Deploy Flannel on k8s

It is quite easy to deploy Flannel on k8s, and there are a few methods too. More detailed info can be seen here. In this document, I will just deploy Flannel with Helm on k8s [1].

1 Environment Description

If you have followed “01_setup_k8s_cluster_on_qemu/kvm_vms-20240622”, now, you should have a k8s 1.30 cluster with 1 control-plane (Fedora 40 QEMU/KVM VM) and two nodes (Fedora 40 QEMU/KVM VMs) as follows.

```
[root@controlPlane ~]# kubectl get nodes -A -o wide
NAME           STATUS     ROLES           AGE   VERSION   INTERNAL-IP       EXTERNAL-IP
controlplane   NotReady   control-plane   63m   v1.30.2   192.168.124.249   <none>
node1          NotReady   <none>          62m   v1.30.2   192.168.124.130   <none>
node2          NotReady   <none>          62m   v1.30.2   192.168.124.220   <none>
[root@controlPlane ~]#
[root@controlPlane ~]# kubectl get pods -A
NAMESPACE     NAME                                   READY   STATUS    RESTARTS   AGE
kube-system   coredns-7b5944fdcf-6nb7c               0/1     Pending   0          63m
kube-system   coredns-7b5944fdcf-8vzhb               0/1     Pending   0          63m
kube-system   etcd-controlplane                      1/1     Running   0          64m
kube-system   kube-apiserver-controlplane            1/1     Running   0          64m
kube-system   kube-controller-manager-controlplane   1/1     Running   0          64m
kube-system   kube-proxy-hckw5                       1/1     Running   0          62m
kube-system   kube-proxy-jsvkv                       1/1     Running   0          63m
kube-system   kube-proxy-qfs2z                       1/1     Running   0          62m
kube-system   kube-scheduler-controlplane            1/1     Running   0          64m
```

Note: core-dns-* pods are still in “Pending” status cause there is no pod network add-on installed in the k8s cluster yet.

2 Deploy Flannel on k8s via Helm

First, make sure “helm” is installed on the control plane. Otherwise, it could be easily installed with the command below.

```
[root@controlPlane ~]# dnf install helm -y
```

Then, you could deploy Flannel on k8s via helm as follows.

```
# Needs manual creation of namespace to avoid helm error
[root@controlPlane ~]# kubectl create ns kube-flannel
namespace/kube-flannel created

[root@controlPlane ~]# kubectl label --overwrite ns kube-flannel
pod-security.kubernetes.io/enforce=privileged
namespace/kube-flannel labeled

[root@controlPlane ~]# helm repo add flannel https://flannel-io.github.io/flannel/
"flannel" has been added to your repositories

[root@controlPlane ~]# helm install flannel --namespace kube-flannel flannel/flannel
NAME: flannel
LAST DEPLOYED: Mon Jul  1 09:03:57 2024
NAMESPACE: kube-flannel
STATUS: deployed
REVISION: 1
TEST SUITE: None
```

3 Post-deployment Check

If you deploy Flannel successfully, then you should be able to see core-dns-* pods are running now, and three kube-flannel-ds-* pods are running in “kube-flannel” namespace now. So each node (1 control plane and 2 nodes) has one kube-flannel-ds-* pod scheduled on itself.

```
[root@controlPlane ~]# crictl images | grep flannel
docker.io/flannel/flannel-cni-plugin                              v1.4.1-flannel1     1e3c860c213d6       10.6MB
docker.io/flannel/flannel                                         v0.25.4             e6c43605b7140       82.1MB
```

```
[root@controlPlane ~]# kubectl get pods -A -o wide
NAMESPACE      NAME                                   READY   STATUS    RESTARTS   AGE    IP                NODE
kube-flannel   kube-flannel-ds-jl5s6                  1/1     Running   0          27m    192.168.124.220   node2
kube-flannel   kube-flannel-ds-lbdsd                  1/1     Running   0          27m    192.168.124.249   controlplane
kube-flannel   kube-flannel-ds-mf8p4                  1/1     Running   0          27m    192.168.124.130   node1
kube-system    coredns-7b5944fdcf-6nb7c               1/1     Running   0          129m   10.244.1.3        node1
kube-system    coredns-7b5944fdcf-8vzhb               1/1     Running   0          129m   10.244.1.2        node1
kube-system    etcd-controlplane                      1/1     Running   0          129m   192.168.124.249   controlplane
kube-system    kube-apiserver-controlplane            1/1     Running   0          129m   192.168.124.249   controlplane
kube-system    kube-controller-manager-controlplane   1/1     Running   0          129m   192.168.124.249   controlplane
kube-system    kube-proxy-hckw5                       1/1     Running   0          127m   192.168.124.130   node1
kube-system    kube-proxy-jsvkv                       1/1     Running   0          129m   192.168.124.249   controlplane
kube-system    kube-proxy-qfs2z                       1/1     Running   0          127m   192.168.124.220   node2
kube-system    kube-scheduler-controlplane            1/1     Running   0          129m   192.168.124.249   controlplane
```

4 Potential Issues

The most possible issue you might hit during deployment is “ErrImagePull”, namely, failure to pull flannel-related images. Here is an example.

```
[root@controlPlane ~]# kubectl get pods -A
NAMESPACE      NAME                                READY   STATUS             RESTARTS   AGE
kube-flannel   kube-flannel-ds-jl5s6               0/1     Init:ErrImagePull   0          31s
kube-flannel   kube-flannel-ds-lbdsd               0/1     Init:ErrImagePull   0          31s
kube-flannel   kube-flannel-ds-mf8p4               0/1     Init:ErrImagePull   0          31s
```

In this case, the three nodes in k8s cluster could not pull the images successfully, you could pull the images on some other machines, and then use “docker save” to export the images.

```
jingyan@Jings-MacBook-Pro Downloads % docker pull
docker.io/flannel/flannel-cni-plugin:v1.4.1-flannel1
v1.4.1-flannel1: Pulling from flannel/flannel-cni-plugin
4edfc05e3af2: Already exists
5a1fdc260f39: Pull complete
Digest: sha256:e88c0d84fa89679eb6cb6a28bc257d652ced8d1b2e44d54a592f0a2cd85dba53
Status: Downloaded newer image for flannel/flannel-cni-plugin:v1.4.1-flannel1
docker.io/flannel/flannel-cni-plugin:v1.4.1-flannel1

jingyan@Jings-MacBook-Pro Downloads % docker save -o flannel-cni-plugin-v1.4..tar1
docker.io/flannel/flannel-cni-plugin:v1.4.1-flannel1

jingyan@Jings-MacBook-Pro Downloads % docker pull docker.io/flannel/flannel:v0.25.4
v0.25.4: Pulling from flannel/flannel
Digest: sha256:17415d91743e53fc4b852676a30a08915f131a2b6848d891ba5786eacd447076
Status: Image is up to date for flannel/flannel:v0.25.4
docker.io/flannel/flannel:v0.25.4

jingyan@Jings-MacBook-Pro Downloads % docker save -o flannel-v0.25.4.tar
docker.io/flannel/flannel:v0.25.4
```

Finally use “crictl import” to load the images on the three nodes.

```
[root@controlPlane ~]# ctr -n=k8s.io images import flannel-cni-plugin-v1.4.1.tar
unpacking docker.io/flannel/flannel-cni-plugin:v1.4.1-flannel1
(sha256:87a6934d598f7f7bc502095affcbcdaa919a5572eccfda27d136932eebb98ab3)...done

[root@controlPlane ~]# ctr -n=k8s.io images import flannel-v0.25.4.tar
unpacking docker.io/flannel/flannel:v0.25.4
(sha256:5e6ccd43d36a89434b6c6f15e3ba22b8870c07540860b62340667a94814e97a8)...done
```

References

[1] https://github.com/flannel-io/flannel?tab=readme-ov-file#deploying-flannel-manually