

Now we are ready to set up the Kubernetes scheduler. This lesson will walk you through the process of configuring the kube-scheduler `systemd` service. Since this is the last of the three control plane services that need to be set up in this section, this lesson also guides you through enabling and starting all three services on both control nodes. Finally, this lesson shows you how to verify that your Kubernetes controllers are healthy and working so far. After completing this lesson, you will have a basic, working, Kubernetes control plane distributed across your two control nodes.

You can configure the Kubernetes Scheduler like this.

Copy `kube-scheduler.kubeconfig` into the proper location:

```
sudo cp kube-scheduler.kubeconfig /var/lib/kubernetes/
```

Generate the kube-scheduler yaml config file.

```
cat << EOF | sudo tee /etc/kubernetes/config/kube-scheduler.yaml
apiVersion: componentconfig/v1alpha1
kind: KubeSchedulerConfiguration
clientConnection:
  kubeconfig: "/var/lib/kubernetes/kube-scheduler.kubeconfig"
leaderElection:
  leaderElect: true
EOF
```

Create the kube-scheduler `systemd` unit file:

```
cat << EOF | sudo tee /etc/systemd/system/kube-scheduler.service
[Unit]
Description=Kubernetes Scheduler
Documentation=https://github.com/kubernetes/kubernetes

[Service]
ExecStart=/usr/local/bin/kube-scheduler \\\
  --config=/etc/kubernetes/config/kube-scheduler.yaml \\\
  --v=2
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
EOF
```

Start and enable all of the services:

```
sudo systemctl daemon-reload
sudo systemctl enable kube-apiserver kube-controller-manager kube-scheduler
sudo systemctl start kube-apiserver kube-controller-manager kube-scheduler
```

It's a good idea to verify that everything is working correctly so far:

Make sure all the services are `active (running)` :

```
sudo systemctl status kube-apiserver kube-controller-manager kube-scheduler
```

Use `kubectl` to check component statuses:

```
kubectl get componentstatuses --kubeconfig admin.kubeconfig
```

You should get output that looks like this:

NAME	STATUS	MESSAGE	ERROR
controller-manager	Healthy	ok	
scheduler	Healthy	ok	
etcd-0	Healthy	{"health": "true"}	
etcd-1	Healthy	{"health": "true"}	