



KUBERNETES ARCHITECTURE

Kubernetes Architecture

Node

A node is an API object created outside the cluster representing an instance. While a cp must be Linux, worker nodes can also be Microsoft Windows Server 2019. Once the node has the necessary software installed, it is ingested into the API server.

At the moment, you can create a cp node with the **kubeadm init** command and worker nodes by passing **join**. In the near future, secondary cp nodes and/or etcd nodes may be joined.

If the kube-apiserver cannot communicate with the kubelet on a node for 5 minutes, the default **NodeLease** will schedule the node for deletion and the **NodeStatus** will change from *ready*. The pods will be evicted once a connection is re-established. They are no longer forcibly removed and rescheduled by the cluster.

Each node object exists in the **kube-node-lease** namespace. To remove a node from the cluster, first use **kubectl delete node <node-name>** to remove it from the API server. This will cause pods to be evacuated. Then, use **kubeadm reset** to remove cluster-specific information. You may also need to remove iptables information, depending on if you plan on re-using the node.

To view CPU, memory, and other resource usage, requests and limits use the **kubectl describe node** command. The output will show capacity and pods allowed, as well as details on current pods and resource utilization.