Installing Network Module:

Installng Calico Networking:

https://docs.projectcalico.org/getting-started/kubernetes/quickstart

Install Calico

1. Install the Tigera Calico operator and custom resource definitions.

```
2. kubectl create -f https://docs.projectcalico.org/manifests/tigera-operator.
   yaml
```

3. Install Calico by creating the necessary custom resource. For more information on configuration options available in this manifest, see the installation reference.

```
4. kubectl create -f https://docs.projectcalico.org/manifests/custom-resources
   .yaml
```

```
$ cat custom-resources.yaml
# This section includes base Calico installation configuration.
# For more information, see: https://docs.projectcalico.org/v3.19/reference
/installation/api#operator.tigera.io/v1.Installation
apiVersion: operator.tigera.io/v1
kind: Installation
metadata:
  name: default
spec:
  # Configures Calico networking.
  calicoNetwork:
    # Note: The ipPools section cannot be modified post-install.
    ipPools:
    - blockSize: 26
      cidr: 10.244.0.0/16
      encapsulation: VXLANCrossSubnet
      natOutgoing: Enabled
      nodeSelector: all()
```

Note: Before creating this manifest, read its contents and make sure its settings are correct for your environment. For example, you may need to change the default IP pool CIDR to match your pod network CIDR.

5. Confirm that all of the pods are running with the following command.

```
6. watch kubectl get pods -n calico-system
```

Wait until each pod has the `STATUS` of `Running`.

Note: The Tigera operator installs resources in the `calico-system` namespace. Other install methods may use the `kube-system` namespace instead.

7. Remove the taints on the master so that you can schedule pods on it.

```
8. kubectl taint nodes --all node-role.kubernetes.io/master-
```

It should return the following.

```
node/<your-hostname> untainted
```

9. Confirm that you now have a node in your cluster with the following command.

```
10. kubectl get nodes -o wide
```

It should return something like the following.

```
NAME              STATUS   ROLES    AGE   VERSION   INTERNAL-IP   EXTERNAL-
IP   OS-IMAGE             KERNEL-VERSION    CONTAINER-RUNTIME

<your-hostname>   Ready    master   52m   v1.12.2   10.128.0.28   <none>
Ubuntu 18.04.1 LTS   4.15.0-1023-gcp   docker://18.6.1
```

Congratulations! You now have a single-host Kubernetes cluster with Calico.

Installing Flannel Network

```
devops@kubeubnode1:~$ kubectl apply -f
https://raw.githubusercontent.com/coreos/
flannel/master/Documentation/kube-flannel.yml
```

https://docs.projectcalico.org/getting-started/kubernetes/flannel/migration-from-flannel

Migrate from flannel networking to Calico networking, live migration

1. Install Calico.

```
2. kubectl apply -f https://docs.projectcalico.org/manifests/flannel-migration
   /calico.yaml
```

3. Start the migration controller.

```
4. kubectl apply -f https://docs.projectcalico.org/manifests/flannel-migration
   /migration-job.yaml
```

You will see nodes begin to update one at a time.

5. Monitor the migration.

6. `kubectl get jobs -n kube-system flannel-migration`

   When the host node is upgraded, the migration controller may be rescheduled several times. The installation is complete when the output of the above command shows 1/1 completions. For example:

```
NAME                COMPLETIONS   DURATION   AGE
flannel-migration   1/1           2m59s      5m9s
```

7. Delete the migration controller.

```
kubectl delete -f https://docs.projectcalico.org/manifests/flannel-migration/migrat
```