



kubernetes

Static pods are a type of pod that is managed directly by the kubelet daemon on a specific node, without the API server observing them. This means that static pods are not subject to the same restrictions as normal pods, such as resource quotas and node affinity.

Static pods are typically used for bootstrapping Kubernetes itself, or for running applications that need to be guaranteed to run on a specific node.

Here are some of the uses of static pods:

- **Running control plane components:** Static pods are often used to run the control plane components of Kubernetes, such as the API server, controller manager, and scheduler. This is because these components need to be guaranteed to run on a specific node, and they are not subject to the same restrictions as normal pods.
- **Running applications that need to be isolated:** Static pods can also be used to run applications that need to be isolated from other pods on the cluster. This is because static pods are not subject to the same resource quotas and node affinity rules as normal pods.
- **Running applications that need to be highly available:** Static pods can also be used to run applications that need to be highly available. This is because static pods can be replicated across multiple nodes, ensuring that the application will always be available.

Here are some of the limitations of static pods:

- **Static pods are not managed by the API server:** This means that static pods cannot be controlled or monitored using the API server.
- **Static pods are not subject to the same restrictions as normal pods:** This can make it difficult to ensure that static pods are properly isolated from other pods on the cluster.
- **Static pods can be difficult to manage:** Static pods are not as easy to manage as normal pods. This is because they are not managed by the API server, and they can be difficult to track and troubleshoot.

- Check which pods already running in cluster
- `kubectl get pods --all-namespaces`

```
vagrant@k8s-master:~/taint-toleratation$ kubectl get pods --all-namespaces
NAMESPACE      NAME                                                    READY   STATUS    RESTARTS   AGE
calico-apiserver  calico-apiserver-7b4c75f5b7-7mrrc                    1/1     Running   7 (122m ago)  5d18h
calico-apiserver  calico-apiserver-7b4c75f5b7-wtkx5                    1/1     Running   7 (122m ago)  5d18h
calico-system     calico-kube-controllers-6dfbf88686-bjq65              1/1     Running   7 (122m ago)  5d18h
calico-system     calico-node-46fw8                                      0/1     Running   6 (121m ago)  5d18h
calico-system     calico-node-65kfl                                      0/1     Running   7 (122m ago)  5d18h
calico-system     calico-typha-554b9f897b-r2ds7                        1/1     Running   13 (121m ago)  5d18h
calico-system     csi-node-driver-98mgb                                 2/2     Running   12 (122m ago)  5d18h
calico-system     csi-node-driver-jlpgm                                 2/2     Running   8 (121m ago)  5d18h
default          testpod                                                1/1     Running   0             54m
kube-system       coredns-5d78c9869d-428cp                             1/1     Running   7 (122m ago)  5d18h
kube-system       coredns-5d78c9869d-hfbl8                             1/1     Running   7 (122m ago)  5d18h
kube-system       etcd-k8s-master                                       1/1     Running   7 (122m ago)  5d18h
kube-system       kube-apiserver-k8s-master                             1/1     Running   7 (122m ago)  5d18h
kube-system       kube-controller-manager-k8s-master                   1/1     Running   7 (122m ago)  5d18h
kube-system       kube-proxy-n7mmv                                      1/1     Running   7 (121m ago)  5d18h
kube-system       kube-proxy-qqqpq                                      1/1     Running   7 (122m ago)  5d18h
kube-system       kube-scheduler-k8s-master                             1/1     Running   7 (122m ago)  5d18h
tigera-operator   tigera-operator-7bf7458-s8bnq                         1/1     Running   12 (121m ago)  5d18h
vagrant@k8s-master:~/taint-toleratation$
```

- how to get to know which one is static pod or not ?

- Usually kubelet create a pod ,and it the end of the pod name there is node name as seen as above.
- Where can I find the location of the folder where manifest files are stored for static pods?
 - The kubelet usually picks up the files and runs the pod using whatever files are inside that folder.
- Check kubelet config files
 - `sudo cat /var/lib/kubelet/config.yaml``sudo cat /var/lib/kubelet/config.yaml`

```
nodeStatusUpdateFrequency: 0s
resolvConf: /run/systemd/resolve/resolv.conf
rotateCertificates: true
runtimeRequestTimeout: 0s
shutdownGracePeriod: 0s
shutdownGracePeriodCriticalPods: 0s
staticPodPath: /etc/kubernetes/manifests
streamingConnectionIdleTimeout: 0s
syncFrequency: 0s
volumeStatsAggPeriod: 0s
vagrant@k8s-master:~/taint-toleratation$
```

- You will get the static pod path file
- Copy the path and cd into it

```
cd /var/lib/kubelet/
```

- vi static-pod.yml

```
apiVersion: v1
kind: Pod
metadata:
  name: testpod
spec:
  containers:
    - name: vishal-nginx
      image: nginx:alpine
      ports:
        - containerPort: 80
```

- save, exit and apply

- kubectl apply -f static-pod.yml

```
vagrant@k8s-master:/etc/kubernetes/manifests$ kubectl apply -f static-pod.yml
Warning: resource pods/testpod is missing the kubectrl.kubernetes.io/last-applied-configuration annotation. kubectl apply should only be used on resources created declaratively by either kubectl apply. The missing annotation will be patched automatically.
pod/testpod configured
```

- Check pods:
- kubectl get pods --all-namespaces

```
vagrant@k8s-master:/etc/kubernetes/manifests$ kubectl get pods --all-namespaces
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
calico-apiserver	calico-apiserver-7b4c75f5b7-7mrrc	1/1	Running	7 (141m ago)	5d18h
calico-apiserver	calico-apiserver-7b4c75f5b7-wtkx5	1/1	Running	7 (141m ago)	5d18h
calico-system	calico-kube-controllers-6dfbf88686-bjq65	1/1	Running	7 (141m ago)	5d18h
calico-system	calico-node-46fw8	0/1	Running	6 (141m ago)	5d18h
calico-system	calico-node-65kfl	0/1	Running	7 (141m ago)	5d18h
calico-system	calico-typha-554b9f897b-r2ds7	1/1	Running	13 (140m ago)	5d18h
calico-system	csi-node-driver-98mgb	2/2	Running	12 (141m ago)	5d18h
calico-system	csi-node-driver-jlpgm	2/2	Running	8 (141m ago)	5d18h
default	testpod	1/1	Running	0	74m
default	testpod-k8s-master	1/1	Running	0	3m9s
kube-system	coredns-5d78c9869d-428cp	1/1	Running	7 (141m ago)	5d18h
kube-system	coredns-5d78c9869d-hfbl8	1/1	Running	7 (141m ago)	5d18h
kube-system	etcd-k8s-master	1/1	Running	7 (141m ago)	5d18h
kube-system	kube-apiserver-k8s-master	1/1	Running	7 (141m ago)	5d18h
kube-system	kube-controller-manager-k8s-master	1/1	Running	7 (141m ago)	5d18h
kube-system	kube-proxy-n7mmv	1/1	Running	7 (141m ago)	5d18h
kube-system	kube-proxy-qgqpq	1/1	Running	7 (141m ago)	5d18h
kube-system	kube-scheduler-k8s-master	1/1	Running	7 (141m ago)	5d18h
tigera-operator	tigera-operator-7bf7458-s8bnq	1/1	Running	12 (140m ago)	5d18h

```
vagrant@k8s-master:/etc/kubernetes/manifests$
```

- As you can see, after we have created static pod , at the end of the pod , node name appended automatically.

My devops repo :

- <https://github.com/vishalk17/devops>

My telegram channel:

-  https://t.me/vishalk17_devops

Contact:

Telegram :  t.me/vishalk17

vishalk17 My youtube Channel :

-  **YouTube** <https://www.youtube.com/@vishalk17>