

ubuntu@ip-172-31-28-92:~$ sudo nano myefs.yaml

---

apiVersion: v1

kind: ConfigMap

metadata:

name: efs-provisioner

data:

file.system.id: fs-e4413b65

aws.region: us-east-2

provisioner.name: example.com/aws-efs

dns.name: ""

---

apiVersion: v1

kind: ServiceAccount

metadata:

name: efs-provisioner

---

kind: Deployment

apiVersion: apps/v1

metadata:

name: efs-provisioner

spec:

replicas: 1

selector:

matchLabels:

app: efs-provisioner

strategy:

type: Recreate

template:

metadata:

labels:

app: efs-provisioner

spec:

serviceAccount: efs-provisioner

containers:

- name: efs-provisioner

image: quay.io/external\_storage/efs-provisioner:latest

env:

- name: FILE\_SYSTEM\_ID

valueFrom:

configMapKeyRef:

name: efs-provisioner

key: file.system.id

- name: AWS\_REGION

valueFrom:

configMapKeyRef:

name: efs-provisioner

key: aws.region

- name: DNS\_NAME

valueFrom:

configMapKeyRef:

name: efs-provisioner

key: dns.name

optional: true

- name: PROVISIONER\_NAME

valueFrom:

configMapKeyRef:

name: efs-provisioner

key: provisioner.name

volumeMounts:

- name: pv-volume

mountPath: /persistentvolumes

volumes:

- name: pv-volume

nfs:

server: fs-e4413b65.efs.us-east-1.amazonaws.com

path: /

---

kind: StorageClass

apiVersion: storage.k8s.io/v1

metadata:

name: aws-efs

provisioner: example.com/aws-efs

---

kind: PersistentVolumeClaim

apiVersion: v1

metadata:

name: efs

annotations:

volume.beta.kubernetes.io/storage-class: "aws-efs"

spec:

accessModes:

- ReadWriteMany

resources:

requests:

storage: 1Mi

---

ubuntu@ip-172-31-28-92:~$ kubectl apply -f myefs.yaml

configmap/efs-provisioner unchanged

serviceaccount/efs-provisioner unchanged

deployment.apps/efs-provisioner created

storageclass.storage.k8s.io/aws-efs created

persistentvolumeclaim/efs created

ubuntu@ip-172-31-28-92:~$ kubectl get sc

NAME PROVISIONER RECLAIMPOLICY VOLUMEBINDINGMODE ALLOWVOLUMEEXPANSION AGE

aws-efs example.com/aws-efs Delete Immediate false 12m

ubuntu@ip-172-31-28-92:~$ kubectl get pvc

NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE

efs Pending aws-efs 12m

ubuntu@ip-172-31-28-92:~$ sudo nano pv-deploy.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: pv-deploy

spec:

replicas: 1

selector:

matchLabels:

app: mypv

template:

metadata:

labels:

app: mypv

spec:

containers:

- name: shell

image: centos:7

command:

- "bin/bash"

- "-c"

- "sleep 10000"

volumeMounts:

- name: mypd

mountPath: "/tmp/persistent"

volumes:

- name: mypd

persistentVolumeClaim:

claimName: efs

ubuntu@ip-172-31-28-92:~$ kubectl apply -f pv-deploy.yaml

deployment.apps/pv-deploy created

ubuntu@ip-172-31-28-92:~$ kubectl get deployments

NAME READY UP-TO-DATE AVAILABLE AGE

apache-deployment 0/0 0 0 2d5h

efs-provisioner 0/1 1 0 18m

nginx 0/0 0 0 4d22h

nginx-deployment 0/0 0 0 2d8h

pv-deploy 0/1 1 0 105s

ubuntu@ip-172-31-28-92:~$ echo -n "xyzIsthebest">./apikey.txt

ubuntu@ip-172-31-28-92:~$ ls

apache-service.yaml apikey.txt empty-dir-vol.yaml ingress.yaml nginx.yaml

apache.yaml calico.yaml external-storage myefs.yaml pv-deploy.yaml

ubuntu@ip-172-31-28-92:~$ kubectl create secret generic apikey --from-file=./apikey.txt

secret/apikey created

ubuntu@ip-172-31-28-92:~$ kubectl get secrets

NAME TYPE DATA AGE

apikey Opaque 1 22s

default-token-v88h7 kubernetes.io/service-account-token 3 5d8h

efs-provisioner-token-bcvwb kubernetes.io/service-account-token 3 87m

ubuntu@ip-172-31-28-92:~$ kubectl describe secrets apikey

Name: apikey

Namespace: default

Labels: <none>

Annotations: <none>

Type: Opaque

Data

====

apikey.txt: 12 bytes

ubuntu@ip-172-31-28-92:~$ sudo nano secret.yaml

apiVersion: v1

kind: Pod

metadata:

name: consumesec

spec:

containers:

- name: shell

image: centos:7

command:

- "bin/bash"

- "-c"

- "sleep 10000"

volumeMounts:

- name: apikeyvol

mountPath: "/tmp/apikey"

readOnly: true

volumes:

- name: apikeyvol

secret:

secretName: apikey

ubuntu@ip-172-31-28-92:~$ kubectl apply -f secret.yaml

pod/consumesec created

ubuntu@ip-172-31-28-92:~$ kubectl exec -it consumesec bash

kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.

[root@consumesec /]#

[root@consumesec /]# mount | grep apikey

tmpfs on /tmp/apikey type tmpfs (ro,relatime,inode64)

[root@consumesec /]# cat /tmp/apikey/apikey.txt

xyzIsthebest[root@consumesec /]#exit

ubuntu@ip-172-31-28-92:~$ kubectl get nodes

NAME STATUS ROLES AGE VERSION

ip-172-31-19-243 Ready <none> 5d v1.19.16

ip-172-31-28-92 Ready master 5d9h v1.19.16

ip-172-31-33-13 Ready <none> 3d14h v1.19.16

ubuntu@ip-172-31-28-92:~$ kubectl taint nodes ip-172-31-19-243 key=value:NoSchedule

node/ip-172-31-19-243 tainted

ubuntu@ip-172-31-28-92:~$ sudo nano taints-pod.yaml

apiVersion: v1

kind: Pod

metadata:

name: nginx

labels:

env: test

spec:

containers:

- name: nginx

image: nginx

imagePullPolicy: IfNotPresent

tolerations:

- key: "example-key"

operator: "Exists"

effect: "NoSchedule"

ubuntu@ip-172-31-28-92:~$ kubectl create -f taints-pod.yaml

pod/nginx created

ubuntu@ip-172-31-28-92:~$ kubectl get pods

NAME READY STATUS RESTARTS AGE

consumesec 1/1 Running 0 31m

nginx 1/1 Running 0 95s

sharevol 2/2 Running 6 16h

ubuntu@ip-172-31-28-92:~$ kubectl describe po nginx

Name: nginx

Namespace: default

Priority: 0

Node: ip-172-31-33-13/172.31.33.13

Start Time: Wed, 17 Nov 2021 17:08:44 +0000

Labels: env=test

Annotations: cni.projectcalico.org/containerID: b3a92fcc03f88e5622eb96226929353d531b1d8e56b8588f16615a5f4dcfe476

cni.projectcalico.org/podIP: 192.168.133.75/32

cni.projectcalico.org/podIPs: 192.168.133.75/32

Status: Running

IP: 192.168.133.75

IPs:

IP: 192.168.133.75

Containers:

nginx:

Container ID: docker://3e863edcd5ccfb4adda0269c59657b0993b744df1aa778d659779884e4454a53

Image: nginx

Image ID: docker-pullable://nginx@sha256:097c3a0913d7e3a5b01b6c685a60c03632fc7a2b50bc8e35bcaa3691d788226e

Port: <none>

Host Port: <none>

State: Running

Started: Wed, 17 Nov 2021 17:08:52 +0000

Ready: True

Restart Count: 0

Environment: <none>

Mounts:

/var/run/secrets/kubernetes.io/serviceaccount from default-token-v88h7 (ro)

Conditions:

Type Status

Initialized True

Ready True

ContainersReady True

PodScheduled True

Volumes:

default-token-v88h7:

Type: Secret (a volume populated by a Secret)

SecretName: default-token-v88h7

Optional: false

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations: example-key:NoSchedule op=Exists

node.kubernetes.io/not-ready:NoExecute op=Exists for 300s

node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:

Type Reason Age From Message

---- ------ ---- ---- -------

Normal Scheduled 2m26s default-scheduler Successfully assigned default/nginx to ip-172-31-33-13

Normal Pulling 2m25s kubelet Pulling image "nginx"

Normal Pulled 2m19s kubelet Successfully pulled image "nginx" in 5.958111066s

Normal Created 2m18s kubelet Created container nginx

Normal Started 2m18s kubelet Started container nginx