1.

```
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

You can now join any number of control-plane nodes by copying certificate authorities
and service account keys on each node and then running the following as root:

  kubeadm join 192.168.236.143:6443 --token r0y9r0.yykws13572vj3sks \
        --discovery-token-ca-cert-hash sha256:d5e6df034da690d279c3833cbe02cb871bc096e141aeb2f60780d6
a43c6c425b \
        --control-plane

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 192.168.236.143:6443 --token r0y9r0.yykws13572vj3sks \
        --discovery-token-ca-cert-hash sha256:d5e6df034da690d279c3833cbe02cb871bc096e141aeb2f60780d6
a43c6c425b
```

```
root@worker:~# kubeadm join 192.168.236.143:6443 --token r0y9r0.yykws13572vj3sks \
        --discovery-token-ca-cert-hash sha256:d5e6df034da690d279c3833cbe02cb871bc096e141aeb2f60780d6
a43c6c425b
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config
 -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.
env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...

This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

root@worker:~#
```

```
root@master:~# kubectl cluster-info
Kubernetes control plane is running at https://192.168.236.143:6443
CoreDNS is running at https://192.168.236.143:6443/api/v1/namespaces/kube-system/services/kube-dns:d
ns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
root@master:~#
```

```
root@master:~# kubectl get nodes
NAME     STATUS     ROLES           AGE   VERSION
master   NotReady   control-plane   35m   v1.25.16
worker   NotReady   <none>          19s   v1.25.16
root@master:~#
```

```
root@master:~# kubectl apply -f https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml
namespace/kube-flannel created
serviceaccount/flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
```

```
root@master:~# kubectl get pods -A
NAMESPACE      NAME                              READY   STATUS    RESTARTS   AGE
kube-flannel   kube-flannel-ds-6ffnm             1/1     Running   0          9s
kube-flannel   kube-flannel-ds-nlrq8             1/1     Running   0          9s
kube-system    coredns-565d847f94-46l7p          1/1     Running   0          5m33s
kube-system    coredns-565d847f94-s7wpj          1/1     Running   0          5m33s
kube-system    etcd-master                       1/1     Running   1          5m47s
kube-system    kube-apiserver-master             1/1     Running   1          5m44s
kube-system    kube-controller-manager-master    1/1     Running   0          5m46s
kube-system    kube-proxy-fxf7p                  1/1     Running   0          75s
kube-system    kube-proxy-r888l                  1/1     Running   0          5m34s
kube-system    kube-scheduler-master             1/1     Running   1          5m47s
```

2.

```
root@master:~# vi deployment.yaml
root@master:~# kubectl apply -f deployment.yaml
deployment.apps/nginx-deployment created
root@master:~# kubectl get deployment
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   0/3     3            0           19s
```

```
root@master:~# kubectl get pods
NAME                                READY   STATUS    RESTARTS      AGE
nginx-deployment-86dcfdf4c6-b2xs5   1/1     Running   1 (33m ago)   34h
nginx-deployment-86dcfdf4c6-ql5dz   1/1     Running   1 (33m ago)   34h
nginx-deployment-86dcfdf4c6-pmdb4   1/1     Running   1 (33m ago)   34h
root@master:~#
```

3.

On master: curl https://get.k3s.io | INSTALL_K3S_EXEC=”server –advertise-address 192.168.236.143” sh

cat /var/lib/rancher/k3s/server/token

on worker: curl https://get.k3s.io |K3S_URL=https:// 192.168.236.143:6443 K3S_TOKEN=’ K1061233892zb3d5db5449e6e055f2829fb81fb5acfd6168da1f8c8bd488cb6a3cbe::server:4ccae6e9dc9d8c9255c4274450a459c9’ sh –

```
root@master:~# kubectl get nodes
NAME     STATUS   ROLES                  AGE    VERSION
master   Ready    control-plane,master   2d13h   v1.28.5+k3s1
root@master:~#
```