



In Kubernetes, cordon and drain are two commands that are used to prepare a node for maintenance.

- Cordon marks a node as unschedulable, which means that no new pods will be scheduled on the node. This is done to prevent new pods from being created on the node while it is being maintained.
- **Drain** evicts all pods from a node, including pods that are running and pods that are in a pending state. This is done to ensure that the node is completely empty before it is taken offline for maintenance.

The cordon and drain commands are often used together to prepare a node for maintenance. First, the node is cordoned to prevent new pods from being scheduled on it. Then, the node is drained to evict all existing pods. Once the node is drained, it is safe to perform maintenance on it.

Here is an example of how to use the cordon and drain commands together:

kubectl cordon node-1
kubectl drain node-1

This will cordon the node named node-1 and then drain it. All pods on the node will be evicted, and the node will be marked as unschedulable. Once the drain command is complete, the node is safe to perform maintenance on.

It is important to note that the cordon and drain commands can have a significant impact on the availability of your applications. If you are cordoning and draining a node that is hosting critical applications, you should make sure that you have a plan to ensure that those applications are still available while the node is being maintained.

Here are some additional things to keep in mind when using the cordon and drain commands:

- The cordon and drain commands can be used on any node in your cluster, regardless of whether the node is running critical applications.
- The cordon and drain commands can be used to prepare a node for any type of maintenance, including software updates, hardware upgrades, and troubleshooting.
- The cordon and drain commands can be used to prepare a node for removal from the cluster





Q. I want to evaluate the drain command in MicroK8s. How will it behave? Will it create the pod on another node when it drains it from the first one?

→ we have two nodes:

```
vagrant@k8s-master:~$ kubectl get nodes
NAME STATUS ROLES AGE VERSION
k8s-master Ready <none> 65m v1.27.4
k8s-worker-1 Ready <none> 58m v1.27.4
vagrant@k8s-master:~$
```

- Check no. of pods we have in current namespace

```
vagrant@k8s-master:~$ kubectl get nodes
              STATUS
                       ROLES
NAME
                                AGE VERSION
k8s-master
              Ready
                                65m
                                      v1.27.4
k8s-worker-1 Ready
                                58m
                                     v1.27.4
vagrant@k8s-master:~$ ls
httpd.yml snap
vagrant@k8s-master:~$ kubectl get pods
No resources found in default namespace.
vagrant@k8s-master:~$
```

- Crete pods with 4 replica
  - vi httpd.yml

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: apache-deployment
 labels:
  app: apache
spec:
 replicas: 4
 selector:
  matchLabels:
    app: apache
 template:
   metadata:
     labels:
      app: apache
     containers:
      - name: apache
       image: httpd:latest
       ports:
        - containerPort: 80
```

- Save, apply

```
vagrant@k8s-master:~$ kubectl apply -f httpd.yml
deployment.apps/apache-deployment created
vagrant@k8s-master:~$
```





Check where pod has been scheduled

```
vagrant@k8s-master:~$ kubectl get pods -o wide
NAME
                                         READY
                                                  STATUS
                                                            RESTARTS
                                                                         AGE
                                                                                ΙP
                                                                                                NODE
                                                                               10.1.235.197
10.1.230.10
10.1.230.11
                                        1/1
1/1
                                                                                                k8s-master
apache-deployment-6696dd9569-wssm2
                                                                         94s
                                                 Running
                                                            Θ
apache-deployment-6696dd9569-xzd9r
                                                  Running
                                                             Θ
                                                                         94s
                                                                                                k8s-worker-1
apache-deployment-6696dd9569-zbvf7
                                         1/1
                                                  Running
                                                            0
                                                                         94s
                                                                                                k8s-worker-1
apache-deployment-6696dd9569-7gjlb
                                         1/1
                                                            Θ
                                                                         94s
                                                                                                k8s-master
                                                 Running
vagrant@k8s-master:~$
```

- 2 scheduled on worker node and 2 on master node
- Q. what if I apply cordon on worker node? Let see
- Cordon marks a node as unschedulable, which means that no new pods will be scheduled on the node. This is done to prevent new pods from being created on the node while it is being maintained.

#### kubectl cordon k8s-worker-1

```
vagrant@k8s-master:~$
vagrant@k8s-master:~$ kubectl get nodes
               STATUS
                        ROLES
                                  AGE
NAME
                                        VERSION
                                  75m
k8s-master
               Ready
                                        v1.27.4
k8s-worker-1
               Ready
                                  68m
                                        v1.27.4
vagrant@k8s-master:~$ kubectl cordon k8s-worker-1
node/k8s-worker-1 cordoned
vagrant@k8s-master:~$ kubectl get nodes
NAME
               STATUS
                                           ROLES
                                                    AGE
                                                          VERSION
k8s-master
               Ready
                                                    75m
                                                          v1.27.4
               Ready, SchedulingDisabled
k8s-worker-1
                                           <none>
                                                    68m
                                                          v1.27.4
vagrant@k8s-master:~$ ■
```

```
🔼 13. vagrant master
                                                                                                               k8
Every 2.0s: microk8s kubectl get pods -o wide
                                         READY
                                                  STATUS
                                                             RESTARTS
                                                                         AGE
                                                                                  ΙP
                                                                                                   NODE
apache-deployment-6696dd9569-wssm2
                                         1/1
1/1
1/1
                                                  Running
                                                                         8m35s
                                                                                                   k8s-master
                                                                                  10.1.230.10
10.1.230.11
                                                  Running
apache-deployment-6696dd9569-xzd9r
                                                                         8m35s
                                                                                                   k8s-worker-1
apache-deployment-6696dd9569-zbvf7
                                                  Running
                                                                         8m35s
                                                                                                   k8s-worker-1
apache-deployment-6696dd9569-7gjlb
                                                             Θ
                                                                         8m35s
                                                                                                   k8s-master
                                                  Running
```

```
🚺 13. vagrant master
                                                                                                              k8
Every 2.0s: microk8s kubectl get pods -o wide
                                        READY
                                                 STATUS
                                                            RESTARTS
                                                                         AGE
                                                                                 ΙP
                                                                                                  NODE
                                        1/1
1/1
apache-deployment-6696dd9569-wssm2
                                                 Running
                                                                         8m35s
                                                                                                  k8s-master
                                                                                 10.1.230.10
10.1.230.11
apache-deployment-6696dd9569-xzd9r
                                                 Running
                                                                        8m35s
                                                                                                  k8s-worker-1
apache-deployment-6696dd9569-zbvf7
                                        1/1
                                                                                                  k8s-worker-1
                                                 Running
                                                            Θ
                                                                        8m35s
                                        1/1
apache-deployment-6696dd9569-7gjlb
                                                 Running
                                                            Θ
                                                                        8m35s
                                                                                                  k8s-master
```





Increase total no of replication to 6 previously was 4

Edit httpd.yml

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: apache-deployment
 labels:
   app: apache
spec:
  replicas: 6
 selector:
   matchLabels:
    app: apache
 template:
   metadata:
     labels:
      app: apache
     containers:
      - name: apache
       image: httpd:latest
       ports:
        - containerPort: 80
```

- save , apply

```
vagrant@k8s-master:~$ vi httpd.yml
vagrant@k8s-master:~$ kubectl apply -f httpd.yml
deployment.apps/apache-deployment configured
vagrant@k8s-master:~$ ■
```

```
Every 2.0s: microk8s kubectl get pods -o wide
                                             READY
                                                       STATUS
                                                                   RESTARTS
                                                                                 AGE
                                                                                        ΙP
                                                                                                           NODE
                                             1/1
1/1
1/1
1/1
1/1
1/1
                                                                                      10.1.235.197
10.1.230.10
10.1.230.11
apache-deployment-6696dd9569-wssm2
                                                       Running
                                                                                 14m
                                                                                                           k8s-master
                                                                                 14m
apache-deployment-6696dd9569-xzd9r
                                                       Running
                                                                                                           k8s-worker-1
                                                       Running
apache-deployment-6696dd9569-zbvf7
                                                                                                           k8s-worker-1
                                                                   Θ
                                                                                 14m
                                                                                       10.1.235.198
10.1.235.199
apache-deployment-6696dd9569-7gjlb
apache-deployment-6696dd9569-98f7k
                                                                                 14m
                                                                                                           k8s-master
                                                       Running
                                                       Running
                                                                                                           k8s-master
apache-deployment-6696dd9569-6qmlc
                                                       Running
                                                                                 15s
                                                                                                           k8s-master
```

- Noticeable changes are,
  - Scheduling got disabled on worker node thus new pod got scheduled on master node
  - No noticeable changed in existing pods behaviour





Q. I want to evaluate drain in microk8s. how it will behave. Will it create the pod on another node, when it drains it from first one?

• **Drain** evicts all pods from a node, including pods that are running and pods that are in a pending state. This is done to ensure that the node is completely empty before it is taken offline for maintenance.

Lets check current state of pod scheduled,

```
🗶 🖊 🔼 13. vagrant master
                                                                               (+)
/ 👔 🔾 🔼 11. vagrant master
Every 2.0s: microk8s kubectl get pods -o wide
                                                                                                                         k8s
                                            READY
                                                                  RESTARTS
                                                      STATUS
                                                                               AGE
                                                                                                           NODE
                                                                                         10.1.235.197
10.1.230.10
10.1.230.11
10.1.235.198
10.1.235.199
10.1.235.200
                                            1/1
1/1
1/1
1/1
apache-deployment-6696dd9569-wssm2
                                                      Running
                                                                               17m
                                                                                                           k8s-master
apache-deployment-6696dd9569-xzd9r
                                                                                17m
                                                                                                           k8s-worker-1
                                                      Running
apache-deployment-6696dd9569-zbvf7
                                                      Running
                                                                               17m
                                                                                                           k8s-worker-1
apache-deployment-6696dd9569-7gjlb
                                                                               17m
                                                      Running
                                                                  Θ
                                                                                                           k8s-master
apache-deployment-6696dd9569-98f7k
                                             1/1
                                                      Running
                                                                  Θ
                                                                               3m23s
                                                                                                           k8s-master
apache-deployment-6696dd9569-6qmlc
                                                      Running
                                                                               3m23s
                                                                                                           k8s-master
```

Lets apply drain on worker node,

kubectl drain k8s-worker-1

```
vagrant@k8s-master:~$ kubectl get nodes

NAME
STATUS
ROLES
ROLES
AGE VERSION

k8s-worker-1 Ready, SchedulingDisabled <none> 82m v1.27.4

k8s-master Ready
vagrant@k8s-master:~$ kubectl drain k8s-worker-1
node;k8s-worker-1 already cordoned
error: unable to drain node "k8s-worker-1" due to error:cannot delete DaemonSet-managed Pods (use --ignore-daemonsets to ignore): kube-s
ystem/calico-node-6bs8t, continuing command...
There are pending nodes to be drained:
k8s-worker-1
cannot delete DaemonSet-managed Pods (use --ignore-daemonsets to ignore): kube-system/calico-node-6bs8t
vagrant@k8s-master:~$
vagrant@k8s-master:~$
vagrant@k8s-master:~$
vagrant@k8s-master:~$
vagrant@kss-master:~$
vagrant@kss-master:~$
vagrant@kss-master:~$
vagrant@ks-master:~$
vagrant@ks-
```

```
Every 2.0s: microk8s kubectl get pods -o wide
                                                                                                                        k8s-master
NAME
                                            READY
                                                      STATUS
                                                                              RESTARTS
                                                                                                     ΤP
                                                                                                                       NODE
                                                                                           AGE
SS GATES
apache-deployment-6696dd9569-wssm2
                                                                                           21m
                                                                                                                       k8s-master
                                                      Running
apache-deployment-6696dd9569-7gjlb
apache-deployment-6696dd9569-98f7k
                                            1/1
1/1
1/1
                                                                                                     10.1.235.198
10.1.235.199
10.1.235.200
                                                                                           21m
                                                                                                                       k8s-master
                                                      Running
                                                                                           7m19s
                                                      Running
                                                                                                                       k8s-master
apache-deployment-6696dd9569-6qmlc
                                                      Running
                                                                              Θ
                                                                                           7m19s
                                                                                                                       k8s-master
apache-deployment-6696dd9569-29lvt
                                                      ContainerCreating
                                            θ/1
1/1
                                                                              Θ
                                                                                                                       k8s-master
                                                                                           6s
apache-deployment-6696dd9569-xgdcq
                                                      Running
                                                                                           6s
                                                                                                                       k8s-master
```

Noticeable changes are as follows:

• Existing pod evicted from worker node and scheduled on or created on another available node, here it is master node.

### **How to remove Cordon:**

Lets check on which node we have applied cordon:

```
vagrant@k8s-master:~$ kubectl get nodes
NAME
               STATUS
                                          ROLES
                                                    AGE
                                                          VERSION
k8s-master
               Ready
                                           <none>
                                                    95m
                                                          v1.27.4
               Ready, SchedulingDisabled
k8s-worker-1
                                                    88m
                                                          v1.27.4
vagrant@k8s-master:~$
vagrant@k8s-master:~$
```

- Remove cordon from worker node

#### kubectl uncordon k8s-worker-1

```
vagrant@k8s-master:~$ kubectl get nodes
NAME
               STATUS
                                           ROLES
                                                    AGE
                                                          VERSION
               Ready
k8s-master
                                                    96m
                                                          v1.27.4
k8s-worker-1
               Ready, Scheduling Disabled
                                                    89m
                                                          v1.27.4
vagrant@k8s-master:~$
vagrant@k8s-master:~$
vagrant@k8s-master:~$ kubectl get nodes
NAME
               STATUS
                                           R0LES
                                                    AGE
                                                          VERSION
k8s-master
               Ready
                                                    96m
                                                          v1.27.4
               Ready, Scheduling Disabled
k8s-worker-1
                                                    89m
                                                          v1.27.4
vagrant@k8s-master:~$
vagrant@k8s-master:~$ kubectl uncordon k8s-worker-1
node/k8s-worker-1 uncordoned
vagrant@k8s-master:~$
vagrant@k8s-master:~$ kubectl get nodes
NAME
               STATUS
                        ROLES
                                  AGE
                                        VERSION
k8s-master
               Ready
                        <none>
                                  96m
                                        v1.27.4
k8s-worker-1
               Ready
                        <none>
                                  89m
                                        v1.27.4
vagrant@k8s-master:~$
```



## My devops repo:

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

## My telegram channel:



### Contact:



# vishalk17 My youtube Channel:

- YouTube https://www.youtube.com/@vishalk17

### Ref:

# **DevOps Pro**

- YouTube https://youtu.be/EKDmz49BhX8