Kubernetes Operations (Kops):

Understanding Labels, Selectors, and Node Selectors

Managing Kubernetes clusters effectively requires robust tools and techniques to orchestrate workloads. Kops (Kubernetes Operations),

a powerful tool for managing Kubernetes clusters, leverages labels, selectors, and node selectors to simplify resource management and scheduling.

STEP-1: LAUNCH INSTANCE WITH T2.MEDIUM AND 30 GB SSD

STEP-2: INSTALL AWS CLI

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"

unzip awscliv2.zip

sudo ./aws/install

TO CHECK VERSION: /usr/local/bin/aws --version

TO SET PATH: vim .bashrc

export PATH=$PATH:/usr/local/bin/

source .bashrc

aws --version

STEP-3: INSTALL KOPS & KUBECTL

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

wget https://github.com/kubernetes/kops/releases/download/v1.24.1/kops-linux-amd64

PERMISSIONS: chmod +x kops-linux-amd64 kubectl

MOVE FILES: mv kubectl /usr/local/bin/kubectl

MOVE FILES: mv kops-linux-amd64 /usr/local/bin/kops

TO SEE VERSION: kubectl version & kops version

STEP-4: CREATE IAM USER WITH ADMIN PERMISSIONS AND CONFIGURE IT IN ANY REGION WITH TABLE FORMAT

STEP-5: CREATE INFRA SETUP

TO CREATE BUCKET: aws s3api create-bucket --bucket musta.k8s.local --region us-east-1

TO ENABLE VERSION: aws s3api put-bucket-versioning --bucket musta.k8s.local --region us-east-1 --versioning-configuration Status=Enabled

EXPORT CLUSTER DATA INTO BUCKET: export KOPS_STATE_STORE=s3://musta.k8s.local

GENERATE-KEY: ssh-keygen

TO CREATE CLUSTER: kops create cluster --name musta.k8s.local --zones us-east-1a --master-size t2.medium --node-size t2.micro

TO SEE THE CLUSTER: kops get cluster

IF YOU WANT TO EDIT THE CLUSTER: kops edit cluster cluster_name

TO RUN THE CLUSTER: kops update cluster --name musta.k8s.local --yes –admin

```
Suggestions:
 * list clusters with: kops get cluster
 * edit this cluster with: kops edit cluster jaggu.k8s.local
 * edit your node instance group: kops edit ig --name=jaggu.k8s.local nodes-us-east-1a
 * edit your master instance group: kops edit ig --name=jaggu.k8s.local master-us-east-1a

Finally configure your cluster with: kops update cluster --name jaggu.k8s.local --yes --admin

[root@ip-172-31-95-6 ~]# ^C
[root@ip-172-31-95-6 ~]# kops update cluster --name jaggu.k8s.local --yes --admin

*********************************************************************************

A new kops version is available: 1.28.4
Upgrading is recommended
More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade_kops.md#1.28.4

*********************************************************************************
```

```
[root@ip-172-31-95-6 ~]# aws s3 ls
2024-11-27 09:32:42 jaggu.k8s
2024-12-05 11:39:55 sample.flm.k8s
[root@ip-172-31-95-6 ~]# aws s3 mb s3://devops.k8s
make_bucket: devops.k8s
[root@ip-172-31-95-6 ~]# aws s3 ls
2024-12-06 10:03:07 devops.k8s
2024-11-27 09:32:42 jaggu.k8s
2024-12-05 11:39:55 sample.flm.k8s
[root@ip-172-31-95-6 ~]# export KOPS_STATE_STORE=s3://devops.k8s
[root@ip-172-31-95-6 ~]# kops create cluster --name jaggu.k8s.local --zones us-east-1a,us-east-1d --master-size t2.medium --master-count 1 --node-size t2
.micro --node-count 2
I1206 10:15:46.918382    3907 new_cluster.go:251] Inferred "aws" cloud provider from zone "us-east-1a"
I1206 10:15:46.918536    3907 new_cluster.go:1168]  Cloud Provider ID = aws
I1206 10:15:46.980303    3907 subnets.go:185] Assigned CIDR 172.20.32.0/19 to subnet us-east-1a
I1206 10:15:46.980402    3907 subnets.go:185] Assigned CIDR 172.20.64.0/19 to subnet us-east-1d
Previewing changes that will be made:


*********************************************************************************

A new kops version is available: 1.28.4
Upgrading is recommended
More information: https://github.com/kubernetes/kops/blob/master/permalinks/upgrade_kops.md#1.28.4
```

```
[root@ip-172-31-95-6 ~]# kubectl get nodes
NAME                  STATUS   ROLES           AGE    VERSION
i-06a65ccd609c0b7a2   Ready    node            56s    v1.24.17
i-0dc46952949253bd7   Ready    node            57s    v1.24.17
i-0e3939c7168aa8da9   Ready    control-plane   2m1s   v1.24.17
[root@ip-172-31-95-6 ~]# mkdir mainfest
[root@ip-172-31-95-6 ~]# cd mainfest
[root@ip-172-31-95-6 mainfest]# vim pod.yml
[root@ip-172-31-95-6 mainfest]#
```

```
[root@ip-172-31-95-6 mainfest]# kubectl get nodes
NAME                  STATUS   ROLES           AGE   VERSION
i-06a65ccd609c0b7a2   Ready    node            10m   v1.24.17
i-0dc46952949253bd7   Ready    node            10m   v1.24.17
i-0e3939c7168aa8da9   Ready    control-plane   11m   v1.24.17
[root@ip-172-31-95-6 mainfest]# vim pod.yml
[root@ip-172-31-95-6 mainfest]# kubectl create -f pod.yml
pod/mypod created
[root@ip-172-31-95-6 mainfest]# kubectl get po
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          23s
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME    READY   STATUS    RESTARTS   AGE    LABELS
mypod   1/1     Running   0          110s   app=swiggy
[root@ip-172-31-95-6 mainfest]# vim pod.yml
[root@ip-172-31-95-6 mainfest]# kubectl create -f pod.yml
pod/mypod1 created
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          3m34s   app=swiggy
mypod1   1/1     Running   0          17s     app=zomato
[root@ip-172-31-95-6 mainfest]#
```

vim pod.yml (Creating mypod1)

```
---
apiVersion: v1
kind: Pod
metadata:
  name: mypod1
  labels:
    app: zomato
spec:
  containers:
    - name: cont-1
      image: nginx
      ports:
        - containerPort: 80
~
```

vim pod.yml (Creating mypod)

```
---
apiVersion: v1
kind: Pod
metadata:
   name: mypod
   labels:
      app: swiggy
spec:
   containers:
      - name: cont-1
        image: nginx
        ports:
           - containerPort: 80
```

```
pod/mypod1 created
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          3m34s   app=swiggy
mypod1   1/1     Running   0          17s     app=zomato
[root@ip-172-31-95-6 mainfest]# vim pod.yml
[root@ip-172-31-95-6 mainfest]# kubectl create -f pod.yml
pod/mypod2 created
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          8m2s    app=swiggy
mypod1   1/1     Running   0          4m45s   app=zomato
mypod2   1/1     Running   0          25s     <none>
[root@ip-172-31-95-6 mainfest]# vim pod.yml
[root@ip-172-31-95-6 mainfest]# kubectl apply -f pod.yml
Warning: resource pods/mypod2 is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by kubectl apply. kubectl appl
y should only be used on resources created declaratively by either kubectl create --save-config or kubectl apply. The missing annotation will be patched
automatically.
pod/mypod2 configured
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          9m22s   app=swiggy
mypod1   1/1     Running   0          6m5s    app=zomato
mypod2   1/1     Running   0          105s    app=uber
[root@ip-172-31-95-6 mainfest]#
```

```
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          9m22s   app=swiggy
mypod1   1/1     Running   0          6m5s    app=zomato
mypod2   1/1     Running   0          105s    app=uber
[root@ip-172-31-95-6 mainfest]# kubectl label pod mypod env=dev
pod/mypod labeled
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          12m     app=swiggy,env=dev
mypod1   1/1     Running   0          9m6s    app=zomato
mypod2   1/1     Running   0          4m46s   app=uber
[root@ip-172-31-95-6 mainfest]# kubectl label pod mypod1 env=Ops
pod/mypod1 labeled
[root@ip-172-31-95-6 mainfest]# kubectl get po --show-labels
NAME     READY   STATUS    RESTARTS   AGE     LABELS
mypod    1/1     Running   0          13m     app=swiggy,env=dev
mypod1   1/1     Running   0          9m46s   app=zomato,env=Ops
mypod2   1/1     Running   0          5m26s   app=uber
[root@ip-172-31-95-6 mainfest]#
```

```
[root@ip-172-31-95-6 mainfest]# kubectl get po -l app=zomato
NAME     READY   STATUS    RESTARTS   AGE
mypod1   1/1     Running   0          12m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l env=dev
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          15m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l env=Ops
NAME     READY   STATUS    RESTARTS   AGE
mypod1   1/1     Running   0          12m
[root@ip-172-31-95-6 mainfest]# kubectl get po app=swiggy
Error from server (NotFound): pods "app=swiggy" not found
[root@ip-172-31-95-6 mainfest]# kubectl get po -l app=swiggy
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          19m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l 'env in (dev,Ops)'
NAME     READY   STATUS    RESTARTS   AGE
mypod    1/1     Running   0          23m
mypod1   1/1     Running   0          20m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l 'app in (swiggy,zomato)'
NAME     READY   STATUS    RESTARTS   AGE
mypod    1/1     Running   0          24m
mypod1   1/1     Running   0          21m
[root@ip-172-31-95-6 mainfest]#
```


```
[root@ip-172-31-95-6 mainfest]# kubectl get po -l app=zomato
NAME     READY   STATUS    RESTARTS   AGE
mypod1   1/1     Running   0          12m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l env=dev
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          15m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l env=Ops
NAME     READY   STATUS    RESTARTS   AGE
mypod1   1/1     Running   0          12m
[root@ip-172-31-95-6 mainfest]# kubectl get po app=swiggy
Error from server (NotFound): pods "app=swiggy" not found
[root@ip-172-31-95-6 mainfest]# kubectl get po -l app=swiggy
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          19m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l 'env in (dev,Ops)'
NAME     READY   STATUS    RESTARTS   AGE
mypod    1/1     Running   0          23m
mypod1   1/1     Running   0          20m
[root@ip-172-31-95-6 mainfest]# kubectl get po -l 'app in (swiggy,zomato)'
NAME     READY   STATUS    RESTARTS   AGE
mypod    1/1     Running   0          24m
mypod1   1/1     Running   0          21m
[root@ip-172-31-95-6 mainfest]#
```