steps, you can identify and resolve common issues that cause worker nodes to fail in a Kubernetes cluster. Restarting the kubelet service often resolves the issue, allowing the node and its pods to return to a healthy state. Always ensure that you monitor your nodes regularly and have alerting mechanisms in place to detect and address problems promptly.

**Congratulations,** you've finished one more troubleshooting! 🎉 Thanks for following along with my content. I hope you found it helpful and informative. If you have any more questions or need further assistance, feel free to reach out.

Thanks for watching!



**-Shubham Tiwari**